2MAPPS (2MASS Production Processing System)

Functional Design Document

Version 3.4

2/12/96

Edited by J.W. Fowler and E.L. Kopan

_____

C. A. Beichman

_____

R. Cutri

_____

T. J. Chester

_____

E. L. Kopan

_____

J. W. Fowler

Table of Contents

1.0 Purpose and Summary 4
1.1 Reference Documents 5
1.2 Version History 5
1.3 Lien List 7

2.0 System Description and Requirements 7
2.1 System Functional Requirement 7
2.2 System Overview and Block Diagram 10
2.3 System Throughput Requirements 11
2.4 System Operational Requirements 14
2.5 System Operational Monitoring Requirements 14
2.6 Subsystem Coding and Documentation Requirements 15
2.7 Subsystem Testability Requirements 17
2.8 System Delivery and Configuration Control 17

3.0 Subsystem Requirements 19
3.1 EXEC (Execution Control Subsystem) 19
3.2 TAPELOAD (Tape Loading to Disk) 21
3.3 DARKS (Darks, Masks, Responsivity Processing) 22
3.4 PCP (Pipeline Control Processor) 23
3.5 PIXCAL (Pixel Calibration Processing) 24
3.6 PIXPHOT (Pixel Photometric Processing) 24
3.7 DFLAT (Frame Flattening) 25
3.8 FREXAS (Frame Extraction, Seeing, Statistics) 27
3.9 MAPCOR (Merge And Purge, Aperture Corrections) 28
3.10 PICMAN (Image Generation) 30
3.11 PROPHOT (PSF-Fit Photometry) 32
3.12 BANDMERGE (Point-Source Bandmerging) 34
3.13 POSMAN (Position Reconstruction) 34
3.14 GALWORKS (Extended Source Processing) 38
3.15 CALMON (Photometric Calibration) 41
3.16 QUALITY (Processing Quality Checking) 43
3.17 DBMAN (Database Processing) 45
3.18 MPCAT (Minor-Planet Catalog Processing) 47

Appendix A. Baseline Hardware Configuration 48

Appendix B. Throughput Estimates and Hardware Requirements 50

Appendix C. Hardware Failure Contingency Plans 52

Appendix D. Dependencies on Camera, Telescope and Survey Design 53

Appendix E. Intersubsystem Function RDFRAME 55

Appendix F. Subsystem Function CONMAN 57

Appendix G. Intersubsystem Function SEEMAN 58

Glossary 59

1.0 Purpose and Summary

The purpose of this document is to specify the functional design of the 2 Micron All-Sky Survey (2MASS) production data processing system, and to specify the subsystem components and the functional requirements on each subsystem. The detailed design of the subsystem components will be specified in separate Subsystem Design Specification documents, one for each subsystem.

The 2MASS Production Processing System (2MAPPS) will process all of the data collected for the 2MASS project from raw data tapes as delivered from the observatory, through the final point and extended source catalogs and image products that will ultimately be delivered to NSSDC and the astronomical community.

The 2MASS Project (as defined in the 2MASS Development Phase Proposal, Ref. 1.1-1) is intended to produce source catalogs and images of the entire sky in the J, H, and K_s near-infrared wavelength bands with a pixel size of 1 arcseconds. Approximately 19 Tbytes (uncompressed) of raw data from two observatories will be processed to create catalogs expected to contain about 10^8 stars and 10^6 galaxies, and 4 Tbytes of images.

The processing task is within the capabilities of a small cluster of modern workstations if the system architecture and software components are designed with performance in mind. To this end, the baseline hardware configuration is summarized in Appendix A so that the processing context may be considered. The design will be as platform-independent as practical consistent with the throughput requirements and the baseline hardware configuration.

This document is also intended to specify the minimum coding, documentation, and testing standards for the IPAC software, such that there is sufficient commonality in the component parts of the system to insure their smooth integration, and sufficient documentation to ensure continuity over the life of the project.

This document specifically does not cover the Survey Coverage Strategy software (onsite *or* offsite), or the Catalog Production Software, both of which will be addressed in separate documents. The Survey Coverage Strategy software and the on-site data handling and quality assurance are the responsibility of UMASS, although IPAC will assist if requested. The Catalog Production Software design will be addressed at a later date. The Catalog production will incorporate identification of asteroids, and association with other astronomical catalogs.

1.1 Reference Documents

1) 2MASS Development Phase Proposal

2) 2MAPPS Requirements Document

3) 2MASS Observatory/IPAC Tape Interface Specification

4) 2MASS Survey Strategy Requirements

5) 2MASS Level 1 Science Objectives and Specifications

1.2 Version History

1.2.1.0) Version 1.0 - 17 November, 1993: Initial Release

1.2.2.0) Version 2.0 - 1 November, 1994

The following changes are incorporated in Version 2.0.

1.2.2.0.1) All Read1 frames are now received at IPAC (previously only about 25% were to be received, i.e., those containing sources bright enough to cause photometric linearity or saturation problems).

1.2.2.0.2) Calibration frames need not be grouped at the beginning of the input tapes received at IPAC from the observatories.

1.2.2.0.3) Data for different bands may be arranged on the tapes in the order most convenient for the observatory computers.

1.2.2.0.4) To accomodate items (2.0.1) through (2.0.3), an entire night's worth of data will be stored on disk as it is processed, taking advantage of improved magnetic storage media.

1.2.2.0.5) Frame-to-frame offsets will be computed using data from all bands to obtain optimal estimates before proceeding to image construction and point-source extraction; this necessitates breaking the previous PIXELWORKS subsystem into two new subsystems, PIXCAL and PIXPHOT; PIXCAL handles DFLAT and FREXAS, while PIXPHOT handles PICMAN and PROPHOT.

1.2.2.0.6) A new GALWORKS subsystem has been substituted for the previous one.

1.2.2.0.7) A liens section has been added (section 1.3).

1.2.3.0) Version 3.0 - 15 September, 1995

The following changes are incorporated in Version 3.0.

1.2.3.0.1) Section 2.6 expanded

1.2.3.0.2) Section 2.8 added

1.2.3.0.3) Section 3.7 (DFLAT) modified for improved frame-flattening method; section 3.3 (DARKS) now includes processing responsivity images

1.2.3.0.4) MAPCOR subsystem added; subsections renumbered in section 3

1.2.3.0.5) Sections 3.12 and 3.14 modified for shifting of point-source band-filling to galaxy processing subsystem

1.2.3.0.6) FREXAS subsystem renamed to FREXAS, with added functions

1.2.3.0.7) POSMAN subsystem module POSFRM replaces COFF and uses astrometric catalogs to reconstruct frame positions; final point-source positions computed by POSMAN module POSPTS

1.2.3.1) Version 3.1 - 19 September, 1995

The following changes are incorporated in Version 3.1.

1.2.3.1.1) Section 1.2 reformatted

1.2.3.1.2) Miscellaneous subsystem relationships clarified

1.2.3.1.3) Miscellaneous editorial touchups

1.2.3.2) Version 3.2 - 2 November, 1995

The following changes are incorporated in Version 3.2.

1.2.3.2.1) Miscellaneous editorial touchups

1.2.3.2.2) PSFMAN replaced by SEEMAN, with different system usage

1.2.3.2.3) Revised CPU allocations

1.2.3.3) Version 3.3 - 30 November, 1995: Added coding requirements concerning precision of calculations and intermediate file formats affecting position and photometric parameters (section 2.6).

1.2.3.4) Version 3.4 - 12 February, 1996: Added the asteroid-tagging subsystem MPCAT to the scan pipeline.

1.3 Lien List

1) The NICMOS bias dependence on illumination has necessitated a change to the flat computation method, as reflected in the previous section, item 3.3. Details concerning whether twilight flats, dome flats, both, or some other source of responsivity information will be used are TBD.

2) A new approach to frame-offset computation and source position reconstruction has been adopted, as reflected in the previous section, item 3.7; this is the most promising approach at this time, but its viability has yet to be firmly proved.

2.0 System Description and Requirements

2.1 System Functional Requirement

The 2MAPPS will be designed to satisfy the requirements contained in the 2MASS Development Phase Proposal (Ref. 1.1-1), the 2MASS Production Processing System Requirements (Ref. 1.1-2), and the 2MASS Level 1 Science Objectives and Specifications (Ref. 1.1-5).

Figure 1. 2MAPPS PROCESSING

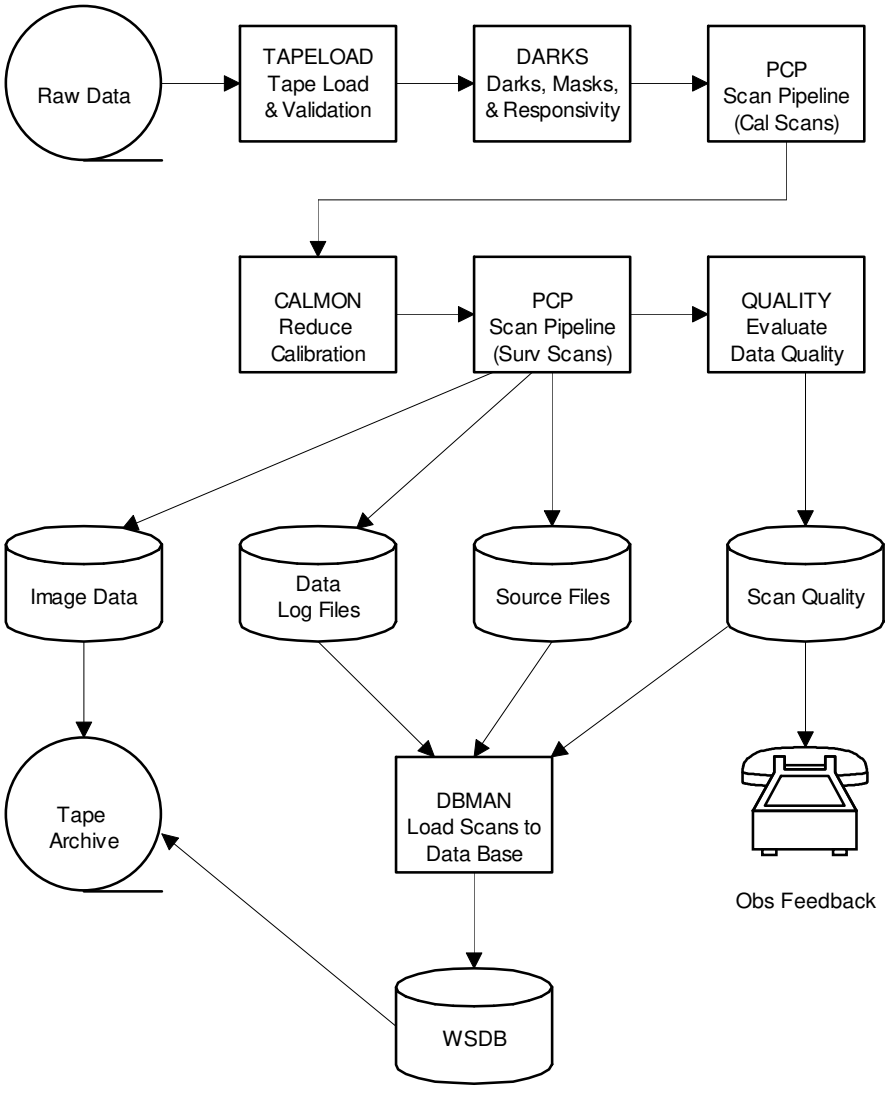
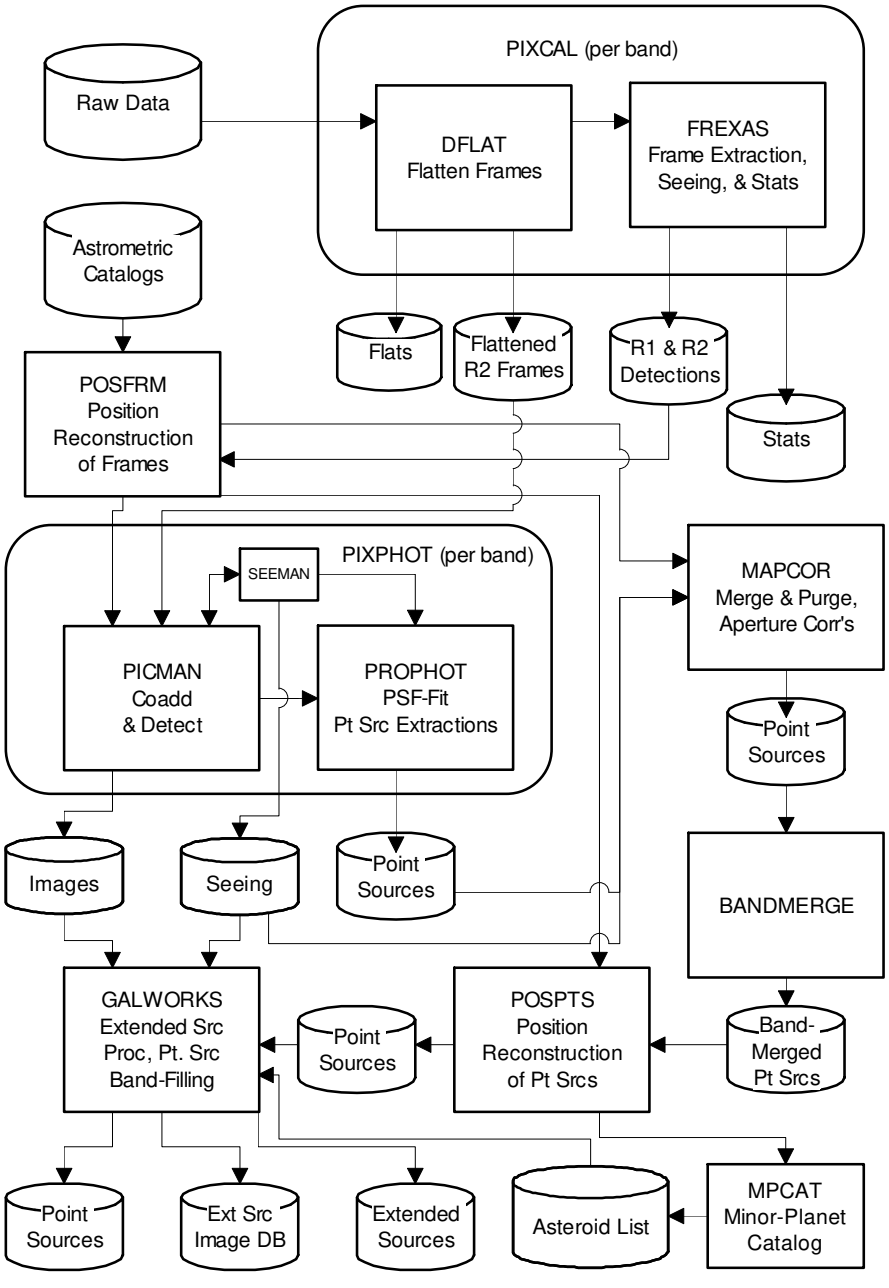


Figure 2. 2MAPPS SCAN PROCESSING¡



2.2 System Overview and Block Diagram

2.2.1 System Block Diagram

Figure 1 shows a top-level block diagram for the 2MAPPS. Figure 2 shows a high-level block diagram of the per scan processing.

2.2.2 System Processing Overview

The 2MASS Production Processing System (2MAPPS) is designed to perform all required processing on the data from a single night (see Ref. 1.1-3) from a single observatory beginning with receipt at IPAC to placement in the Working Survey Data Base (WSDB). This processing may be summarized as follows:

- load one night's data from one observatory:
    - read tapes to disk
    - check vs data logs for data completeness
    - check data integrity
    - write backup tapes
- compute and analyze darks
- for each scan:
    - get next scan
    - per band:
        - data preparation processing:
            - compute flats and analyze
            - per Read2-Read1 and Read1 frame:
                - get next frame
                - linearize
                - bias subtract and flatten frame
                - compute background, noise, etc.
                - detect point sources
                - estimate seeing parameters (Read2-Read1 only)
    - compute multi-band frame offsets
    - per band:
        - point source and image processing:
            - per frame:
                - get next frame
                - linearize
                - bias subtract and flatten
                - push frame to coadder:
                    - coadd panel
                    - detect sources in panel

- output finished coadd if done and re-init
- push frame to extractor:
- extract detected panel sources
- merge Read1 and Read2-Read1 source lists
- band merge point sources
- position reconstruction:
- read source lists and determine mapping
- map source lists from scan coordinates to RA and Dec
- catalog asteroids covered by scan
- galaxy processing:
- read coadded images and threshold extract
- smooth and re-threshold for low surface brightness objects
- collate threshold extractions with point source extractions
- analyze and classify sources in collated list
- calibration:
- compute and analyze calibration for night
- apply air mass and zero point calibrations to source lists
- update image headers for calibration and position
- run QA jobs and update quality logs
- confirm sources in scan overlaps and analyze
- flag low quality scans
- create nightly summary plots and output
- archive images and sources to tape
- load source lists to WSDB
- update survey scan logs and survey execution plan

Since I/O bandwidth is a potentially limiting factor, the system design attempts to minimize I/O by reducing the number of times the image data are read and written to disk. This includes not writing out the frame data at intermediate stages of processing (differenced frames, linearized frames) but reading in only the raw data and writing out the coadded images during production processing. In order to compute the frame offsets simultaneously from all three bands, however, it is necessary to write the flattened frames to disk and read them in again later. If needed for debugging purposes, the other intermediate processing results will have to be regenerated by rerunning the scans with appropriate flags set. Likewise, large volume intermediate data products will not be saved to tape, in order to reduce the already daunting volume of the tape archive (which must also be backed up).

2.3 System Throughput Requirements

The system processing shall keep pace (with a time lag) with the data capture rates from two observatories, with a factor of two allowance for reprocessing. This requirement will be met in two shifts per day (16 hours), 7 days/week.

Contingency for algorithm growth will be provided by expected increases in hardware performance

in the two-year period prior to final acquisitions, hardware upgradibility, and the ready availability of additional hardware at reasonable cost.

The system throughput requirement is also broken down to the subsystem level as preliminary guidelines for subsystem design.

The anticipated data acquisition rates are as follows:

Hemisphere	Ave. avail hrs/night	Photometric hrs/night	Marginal hrs/night	Ave. proc. hrs/night
Northern	10	3	2	5
Southern	10	5	2	7
Average hours of data/night:				12
x 2 for reprocessing:				24

Thus we need processing throughput capacity for 24 hours of data in a 16 hour period. This means we need the capacity to process 1.5 hours of 3-band data from one observatory per hour.

The baseline hardware configuration (see App. A) has 24 Sparc 10 class cpus in the Production cluster, so the required throughput of 3-band data from a single observatory per processor is:

24 hours data/ 16 hours/ 24 processors approx 0.07 hr data/hr/processor

This matches the estimated throughput rate based on extrapolations from prototype software (see App. B). A 100% margin in floating point performance is available by plugging in more processor boards into the baseline hardware. Additional margin is provided by purchasing more machines, and also by expected increases in hardware performance in the two year period before the final acquisitions are made.

Each of these processors is rated at ~96 floating point Specmarks (SPECfp92), giving a baseline processing power of ~2300 SPECfp92 or ~720 Mflops.

The preliminary throughput allocation for each subsystem component is as follows (where the system throughput total is 1.0):

Throughput Allocations per Subsystem

Process	Function	Fraction
TAPELOAD	**load and validate input tapes**	**---**
EXEC	**pipeline execution control program**	**---**
MONITOR	**pipeline execution monitor program**	**---**
DARKS	**create and analyze darks**	**0.003**
RDFRAME	**read frames, parse header files**	**0.05**
DFLAT	**create, analyze, apply sky flats**	**0.09**
FREXAS	**compute per frame statistics**	**0.04**
POSFRM	**compute frame offsets from all bands**	**0.007**
PICMAN	**combine overlapping frames into images**	**0.17**
PIXPHOT	**detect compact sources**	**0.20**
PROPHOT	**detect and measure compact sources**	**0.20**
MAPCOR	**merge and purge and aperture corrections**	**0.02**
CALMON	**analyze calibration data and calibrate sources**	**0.002**
BANDMERGE	**bandmerge point sources**	**0.006**
POSPTS	**reconstruct source positions**	**0.006**
MPCAT	**minor-planet catalog**	**---**
GALWORKS	**detect and measure extended sources**	**0.20**
CONMAN	**confirm and analyze sources in overlaps**	**0.002**
QUALITY	**analyze and monitor data and processing quality**	**0.004**
TOTAL		**1.00**

This table is intended to serve as a guideline and a design goal for each subsystem. Backup information is provided in Appendix B.

The TAPELOAD, EXEC, and MONITOR functions are bookkept separately. The EXEC and MONITOR functions will not use appreciable resources and they will execute outside of the 2MASS Production hardware cluster. The TAPELOAD function will be limited by the tape input unit throughput times the number of tape drives utilized. The time to read a full 8mm tape is on the order of 3 hours. The current plan is to use 3-6 drives in parallel during the loading process.

The data base loading process (DBMAN) process will run on a separate data base machine, so is not included in the above table. Tests have shown that the data base loading and maintenance operations can be performed in less than one hour throughput per observing night (average) on the baseline database machine. The baseline data base software will be SYBASE.

These throughput requirements are based on the assumption of three chips per observatory, an integration time of 1.3 seconds, an in-scan overlap of depth 6, and a pixel size of 2.0 arcsec. The processing time scales linearly with the data volume. In addition, roughly 50% of the processing time scales as the ratio of the number of pixels required to represent the psf.

2.4 System Operational Requirements

The human interaction required to run the 2MAPPS system will be minimized.

Concise summary plots and tables will be created with each night's run. These will allow the run to be verified with less than a total of one work-hour of attention by cognizant personnel in the nominal case (after the validation phase). The goal will be to allow validation in less than 30 minutes in the nominal case. In addition, extensive tools will be provided that will allow efficient investigation of any problems evident from the summary information.

As a baseline, the human interaction required will be as follows:

1) Load the input and output tapes for a given night's observing into tape stack loaders.

2) Start the tape load and validation job.

3) Check the tape validation summary, and take corrective action if the tapes were not validated (the tape validation job will also make back-up tapes if not already done).

4) Start the production run pipeline for the night's processing.

5) Start the execution monitor job (see Sec. 2.5).

6) Collect and organize the pipeline summary output.

7) Validate the night's processing by checking the pipeline summary output.

8) Initiate the processing to integrate the night's run into the Working Survey Database.

2MASS observing data tapes will be validated no more that 48 hours (TBD) after receipt at IPAC so that data problems may be fed back to the observatory as soon as possible. Every effort will be made to perform the validation within this interval, even if the complete processing lags behind, in which case steps 1--3 above will be used as a tape validation process.

The production processing system hardware will be in a separate network (see App. A) with access restricted by password to designated IPAC 2MASS and DMT personnel.

2.5 System Operational Monitoring Requirements

Provisions will be made throughout the subsystem and system design to facilitate the monitoring of pipeline execution and the tracking and isolation of execution errors.

A pipeline execution monitor program will be provided which will allow data management and software development personnel to monitor pipeline execution progress and display execution diagnostics in a hierarchal summary to facilitate easy detection and isolation of problems. This program will operate independently of the pipeline execution.

One or more copies of the pipeline monitor will be able to execute asynchronously on separate workstations from the pipeline, accessing status files produced by the executive program (EXEC) and the Pipeline Control Programs (PCP's) and summarizing and displaying status information and diagnostics. The pipeline processing will be able to function independently of the pipeline monitor(s). The pipeline monitor program will also have the capability to display the execution status on each PCP operational (see sections 3.1 and 3.4), showing the target cpu, the scan being executed, and the start and stop time of the last scan executed. Additional summary information on the night's processing and the processing of each scan will be available, such as pipeline start time, source count summaries, etc.

On trapping an error, each system component will call a system-supplied error message routine with an explanatory message and an integer severity level from 1 to 8 (lower-numbered errors are more critical). The system error message routine will concatenate the error messages from the various system components into status files, which can be accessed by the pipeline monitor program(s). The pipeline monitor program will be able to display critical error messages (below a tuneable threshold) in order to get the immediate attention of the operator, who will normally be running a copy of the pipeline monitor program on a local console. The remaining messages will be concatenated in two output lists, one sorted by time, and one sorted by time within severity level. The first list will give errors in context. The second list will be in decreasing order of severity to allow fast detection of critical errors. The system error message routine will attach a time tag to each error message received. The error message text will identify the originating subsystem at the beginning of the message.

2.6 Subsystem Coding and Documentation Requirements

Subsystem coding shall be performed primarily in ANSI standard Fortran77 and C. Flint and Lint shall be used (for Fortran and C routines, respectively) to verify consistency of calling arguments and common blocks and to detect usage of variables not assigned values. Fortran routines shall be compiled with the -u option to require all variable names to be declared in type statements, in order to reduce the impact of typographical errors.

Subsystem cognizant engineers shall maintain documentation of the following types of information in formats to be determined by the design team.

1.) Requirements and Functional design (Subsystem Design Specification, SDS)

2.) Software interface specifications for output files and interfaces between subsystems

3.) Test environment and regression test baseline

4.) NAMELIST parameters (names, types, defaults, units, descriptions)

5.) Configuration file formats

6.) Intermediate file formats

7.) Logical I/O unit numbers and associated files

8.) Version history

9.) Lien list (if applicable)

10.) Risk list (if applicable)

Delivery of subsystem software to 2MAPPS shall include all relevant source code, include files, make files, and all scripts and data files needed for executing the subsystem in the system environment. Delivery and configuration control is discussed in section 2.8 below.

All subsystems shall write their version numbers as part of all log-file headers and other output files. All modifications to subsystem code shall be accompanied by incremented version numbers. Subsystems implemented as main programs shall write their name and version numbers to the standard output device during initialization; subsystems implemented as subroutines called by another subsystem shall provide callable entry points that write their names and version numbers to the standard output device, and these entry points shall be called by the main-program subsystem during its initialization. This logging of names and version numbers shall be executed even if initialization errors are detected that result in aborted execution.

Messages written to the standard output and standard error devices must begin with the subsystem name (after any monitor sentinel, if appropriate). In multi-line messages, each line must either begin with the subsystem name or else it must be indented to indicate that it is part of a message from the last indicated subsystem. This rule is intended to clarify which subsystem wrote a given message in interleaved message streams; it may be relaxed if there is no possibility of ambiguity regarding which subsystem wrote the message.

Fatal error messages shall be written to the standard error device so that the system monitor will pick them up. In Fortran, this is done by writing to unit zero. Fatal-error termination will include setting the status code to an appropriate value (to be established by the design team). In Fortran, termination with a status code of NCODE is accomplished by the statement CALL EXIT(NCODE).

Main-program subsystems shall verify that the correct number of command-line parameters have been passed. This is done in Fortran by testing the intrinsic function IARGC(). If the number of arguments is incorrect, the program shall terminate after printing its name and version number on the standard output device (in Fortran, unit 6) and after calling any applicable subroutine subsystems' entry points to invoke their name-version display function.

All read operations shall include appropriate END and ERR branches to code that handles these conditions. All read errors and unexpected END-OF-FILE events shall be reported on the standard error device, and execution may be aborted if appropriate (in which case a fatal error code shall be returned via EXIT, as discussed above). I/O errors in READ, WRITE, and OPEN statements can also be diagnosed via the IOSTAT parameter and handled appropriately.

Scripts delivered to execute the subsystem code shall avoid generating inappropriate error

messages, such as attempts to delete nonexistent files appropriate to a different environment.

To guarantee precision sufficient to support the accuracy requirements specified in the 2MASS Level 1 Science Objectives and Specifications document (see section 1.1, item 5), the following precision requirements are levied on the 2MAPPS subsystems. Position coordinates and their uncertainties shall be represented in disk files with a precision corresponding to at least 0.02 seconds of arc, and computations involving position coordinates and their uncertainties shall be done in double precision as needed to guarantee conservation of this precision. This implies that file formats representing position parameters must be accurate to three decimal places when represented in seconds of time, two decimal places when represented in seconds of arc or camera pixels, and six decimal places when represented in floating-point degrees. Photometric quantities and their uncertainties shall be represented with a minimum precision of 0.001 magnitudes, with similar requirements regarding computational precision needed to guarantee conservation of this precision and corresponding precision when other photometric representations are used.

2.7 Subsystem Testability Requirements

Each 2MAPPS subsystem must be testable in a manner that does not depend on the system-level end-to-end test environment. This is required for subsystem development. The 2MAPPS system design is optimized for processing efficiency at the expense of simplified serial processing flow. This makes it necessary to consider testability from the earliest stage of subsystem design. Each subsystem will need its own dedicated test environment that simulates the embedding of that subsystem in the integrated system, including any simulated data needed to activate the test environment.

In addition, each subsystem will provide the capability to output intermediate data (such as flattened frames from DFLAT) to allow inspection and to provide inputs to downstream processes so that each subsystem may be run in standalone mode. Subsystems will be run in standalone mode during subsystem testing, system integration, and debug and analysis.

2.8 System Delivery and Configuration Control

The 2MASS software delivery system contains three directory levels under the /2massc partition. The directories are:

1.) dev (development area): Software is written, compiled and tested by the developer here. Also, the regression test baseline (RTB) is created in this area.

2.) del (delivery area): Software is copied into this area, compiled and integrated with other software once the developer feels the dev version is stable and working properly. New tests may be run by the configuration manager and other developers or analysts.

3.) ops (operations area): Once the software is well tested and approved, it is installed into the operations area by the configuration manager, for use as production software. Only the

configuration manager has write access to the operations area.

All three areas will have the same structure. They contain the following subdirectories for installation of software:

bin (where executables are installed)
include (where include files are installed)
lib (where libraries are installed)
doc (where documentation is installed)
src (where source code is kept)

Programs are developed in /2massc/dev/src. For each program written, the programmer makes a subdirectory under /2massc/dev/src. A Makefile is used to compile, link and install the software in ../../bin.

No delivery system path names are hardcoded into either Makefiles or programs. The use of relative paths above allows for portability of the entire software delivery system with little or no impact on developers.

The following files are delivered to the system by the developer along with the source code:

1.) Change.Log (history of changes): Whenever a new version is delivered, update this file, so we know which version contains which modifications.

2.) Dependencies: Identifies all the build- and run-time dependencies.

3.) ToDo: Things that either need to be done in the future, or possible enhancements you can think of.

4.) Makefile: The commands that compile, link and install the software.

Every Makefile has the following targets:

1.) make clean (which removes all objects and binaries for this application)

2.) make all (which compiles/builds all components for this application)

3.) make install (which puts all user components in ../../bin, ../../lib, etc.)

When programs are ready for delivery to the system:

1.) The developer makes a directory in /2massc/del/src that is the name_version# of the deliverable, for example, scan_1.1.

2.) The developer then copies all the source code, the Makefile, Dependencies file,

Change_log file and To_do file for the software into the new directory.

3.) Next, the software is built and installed in the appropriate areas, as needed.

Once the installation in the delivery area is complete, the developer notifies the configuration manager, who iterates with testers and other developers to make sure all the needs of the delivery system, the software requirements, and science drivers are met. If the software has other software dependencies, or another program is dependent on it, the integration of the pieces is studied and tested carefully to make sure all the pieces behave normally (note that multi-subsystem processors such as PIXCAL/DFLAT/RDFRAME/FREXAS are integrated by the subsystem cognizant engineers and delivered to a single directory). The system test and integration process can involve redelivery of the software over and over until all the parties agree that the software is ready to go "operational". No changes are made in the delivery area. All changes are made back in the programmer's development area and the code is then redelivered.

Once it is agreed that the code should go into the operations area, the configuration manager creates a back up of the current operational version of the program, and copies the new version into /2massc/ops. (This is done at a predetermined time, when production is not running, and processing effects are understood.) It is then recompiled and installed one final time, and the RTB is run again. Operations staff is notified of the change.

3.0 Subsystem Requirements

The requirements on the subsystems are specified here to the level required to establish:

1) How the system design will be affected by subsystem components.

2) How each subsystem fits into the system.

3) What the inputs, functions, and outputs of each subsystem are.

The mapping of System Functional Requirements into each subsystem, and the detailed design of each subsystem, are contained in the Subsystem Design Documents.

Although specified as subsystem components, the EXEC, PCP, PIXCAL, and PIXPHOT subsystems are specified in greater detail, as they are integral parts of the system design.

3.1 EXEC (Execution Control Subsystem)

Function: To run the processing pipeline for one night's data from one observatory; to provide system-monitor capability

Called By: Data Management Team.

Input: NAMELIST and configuration files

Output: Processing summaries

Processing: (pipeline)

The processing of the data from one observatory for one observation reference date is controlled by the EXEC subsystem, which performs the following tasks after the TAPE- LOAD subsystem has loaded all relevant files and data validation has been completed:

1.) invokes the DARKS subsystem to create and analyze dark, mask, and responsivity maps;

2.) starts a prescribed number of PCPs (Pipeline Control Programs) in "calibration-scan-only" mode and waits for them to finish processing all the calibration scans, monitoring their progress; scan processing includes linearizing the data frames, computing flats, offsets, and the seeing model, extracting point sources and correcting them for saturation, reconstructing positions, band merging, coadding images, and extracting galaxies;

3.) invokes the CALMON subsystem to perform calibration processing;

4.) starts a prescribed number of PCPs in survey-scan-only mode and waits for them to finish processing all survey scans, monitoring their progress; scan processing includes everything mentioned in item (2);

5.) invokes the QUALITY subsystem to perform quality checking;

6.) performs tape archiving and summary processing;

7.) invokes the DBMAN subsystem to update the WSDB;

8.) updates survey scan logs and the survey execution data base.

Processing: (system monitor)

This subsystem is also responsible for providing the system monitor utility. This is a program that may be run at any time from any terminal connected to the 2MAPPS network. Its function is to provide system-wide status information that is updated at a user-selectable frequency (default: 10 seconds). This information is displayed on the monitor of the terminal from which the program was executed and includes: which scans are being processed by which PCPs, which CPUs are running which PCPs, how long each PCP has been working on its current scan, etc.; warning messages will be displayed when any PCP appears to be taking too long to process its current scan, and information from the system error log file will be relayed to this display. For further information on this function, see section 2.5.

3.2 TAPELOAD (Tape Loading to Disk)

Function: The TAPELOAD subsystem reads onto IPAC disk the entire tape set (one observing night's worth) of files (written at the observatory and shipped to IPAC) and creates back-up tape copies.

Called By: Data Management Team

This subsystem is a stand-alone software utility at the front end of 2MAPPS.

Input: One or more tape cassettes. Each tape may contain the following:

1) a tape label

2) a tape directory file of this tape

3) the tape set catalog file (a comprehensive catalog specifying every file contained in the total tape set). This file is contained only as file no. 2 on the final tape cassette.

4) scan files

5) dark sequences

6) calibration scans

Output:

1) a complete set of IPAC disk files for the observing night with each file loaded into its appropriate directory according to a predesigned tree structure.

2) back-up tape copies of each tape in the input tape set

3) periodic status messages indicating the progress of the tapeload

4) a final TAPELOAD completion message summarizing the loading status

Output: Dark frames, responsivity frames, scan files, data logs

Design Assumptions: The Observatory/IPAC interface specification (Ref. 1.1-3) will be adopted. Reference 1.1-3 has been used as the interface definition for the purposes of this document. This interface needs to be negotiated as soon as practical. Major changes in this interface could have major repercussions on the design.

Processing: By using a combination of Fortran programs and UNIX scripts, the tape cassettes will

be loaded from tape to IPAC disk. The TAPELOAD subsystem will verify that each physical cassette actually contains the proper data files as specified in that tape's directory. It will also verify that each file on that tape has been successfully loaded to IPAC disk. In addition, the catalog file will be used to ensure that the entire tape set has been loaded.

3.3 DARKS (Darks, Masks, Responsivity Processing)

Function: To analyze dark frame sequences and compute dark frames, masks, and responsivity frames

Called By: EXEC

Input: NAMELIST and configuration files, dark sequences, (TBD) twilight/dome flats

Output: Dark/bias frames, masks, responsivity frames, and processing summaries

Processing: The DARKS subsystem is invoked by the EXEC subsystem to compute dark frames, masks, and responsivity images for frame flattening. The DARKS subsystem will process all the dark frame sequences for a night and create the super-dark frames prior to processing any of the individual calibration or survey scans, so that the calculated super-dark frames will be available.

The DARKS subsystem will process each dark frame sequence (consisting of 100 or more frames for each of the three chips taken with the dark slide in place) as a unit. A trimmed-average dark frame, along with corresponding trimmed-average population sigmas, will be computed. Dark masks for pixels with anomalous dark values or dark sigmas will be created. Composite dark masks, histograms of the dark frames and dark sigma frames, and summaries of masked pixels and changes from previous darks will be output.

Dark frame sequences will be taken at the start and end of each observing night, and possibly at intervals during the night. The results from the dark sequences during a night will be monitored for consistency. An increase in masked pixels >TBD will be flagged to the run-time monitor as a high level warning.

The DARKS subsystem will process all twilight and/or dome flat data (TBD) to compute responsivity images, i.e., images of the pixel responsivity scale factors in each band. These will be used in DFLAT, where sky offsets are computed to go with the responsivity images, which then allows the raw frames to be flattened. Twilight and/or dome flats will be taken at TBD intervals. When no new data are available, the chronologically nearest responsivity images will be used.

3.4 PCP (Pipeline Control Processor)

Function: To process an individual scan

Called By: EXEC

Input: NAMELIST and configuration files

Output: Processing summaries

Processing: The PCP subsystem is multiply invoked by the EXEC subsystem to control the processing of individual scans, either in calibration-only or all-unprocessed scans mode. Typically, several PCPs run on each of several machines; the total number depends on the amount of data to be processed and the number of processors (i.e., CPUs) available. Each PCP's unit of work is a single scan, for which it processes the three bands in the order J, H, and then K_s. For each band, the (read2-read1) frames are processed first, then the read1 frames. After each PCP finishes the scan it has been working on, it will update the log file to show that it has finished that scan and then seek another scan not yet being processed by any other PCP; this is done by reading the nightly log file and searching for appropriate scan entries not identified as taken by other PCPs. When a PCP takes a scan to process, it updates the log file to show that the scan has been taken, identifying itself by an ID assigned to it by EXEC. When no appropriate unprocessed scans are found, the PCP terminates. For each scan, the PCP performs the following tasks:

1.) processes bands J, H, K_s in order; for each band, invokes the PIXCAL subsystem (this subsystem calls DFLAT to flatten the frames; DFLAT uses the RDFRAME intersubsystem function to obtain the frame data) and calls FREXAS to compute the seeing parameters, frame statistics, and point-source detections used later by POSFRM to compute frame offsets;

2.) invokes the POSFRM processor to reconstruct the frame positions;

3.) processes bands J, H, K_s in order; for each band, invokes PIXPHOT to coadd the frames to make images and to extract point sources.

4.) invokes the MAPCOR subsystem to merge Read1 photometry, flag artifacts (ghosts and persistence objects) and confused objects (too close to bright objects), and compute and apply aperture corrections and normalization to the photometry;

5.) invokes the BANDMERGE subsystem to combine extractions of the same point sources in different bands;

6.) invokes the POSMAN subsystem module POSPTS;

7.) invokes the GALWORKS subsystem;

8.) updates the scan-processing log file.

3.5 PIXCAL (Pixel Calibration Processing)

Function: To perform all pixel-oriented calibration for one band of one scan

Called By: PCP

Input: NAMELIST and configuration files

Output: Processing summaries

Processing: The PIXCAL subsystem is invoked for each band of each scan by the PCP subsystem to coordinate the processing of that band-scan combination by the DFLAT and FREXAS subsystems. DFLAT is called to compute, analyze, and apply flats, and FREXAS is called by DFLAT to compute the seeing model and other statistical parameters, and to extract sources to be used by POSFRM to compute frame offsets.

3.6 PIXPHOT (Pixel Photometric Processing)

Function: To perform all pixel-oriented photometric work for one band of one scan

Called By: PCP

Input: NAMELIST and configuration files; all (Read2-Read1) data frames prepared by DFLAT are read from disk.

Output: Processing summaries

Processing: The PIXPHOT subsystem is invoked for each band of each scan by the PCP subsystem to coordinate the creation of coadded images by the PICMAN subsystem, the detection of point source candidates, the estimation of seeing as a function of time by the intersubsystem function SEEMAN, and the estimation of point source parameters by the PROPHOT subsystems.

3.7 DFLAT (Frame Flattening)

Function: To compute sky offsets and to apply them to the frame data.

Called By: PIXCAL

Input:

1.) NAMELIST and configuration files

2.) Frame data

Output:

1.) Calibrated frame data (passed to PICMAN and PROPHOT)

2.) Processing summary

Processing:The DFLAT subsystem is called by the PIXCAL subsystem to compute the sky offsets for the scan and to apply them, along with the responsivity scale-factor images provided by the DARKS subsystem, to the scan's Read1 and (Read2-Read1) frames. DFLAT performs the following tasks:

1.) initializes itself, calls the RDFRAME intersubsystem function (see Appendix E) to provide the number of frames in the scan, and calls FREXAS for initialization processing;

2.) computes the central-frame numbers for each stack to be used for sky offset computation;

3.) reads the responsivity image (e.g., from twilight flat analysis).

4.) reads the mask and the superdark;

5.) loops over the number of sky offset frames to be computed; for each one, the following tasks are performed, where references to pixels apply only to pixels not turned off in the mask:

a.) RDFRAME is called to supply the (Read2-Read1) frame for each frame in the stack;

b.) each frame in the stack is linearized;

c.) the superdark provided by the DARKS subsystem is subtracted from each frame in the stack, and the result is divided by the responsivity image;

d.) the stack median for each pixel is computed and subtracted from the corresponding pixel in each frame in the stack; then the trimmed average of this result for each pixel over the stack is computed; the result is called the “sky offset frame” or “skoff”;

e.) statistical tests are applied to the pixels during the sky offset computation, and pixels with out-of-limits behavior are blanked in that sky offset frame;

6.) linear fits to each pixel over all sky offset frames are computed and statistical tests over all sky offsets are performed; pixels with out-of-limits behavior are turned off in the mask, and a new mask is written out;

7.) when called in survey-only mode (after calibration scans have all been processed), DFLAT obtains from the scan parameters file the scan numbers of the calibration scans most closely bracketing in time the survey scan being processed; the sky offsets from these calibration scans are read and compared to those computed for the survey scan, and tests are made to determine whether the survey-scan skoffs are contaminated by high source densities; if so, a sky offset frame is interpolated from the calibration scans and used instead.

8.) For each (Read2-Read1) and Read1 frame in the scan, the frame is obtained again from RDFRAME, and for each pixel in the frame, the following tasks are performed:

a.) for (Read2-Read1) frames, the frame is linearized;

b.) the superdark is subtracted;

c.) the sky offsets bracketing the frame are used to interpolate a sky offset value; if either sky offset has that pixel blanked, the linear fit is evaluated for the current frame;

d.) the pixel is divided by the corresponding responsivity-image pixel, and the corresponding sky offset pixel is subtracted from the result; this yields the calibrated pixel;

9.) the frame is passed to the FREXAS subsystem;

10.) for (Read2-Read1) frames, the frame is written to disk;

After all frames have been processed, DFLAT calls FREXAS for termination processing and prints a processing summary.

3.8 FREXAS (Frame Extraction, Seeing, Statistics)

Function: The FREXAS subsystem computes statistical parameters (background and noise) for each flattened frame, extracts source amplitudes and positions from each frame, and computes an enpixelated-energy parameter during each scan to quantify the effect of seeing on the psf.

Called By: DFLAT

Input:

1) Flattened, calibrated frames.

2) Frame header in FITS format.

Output:

1) Detected source positions and amplitudes from each frame.

2) Enpixelated energy parameter.

3) Summary file containing frame statistics for each frame.

Processing: The FREXAS subsystem is passed one frame at a time by DFLAT for the current band. The frames are passed sequentially from the first frame in the scan to the last. FREXAS computes statistical parameters such as the median background level and the estimated noise for each frame, using a histogram. The airmass parameter in each frame header is also written out to the frame statistics file for use by CALMON. In addition, FREXAS extracts the sources from each frame and estimates their frame coordinates and outputs them to a file. FREXAS will also estimate the enpixelated energy (a measure of the seeing) as a function of time.

3.9 MAPCOR (Merge And Purge, Aperture Corrections)

Function: To merge the Read1 photometry for saturated point sources with the Read2-Read1 photometry; to "purge" ghosts, diffraction spikes, persistence sources, and confused sources near bright point sources; to compute and apply aperture corrections to the aperture and PSF-fit photometry.

Called By: PCP

Input:

1) NAMELIST and configuration files

2) Read2-Read1 source file from PROPHOT, with both PSF-fit photometry and aperture photometry using a sequence of apertures

3) Read1 source file from POSFRM

4) Seeing history during the scan from FREXAS

Output:

1) Read1 and Read2-Read1 merged source extraction list with aperture-corrected PSF-fit photometry and "purge" flags indicating sources that are ghosts, diffraction spikes, persistence, or confused

2) Processing summary file(s) containing processing parameters and statistics and calculated aperture correction data

Design Assumptions: It is assumed that enough camera pre-survey test data will be processed to accurately determine empirically the many namelist-provided parameters used by MAPCOR, such as position and magnitude search windows and magnitude limits. These parameters would then be checked periodically during the survey.

MAPCOR needs seeing information from FREXAS, determined preferably at several intervals during each scan. This would highlight significant seeing changes, and possibly help determine source confusion radii.

Read2-Read1 data from PROPHOT should include both PSF-fit and aperture photometry data for each source, with apertures ranging from 2 arcsec to 10 arcsec (or TBD). The aperture range needs to be adequately sampled and large enough for accurate curve-of-growth and aperture correction determination. At present, using aperture radii between 2" and 10", in steps of 1", seems ideal.

Processing: MAPCOR will read in the Read1 and Read2-Read1 source files from FREXAS and PROPHOT respectively, and search each source list for ghosts, diffraction spikes, persistence

sources, and confused sources near bright point sources. Then, using aperture photometry data for Read2-Read1 sources and perhaps the seeing history during the scan from FREXAS, it will calculate the aperture correction to an "infinite" aperture magnitude and the PSF photometry normalization, and apply these corrections to the aperture and PSF-fit photometry. It will then merge the Read1 and Read2-Read1 source data lists and write the resulting source data to a disk file.

Purging Bright Star Artifacts

MAPCOR will search the source lists for artifacts created by each source brighter than a namelist-provided limit. Ghosts or filter glints, presently found as point sources only in the J-band data, are located near the bright source at a very predictable position and magnitude offset. Diffraction spikes have a predictable shape and size which depends on the magnitude of the bright source. Any extracted sources found within the predicted spike area will be considered pieces of the diffraction spike. Positions at an integral number of frames after the bright source are searched for persistence sources which are progressively fainter than the bright source. The source lists are also searched for other sources found within the magnitude-dependent confusion radius of the bright source. In all cases, position and magnitude search windows are extended to take noise and errors into account. The bright source confusion radius might also be increased when seeing worsens (TBD).

The source lists are "purged" by setting various flags for each source determined to be an artifact or confused source. The actual entry in the source list will not be deleted, but the flagged sources will be ignored for all subsequent MAPCOR processing, and for much of the downstream subsystem processing as well.

Aperture Corrections

Aperture corrections to an "infinite" aperture magnitude are calculated using the aperture photometry data for valid (non-purged) sources in the Read2-Read1 source list. This data will consist of magnitudes using apertures ranging in radius from 2 arcsec to 10 arcsec (or TBD). A seeing estimator calculated at several intervals during each scan would indicate whether seeing conditions changed considerably during the scan. If so, aperture corrections could be calculated separately for different scan segments if the point source density is over a certain threshold (TBD).

MAPCOR calculates the magnitude curve-of-growth using the sequence of aperture magnitudes. The "infinite" aperture is the aperture at which the curve converges within a namelist-provided limit. If instead the curve begins to diverge, indicating a bad case of confusion, the aperture just before the point of divergence is taken as the "infinite" aperture. If the curve neither converges nor diverges, probably indicating very poor seeing conditions, the maximum allowable aperture (presently 10 arcsec radius) is taken as the "infinite" aperture. MAPCOR then calculates the aperture correction from the standard aperture to the "infinite" aperture and applies the aperture correction to all of the valid Read1 and Read2-Read1 sources' standard aperture magnitudes.

The PSF-fit magnitudes must now be normalized to the aperture corrected aperture magnitudes, because the PSFs are not perfectly determined for various seeing conditions. MAPCOR calculates the median of the offset between each valid Read2-Read1 sources' aperture corrected aperture magnitudes and PSF-fit magnitudes, and applies it to the PSF-fit magnitudes to produce aperture corrected PSF-fit magnitudes.

If a stable and accurate seeing estimator is available, it should be possible to create a look-up table of aperture corrections corresponding to different values of this estimator. MAPCOR will have the option, to be set in the namelist, of using this table instead of calculating the aperture corrections for each scan segment.

Read1 and Read2-Read1 Merge

The Read1 sources, in order of brightest to faintest, are now merged into the Read2-Read1 source list by matching positions and magnitudes within namelist-provided search limits. If the Read1 magnitude is brighter than one namelist-provided threshold, the matching Read2-Read1 source's magnitude data is replaced by the Read1 magnitude data and one flag is set; if it is also brighter than another threshold, both the matching source's position and magnitude data are replaced by the Read1 data and another flag is set. If no Read2-Read1 source matching the Read1 source is found, the Read1 source is added to the Read2-Read1 source list. To prevent duplicate ID numbers from appearing in the merged list, the ID numbers given to Read1 sources by PROPHOT should be distinctive; for example, they could all be greater than 800000.

Finally, MAPCOR will write the merged source list (including the purged sources) to a disk file which will be read by BANDMERGE. It will also write processing summary file(s) containing processing parameters and statistics such as numbers of various types of purged sources, aperture magnitude curve-of-growth data in a format that can be used for plotting, aperture correction and normalization statistics, and merged source statistics.

Messages are sent to the monitor program and written to the processing summary file(s) whenever possible processing and/or data problems are found, including (but not limited to) empty source files, a large number of confused sources, small number statistics, lack of curve-of-growth convergence, or confusion during the Read2-Read1 and Read1 merge.

3.10 PICMAN (Image Generation)

Function: The PICMAN subsystem registers and combines the overlapping flattened, calibrated frames from a single scan into a set of coadded images for that scan. The PICMAN subsystem creates a separate set of images for each band.

PICMAN will also detect sources in the coadded image and create a list of candidate source detections to be read by PROPHOT.

Called By: PIXPHOT

Input:

1) Flattened, calibrated frames.

2) Cumulative frame offsets.

3) Frame header in FITS format.

4) List of bright objects in each frame (from FREXAS).

5) Models for persistence and internal reflection removal.

Output:

1) FITS formatted coadded images and counts.

2) Candidate source detections (to be read by PROPHOT)

3) Summary files.

Processing: The PICMAN subsystem is passed one frame at a time by PIXPHOT for the current band. The frames are passed sequentially from the first frame in the scan to the last. PICMAN registers the frames according to the pixel displacements computed by the POSFRM subsystem, interpolates them to a common coordinate system, and coadds the interpolated data into the output images. PICMAN will create a set of output images one camera frame wide by ~2 camera frames long. This set of images will cover that part of the sky overlapped by at least 6 frames in the current scan.

PICMAN creates the coadded images for the scan sequentially, combining all frames passed by PIXPHOT overlapping the current image, writing that image to disk in FITS format along with an image containing the counts for each pixel, and then continuing on with the next image. The coadded images for all three bands will register identically on the sky, having a common pixel size, and a common centering and twist orientation. Each image will orthographically project the sky with a projection tangent point at the center of the image. The information from each input frame will be interpolated to the image pixels and coadded with the interpolated information from overlapping frames. The nominal pixel size will be 1/2 the camera pixel linear dimensions, giving approximately 4 coadded image pixels per camera pixel.

PICMAN will be passed each frame sequentially by PIXPHOT along with the cumulative offset in in-scan and cross-scan for the first pixel of that frame relative to the first pixel of the first frame. PICMAN will also read a list of bright objects in each frame, and mark pixels affected by persistence or internal optical reflections as dead prior to coadding. The affected pixels will have to be defined as a function of source brightness in models input to PICMAN. The bright source cleanup capabilities will be optional under control of flags in namelist input.

PICMAN will also pass each new panel for which all data has been coadded to a source detection subroutine, which will detect candidate sources and create a detections list to be passed to PROPHOT. PROPHOT will subsequently read the detections list, separate confused sources, and optimally estimate the parameters of each source (see Section 3.11).

PICMAN will also output an ascii summary file containing the following for each coadded image:

- the cumulative in-scan and cross-scan offset of the first pixel in the image relative to the first pixel of the first flattened frame.

- the right ascension and declination of the image center, and the twist angle relative to J2000.

- the median background level and estimated noise for that image.

PICMAN will receive a scan termination call from PIXPHOT at the end of the processing for each band of each scan, at which time it will complete and output the last image for that band, and the summary file for that band.

Methods of generating and maintaining an on-line image archive of images will be investigated. A lossy compression technique will be required for this to be feasible.

3.11 PROPHOT (PSF-Fit Photometry)

Function: The PROPHOT subsystem is responsible for the extraction and characterization of point sources in a complete scan.

Called By: PIXPHOT

Input:

1) Flattened data frames (Read2-Read1) generated by DFLAT and passed by PIXPHOT

2) Frame offset list generated by POSFRM and passed by PIXPHOT

3) Source detections list generated by PICMAN and passed by PIXPHOT

Output:

1) Source extraction list, consisting of:

    a) unique ID numbers for point sources in the scan,

    b) refined X,Y coordinates in the scan system,

    c) positional uncertainties,

    d) point-spread-function fit magnitudes,

    e) flag for blended sources, and

    f) error estimates and diagnostics of goodness of fit.

Design Assumptions: The subsystem requires several pieces of information. First, a fiducial

measurement of the optical point spread function (PSF) must be determined before the start of the survey, and thereafter whenever the camera setup is significantly modified. A PSF must be defined in each band as a function of position with the chip. All fiducial PSFs must be made under similar, very good seeing conditions.

To produce a PSF appropriate to the scan being processed, a measurement must be made of the broadening of the PSF due to seeing and focus changes. The determination of broadening may be very hard to do on timescales shorter than "normal"(6-degree) scan durations (too few high S/N stars to subdivide in a scan), and may not be representative of a particular scan if done less frequently than once every 15 minutes.

The PROPHOT subsystem uses an input source list from which it gets the initial centers of all point sources. The routine also requires fairly accurate knowledge of the frame-to-frame offsets. This will insure a good starting position for PROPHOT source fitting. The relative position uncertainties in fitting sources will be passed on to POSPTS to combine into the absolute position uncertainties.

Processing: PROPHOT photometers all stars on the Read2-Read1 frames via PSF fitting. This is necessary because of the crowding expected in the lower galactic latitudes and other confused regions, and it provides higher S/N for faint sources than aperture photometry. Single processed frames are passed to the routine by PIXPHOT. Once PROPHOT has data from six overlapping regions, the routine produces a set of photometry, both PSF-fit and aperture, for all sources whose centers lie in the common panel, as sampled by overlapping subsections of six frames. The brightest stars in that panel may be saturated.

The aperture photometry will be used in MAPCOR to compute aperture corrections to the PSF fit measurements.

The PSF fitting code in PROPHOT takes the six samples of the star and converges, using a chi-square minimization technique, to final estimates of the X,Y position of the object and the height of the light distribution above a background, relative to a fiducial PSF. The individual errors in the position and the brightness are output for each source, along with diagnostics of the goodness-of-fit.

When the goodness-of-fit parameters indicate that a poor fit may be due to multiple sources, an additional algorithm is used to attempt to determine if a blend of PSFs would better fit the detected source. A flag indicating the blended nature of the object is set.

Another flag is used when a very bright (well-saturated) star is detected. The resultant persistence features that may be detected are easily identifiable and are flagged as such in the output list. This information will be used, for example by GALWORKS, to treat the persistence detections as spurious sources.

Proper PSF fitting of stars near the in-scan edge of a particular subsection may require data slightly outside the strict overlap region. Thus, all six frames need to be retained in their entirety, and an additional frame should be kept to provide data outside the edges of the end subsections.

Point sources very near the edge of frames in the cross-scan direction (within about six pixels of the edge of the frame with the closest border) are not photometered, since they are well-sampled in the adjacent scan. This will avoid problems with measurements of sources whose PSF is truncated, possible confusion in identifying blended sources, and interference with parameters designed to indicate how much the source is like a PSF.

PROPHOT will only attempt to measure magnitudes for sources that lie on all six frames. Thus,

at the beginning and end of each scan, PROPHOT will not measure sources on pieces of five frames. There will be sufficient overlap in the survey that this restriction will not create gaps in coverage and will insure continuity in the quality of data.

The diagnostics of goodness of fit for each source will include two parameters, SCALE and SHARP, that indicate how closely the source approximates the PSF used, in both scale and sharpness. Deviations from unity in these numbers should occur for non-point-source objects. Thus, SCALE and SHARP will be passed to GALWORKS to help sort out such sources.

3.12 BANDMERGE (Point-Source Bandmerging)

Function: To merge the source lists produced by the MAPCOR subsystem for the separate bands of each scan into a single bandmerged source list.

Called By: PCP

Input: BANDMERGE reads source lists produced by MAPCOR and calibration parameter values produced by CALMON for the three bands of a single scan.

Output: BANDMERGE produces a band-merged calibrated source list for the scan, including confusion status information and refined position parameters when high-quality sources are merged; a processing summary is also produced that provides statistical information including position discrepancy behavior and band-combination counters.

Processing: The BANDMERGE subsystem is called once per scan by the PCP subsystem to merge the source lists produced for separate bands by the MAPCOR subsystem. A signal-to-noise ratio boundary will separate “high-quality” and “low-quality” point source extractions. Only high-quality sources may contribute position information. Position refinement will be performed after all merging decisions have been made. The photometric transformations computed by CALMON will be applied to the output point sources and upper limits.

Design Assumptions: The sources are processed as “seeds” for merging in decreasing order of S/N. The position errors are assumed to be distributed as circular gaussian random variables.

3.13 POSMAN (Position Reconstruction)

Function: The POSMAN subsystem maps the data frames and point source extractions into equatorial coordinates and supplies associated uncertainties. The subsystem consists of two main modules. The first of these is POSFRM, which maps the frames to inertial coordinates using single-frame extractions from FREXAS and astrometric positions from the Tycho catalog. The second module is POSPTS, which refines the position reconstruction using the multi-frame point-source extractions from BANDMERGE and attaches the final positions to point sources.

3.13.1 POSFRM

Called By: PCP

Input:

1) Files of single-frame point source extractions from FREXAS, including positions and relative position uncertainties in the XY-frame coordinate system. Relative uncertainties will include effects due to point source extraction errors. Files of Read1 and Read2-Read1 extractions for all three bands will be included. All will be combined to provide the most possible points for reconstructing the frame positions. A set of Read1 scan coordinates for use in MAPCOR will be determined using the computed frame geometry.

2) File of astrometric reference sources. This will consist of the highly accurate Tycho catalog sources from the Hipparcos mission. Sources from a very high-density catalog such as the UJ or GSC will also be used to confirm the pattern matching, but will not be used for position determination. The confirming file will be referred to as the High Density Low Precision Star Catalog (HDLPSC).

3) NAMELIST file to control POSFRM and POSPTS

4) Estimate of initial position from telescope

5) File of distortion model parameters (one set of parameters per band)

Output:

1) Files (one per band) of frame-to-frame offsets and reconstructed frame centers in equatorial coordinates

2) Files (one per band) of Read1 extractions in XY-scan coordinates

3) Frame extractions confirmed and band-merged into a single XY-scan file with Tycho match associations.

4) File of POSFRM common block data passed to POSPTS

5) File of tracking and error messages

6) File of residual position differences for quality monitoring

Design Assumptions:

1) The initial position estimate from the telescope will be within 10-15 arcseconds (true angle) of the actual position.

2) Changes to the telescope and/or camera which could affect the mapping of point sources onto the three chips will be rare and IPAC will be notified of any such change prior to processing the affected scans. These include the following:

a) changes to the optical configuration (including recollimation)

b) changes in the orientation of the three chips with respect to the telescope

c) dewar warm up

3) The telescope will be designed to minimize focus shift due to changes in the ambient temperature. Any refocusing required will always be done in a consistent manner.

Processing: The POSFRM module is invoked once for each scan to do the following operations:

1) Read in single-frame point source extractions (Read1 and Read2-Read1) generated by FREXAS for all three bands.

2) Replace frame positions of saturated Read2-Read1 sources with positions derived from Read1

3) Remove distortion in each frame using band-dependent models.

4) Confirm sources showing up in multiple frames of the same band and merge confirmed sources showing up in more than one band.

5) Generate a set of confirmed/merged point sources in XY-scan coordinates

6) Pull out reference sources covering the expected field (plus uncertainties) and convert them into a precisely defined astrometric XY-scan system with the Y-axis aligned along the nominal scan path.

7) Match the merged/confirmed frame extractions to the Tycho sources in XY-scan coordinates using a “two-peg pattern matcher”. Confirm that the correct overall match has been achieved by monitoring the HDLPSC position difference residuals. Although the HDLPSC residuals can be expected to reflect systematic errors on the order of a couple arcseconds, the scatter about the systematic difference history should be quite small for a correct overall match.

8) Simultaneously solve for all frame positions to minimize the frame-to-frame overlap point source differences and the differences with the Tycho reference stars.

3.13.2 POSPTS

Called By: PCP

Input:

1) File of confirmed/merged single-frame extractions with Tycho matches from POSFRM

2) File of common block data passed from POSFRM

3) File of band-merged point source extractions from BANDMERGE

4) File of high-accuracy Tycho star positions and magnitudes from Hipparcos mission

Output:

1) File of reconstructed point source positions in equatorial coordinates with uncertainties

2) File of residual position differences with respect to Tycho

3) File of tracking and error messages

4) Summary file of fit parameters and quality indicators

In addition to the fit parameters this ASCII file will include:
- number of matched points used; number of rejects; number of refits
- percent of scan without matches (start, mid, end)
- population sigmas on X and Y fit residuals
- maximum cross-scan position deviation from linear
- maximum in-scan rate deviation from linear

Processing: The POSPTS module is invoked once for each scan to do the following operations:

1) Read in extracted point sources generated by BANDMERGE in XY-scan system. Sources may be composed of extractions in one, two, or three bands.

2) Use match results from POSFRM to match extracted point source extractions with Tycho stars in order to generate position-error data.

3) Perform a "linear" fit of position errors by rotating, then both translating and linearly stretching the XY-scan system associated with the extractions to give the best registration with the matching Tycho reference sources. Use inverse-variance weighting with the weights reflecting the relative position uncertainty associated with each of the possible band-merge combinations.

4) Determine uncertainties associated with the fit as a function of Y position as well as band-merge combination.

5) Use the fit results to map the position of each point source extraction into the astrometric XY-scan system, and from there into J2000 equatorial coordinates. Output the position and uncertainty associated with each of the extracted point sources. Output fit parameters file which includes all fit model constants needed to convert quickly back and forth between merged XY-scan coordinates and J2000 equatorial coordinates.

3.14 GALWORKS (Extended Source Processing)

Function: GALWORKS detects and estimates parameters for probable galaxies and other extended objects that are smaller than 0.8 arcmin, and outputs an image database containing the coadd image of these objects as well as coadd images of (pieces of) larger galaxies.

Input:

1) coadd images for all 3 bands in a given scan (& supercoadd; TBD)

2) catalogs for all 3 bands from PROPHOT

3) large galaxy catalog and other large extended objects

4) seeing history for entire scan containing the coadd frames

5) Calibration information to calculate magnitudes and upper limits

Output:

1) Extended source list, consisting of:

a) unique ID numbers for each source in the scan

b) band-merged equatorial position with uncertainty for each source

c) integrated magnitude in each band with uncertainty or upper limits

d) probability that the source is extended

e) central surface brightness with uncertainty

f) size and shape parameters with uncertainties

g) flag if source is also detected as a point source (TBD)

2) Low Surface Brightness Object list, with the same parameters as the Extended Source list.

3) Extended Source Image Database

4) Band-filled Point Source information

Design Assumptions: Galaxy extraction requires circular PSFs throughout the coadd that do not vary rapidly along the scan. Data showing rapid variations in seeing that are not tracked will be rejected. The input PSF FWHM will be used as a parameter to establish the zero-point for the relations used to distinguish stars from galaxies, such as the central surface brightness and size.

Areas around bright stars will not be searched for galaxies. Bright stars off the edge of the scan will cause problems, but GALWORKS can not reliably find these stars until they have been observed by 2MASS. Thus DBMAN will have to periodically mark sources in the extended source database that are found to be near newly-observed bright stars.

How to treat galaxies with multiple nuclei, or that overlap other galaxies, or that have stars superimposed on them, are issues TBD.

Processing: This subsystem performs four primary tasks: object detection, parameterization, classification and galaxy image extraction. These tasks are optimized to identify and parameterize galaxies from the coadd images. Before these tasks are performed, the coadds are first "cleaned" of bright stars ($K_s < 7$) and large (> 2 arcmin) galaxies. (The coadd images of these bright galaxies are saved in the Extended Source Image Database.) A secondary task is also performed: the band-merged point source list is band-filled, i.e., in bands where no extracted point source is available, upper limits are obtained from the coadded image, and the band-merged band-filled point source list is written out.

GALWORKS processes coadds from each band separately, and then combines the detections from multiple bands to produced refined parameters for each galaxy. The coadds are then consulted for each source to produce consistent integrated magnitudes and upper limits, including color terms if desired.

For bright stars, cleaning consists of masking pixels in circular regions centered on these objects, as well as horizontal, vertical, and diagonal rectangular regions containing diffraction spikes from these bright objects. The positions and magnitudes of bright stars are obtained from PROPHOT output.

For large galaxies, cleaning uses a catalog containing positions, magnitudes, and sizes of large optical galaxies. The catalog was generated from a "NED" search of the UGC catalog. Pixel masking is performed for regions defined by the size of the galaxy as given in the catalog. In addition, other

large extended objects (e.g., H II regions) are blanked from the coadds.

With bright stars and large galaxies masked from the coadds, the coadds are then searched for extended sources. Object detection is necessary since the PICMAN subsystem is optimized to detect point sources, and thus may be incomplete for galaxies and other low surface brightness objects. The most critical operation for galaxy detection and parameterization is the correct determination of the coadd background, which is subtracted from the coadd. A 2-dimensional background is fit across the entire coadd. (See galworks subsystem design notes for algorithm used.)

Two different algorithms are used to find galaxies. The first algorithm operates directly on the coadd, and finds "normal" galaxies that have a central peak greater than 3 sigma in a single coadd pixel. The second algorithm operates on the coadd after all point sources and galaxies found by the first algorithm are masked off. It then "averages up" the masked-off coadd to find low surface brightness galaxies (and other extended objects) not detected by the first algorithm (if they exist).

Algorithm 1

The purpose of the detection step for algorithm 1 is to find "normal" galaxies that were not detected as point sources by PICMAN. Thus all stars found by PROPHOT are masked off using a radius large enough to ensure that no flux remains in the coadd from them. Object detection consists of a two step process. The first step is to find local maxima objects. In the second step, all local maxima objects are compared to the "local" background value, which is computed in an annulus centered on the object. If the maximum pixel value of the object is less than 3-sigma above the "local" background value, then the object is rejected. This operation is designed to eliminated noise spikes and other false detections that tend to occur along steep flux gradients associated with moderate to large galaxies.

The next level of processing is to parameterize each detected object (including point sources from PROPHOT) and perform a set of classification operations that will distinguish stars from galaxies. The key parametric values computed for each object are the magnitude, central surface brightness, elliptical fit to the 3-sigma isophot, and the source extent: generalized galaxy profile shape, wedge-masked galaxy profile shape, major-axis profile shape, and the elliptical profile shape. Neighboring detections are masked out during the elliptical and profile fits. (See galworks subsystem design notes.)

The magnitude is determined via aperture photometry using several fixed size apertures, as well as a total magnitude derived from an ellipse fitted to the galaxy, and elliptical isophotal magnitudes (20 and 21 mag per sq. arcsec).

Galaxies and other extended sources are then selected from these objects by applying thresholds in central surface brightness and in the galaxy profile shapes as a function of magnitude. These thresholds apply to the quantitative distance these parametric values have from ridgelines defined by pure stellar sources. The ridgelines can be fully described by the FWHM of point sources from PROPHOT and from numerical simulations, as well as from parameterization of real stars found from PROPHOT. Double stars and some other false sources are rejected with the wedge-masked and major-axis galaxy profile shapes. Other artifacts, including geometrically well-defined "ghosts" and persistence are also rejected. A score is assigned to each object that represents the probability of that object being a galaxy.

Algorithm 2

The purpose of algorithm 2 is to recover low-surface brightness galaxies that might have been missed by algorithm 1. All stars found by PROPHOT are masked off using a radius large enough to ensure that no flux remains in the coadd from them. All galaxies found by algorithm 1 are masked off similarly. A filter is then applied to the masked-off coadd that detects all sources with integrated flux above a 10 sigma threshold. Another filter will derive circular aperture magnitudes for various radii for all detections.

The sources found by algorithm 2 will be kept in a separate file, as it is anticipated that the reliability of such detections will be low.

After the source lists (from each algorithm) are band-merged, the refined position and source shape are used to consult the coadds to produce an integrated magnitude or upper limit in each band.

Sources found from algorithm 1 will go into the main galaxy database, which will include the following information: position (RA, DEC), central surface brightness, circular aperture magnitudes (radii = 6, 10 and 15 arcsec), optimized elliptical aperture magnitude (the "integrated magnitude"), elliptical shape parameters (eccentricity, semi-major axis, rotation angle), mean background value and sigma, and profile shape parameters (alpha, beta). If it is desired, the determinations of the position, source size and shapes, and integrated magnitudes, as determined for each band individually, will also be output.

Sources found from algorithm 2 will go into a separate low surface brightness galaxy database, which will include only the following information: position (RA,DEC) and circular aperture magnitudes (radii = 6, 10, 15, ... TBD pixels).

The final task is to build the extended source image archive. For objects classified as extended sources, or known to be large galaxies from the optical input catalog, regions surrounding the objects are extracted from the coadds and stored to disk. These subimages have a size such that at least 95% of the flux of the galaxy is contained in the image.

3.15 CALMON (Photometric Calibration)

Function: The CALMON subsystem is responsible for computing and applying the photometric calibrations for one night. It will also maintain a log of zeropoints, color terms, and other pertinent information to trace photometric stability, in magnitude and color, over the whole survey.

Called By: EXEC

Input: The initial entry to CALMON runs the calibration production and requires:

1) Band-merged point source extraction lists for "calibration" PCPs, and

2) Standard Star Calibration Database (SSCD) consisting of:

a) unique IDs and star names,

b) Right Ascensions and Declinations (epoch = 2000.0), and

c) columns of magnitudes and associated accuracies.

The second entry to CALMON will apply the previously-determined calibration to all of the point source photometry for the night, requiring:

3) All of the band-merged source listing for each scan made that night.

Output:

1) Table of photometric calibrations for the night, with an estimate of the absolute calibration uncertainty; also included are diagnostics of the extinction and color transformations, and

2) Band-merged calibrated point source magnitudes for each scan.

Design Assumptions: This routine requires a standard star calibration database (SSCD). This is a list of standard stars, their Right Ascensions and Declinations, and magnitudes in the three colors with errors. All of the standards need to be on the same photometric system, and should be spaced at intervals for ease of observation and to provide variety of airmasses/magnitude/color. Because standards will be taken as two-degree scans, there will be stars in addition to the "primary" standards in the field. These secondary standards will be most useful in providing additional photometric measurements to help monitor the photometric behavior and extinction corrections for the night. The potential spread in color of these stars will aid in understanding and tracking color terms in the calibration. Seeing and other PSF information may also be derived from the secondary standards.

The sparseness of the standard fields, and the precision of the position-reconstructed coordinates, should be sufficient to easily match detections with standards. To match up stars in the SSCD with stars in the calibrations scans, the positions of stars in the scan source list should be known to better than 2 arcseconds.

Before the full survey begins, an initial run of calibration data is required. We need well determined transformations (esp. color terms) between the instrumental and standard systems. A large contiguous set of photometric data would also help determine the local extinction coefficients. A run of 7 - 8 days, dedicated to the photometric and PSF calibration of the instrument, should be sufficient. The results from this run would be stable color and extinction coefficients. At regular intervals (every two months), the extinction and color terms should be checked (with existing survey data) and possibly changed if significantly different.

Processing: The first call to CALMON produces the calibrations necessary to transform the point source measurements onto a standard system. The positions and extents of all calibration scans are checked against the SSCD for any and all coincident standard stars. The magnitudes of the identified stars are culled from the corresponding list of point source extractions.

Calibration observations must be taken roughly every two hours. Given the previously-set color and extinction terms, magnitude offsets for each band are calculated at those times during the night.

The magnitude offset for a particular band are examined as a function of time during the night. If large variations in any offset are seen, the data on that night are flagged as non-photometric. Smaller, longer-term variations in the offsets are corrected for by using a least-squares fit to the change in offset relative to the mean magnitude offset (a "night correction"). Thus for a particular time of the night, an appropriate magnitude offset can be calculated.

The final step for CALMON is the calibration of all of the band-merged survey source listings produced by PROPHOT. Transformations using the magnitude offsets, and appropriate color and extinction terms, will be applied. Provisional calibrations will be made for objects matched with fewer than three bands. The contribution of errors in the calibration process will be calculated and included in the final measurement error of each magnitude.

One part of the CALMON output consists of a calibrated source list for each scan in the night. A secondary output, the calibration log of transformations between the PROPHOT magnitudes and the standard star system, will be used to determine whether the night was photometric or not, and to track photometric behavior over long periods during the survey.

3.16 QUALITY (Processing Quality Checking)

Function: The QUALITY subsystem calls the CONMAN function to perform scan-to-scan point-source confirmation (see Appendix F), evaluates and summarizes the quality of each scan and assigns a quality grade to each scan for use by the Survey Planning software. QUALITY will access summary files and data output by other subsystems, including the statistics output by CONMAN.

QUALITY will also condense the summary information from the rest of the 2MAPPS system in a hierarchal fashion to facilitate the work of the cognizant scientist/engineers verifying the quality of each observing night's data reduction.

Called By: EXEC

Input:

1) Summary files output by other programs in 2MAPPS

2) Data files output by other programs in 2MAPPS

Output:

1) Scan quality summary files for use in Survey planning.

2) Processing quality abstractions and plots.

Processing: The QUALITY subsystem will consist of a master script which will run a collection of scripts and programs which will evaluate and summarize information from the other subsystems in 2MAPPS.

The primary job of QUALITY will be to evaluate the quality of each scan and assign a grade for use by the Survey Planning software. The main inputs to this process will be the outputs of CALMON, POSMAN, and CONMAN. The goal will be to automate this process, but facilities will be included to override this decision process by the cognizant scientist/engineer. The default operational mode will be for the cognizant scientist/engineer to modify/approve the judgments of QUALITY prior to updating the permanent data base.

QUALITY will provide a summary set of information abstracted from the outputs of the 2MAPPS system to facilitate the quality checking of each night's processing by the cognizant scientist/engineer. Tools will be made available to allow back tracking information in a hierarchal fashion to progressively increased levels of detail, such that suspected problems can be investigated in an expeditious manner.

Data will be accepted or rejected at a minimum granularity of a scan.

Summary information will include at least the following:

1) Plots of calibration stars vs color and airmass.

2) Background plots for each band vs time and zenith angle.

3) Noise plots for each band vs time.

4) Seeing plots vs time.

5) Scatter plots of differential photometry, positions, and completeness from scan overlaps for each band.

6) Fraction of catalog stars matched for each scan and match statistics.

7) Source counts for each scan.

8) Galaxy counts for each scan.

The feasibility of visually inspecting a compressed format of all images is being considered.

3.17 DBMAN (Database Processing)

Function: The DBMAN Subsystem will update tables in databases and provide convenient access to them.

Databases: There will be two major databases. One will be called the "Internal Database" and will be the working database. The other will be called the "Released Database" and will be the database available to the public. Each of these two major databases will contain the following tables:

- Point source catalog (big, split up for parallel access)
- Extended source catalog
- Coadded image catalog
- Calibration file and history file
- Point Spread Function history file
- Optical distortion model history file
- Flat and dark frame history file
- Survey planning file
- Observation scan log
- Scan processing and quality log
- Scan statistics log
- IPAC operational parameters log
- Observatory operational parameters log
- Tape logs

Called By: The DB_ADD program (add one day of data to the Internal Database) will be run either by EXEC or manually, after the pipeline processing has been completed and determined to have no serious flaws. It is anticipated that when production first begins, during the "shakeout" period, DB_ADD will be run manually only after a thorough checkout of the night's data and possible reprocessing has been completed. When data re-processing becomes rare, DB_ADD will be called by EXEC following data verification by the QUALITY subsystem. Other programs and subroutines will be called as required.

Input:

The following DBI (database input) files for each day:

1) source listings (band-merged and calibrated) [from CALMON]

2) Star catalog associations

3) other scientific data [from various subsystems]

4) processing status [from all pipeline subsystems]

Note: DBI files will be normal ASCII files to allow easy viewing, editing, and shipment to 2MASS science team members.

Output:

1) processing summaries from production programs

2) various queries and reports from access programs

3) Final catalog (FITS format)

Processing:

1) Production Software:

DB_ADD - Add one day of data to the Internal Database. (First looks at all inputs and aborts if irregularities are found.)

DB_DELETE - Delete one scan of data from the Internal Database.

DB_DUPS - Flag duplicate sources in the Internal Database. (Will be run periodically as required.)

DB_PUBLIC - Create Released Database from current internal Database

DB_FINAL - Create Catalog from Released Database (probably FITS)

DB_EXT_SRC_CLEANUP - Mark extended sources within TBD arcsec of bright sources as potentially contaminated.

2) Access Software:

DB_ACCESS - GUI interface providing access to the Internal Database or Released Database.

DB_PROCSTAT - Give a summary of the current processing status.

DB_SOURCES - List data for all sources near a given coordinate.

(subroutines) - Access Internal Database or Released Database.

Design Requirements:

1) The pipeline processing will never access the database directly. This allows pipeline processing to proceed even if the database server machine is down. It also prevents database contamination in case a scan needs reprocessing.

2) It must always be possible to scan either database with any column limits in 30 minutes or less.

3) SYBASE is the intended data base software.

3.18 MPCAT (Minor-Planet Catalog Processing)

Function: To generate a list of asteroids in the region covered by a particular scan

Called By: PCP

Input: Scan geometry and time parameters, asteroid orbital elements

Output: List of asteroids covered by scan

Processing: The four corners defining the area covered by a scan are used to identify which asteroids in a corresponding data base were in the area when the scan occurred. Standard orbital calculations are performed, and a list of covered asteroids is generated for use in point-source catalog association and related tasks. TBD: this processor may make the associations between asteroids and band-merged point sources, in which case a status bit in each associated point-source record will be set.

Appendix A. Baseline Hardware Configuration

This appendix is intended to specify the baseline computer hardware system to be used for the production processing of the 2MASS data.

This baseline hardware specification is intended only for establishing a context for the design of the 2MAPPS production software. The actual hardware used for the production processing will change to take advantage of what is available at the time. The system used will run UNIX System V or a variant.

The baseline hardware for 2MASS processing consists of four subsystems:

1.) 2MASS Production System
- 6x
  - Sparc 1000 M 1104 (4 processor)
  - 4x
    - 64 MB RAM
    - 16 GB SCSI II Disk
    - 30 MFlops
- 8x 8mm (5 GB) tape drives with stackers

2) 2MASS Analysis System
- 1x
  - Sparc 1000 M 1104 (4 processor)
  - 4x
    - 64 MB RAM
    - 16 GB SCSI II Disk
    - 30 MFlops
- 4x 8mm (5 GB) tape drives with stackers

3) 2MASS Production Database Server
- 1x
  - Sparc 1000 M 1104 (4 processor)
  - 4x
    - 64 MB RAM
    - 8 GB SCSI II Disk
    - 30 MFlops

4) 2MASS Database System

1x

Sparc 1000 M 1104 (4 processor)

4x

64 MB RAM

10 GB SCSI II Disk

30 MFlops

2x 8mm (5 GB) tape drives with stackers

5) Archive Server: acquisition of an archive server for all or a subset of the 2MASS image data base will be reviewed periodically.

The Sparc 1000 series can accommodate up to 8 processors. The initial acquisitions will be 4 processor systems to achieve a proper I/O bandwidth margin for the anticipated processing load. The ability to double the number of processors provides a Mflops margin in the event the target code speed-ups are not achieved.

The four major components of the 2MASS processing system will be interconnected with high-bandwidth optical fiber, and a gateway will connect the 2MASS system with the resources of the rest of IPAC.

The analysis system has been acquired and is being used at IPAC (1 Sparc 1000 M 1104) and further aquisitions will ramp up to anticipate the production processing load by approximately 1 year.

Appendix B. Throughput Estimates and Hardware Requirements

The following estimates are based on the current pipeline run-times on in-house Sun Sparc 10/41 cpus. I/O estimates are based on an in-house Sun Sparc Center 1000.

The following table contains current pipeline cpu times for the equivalent processing jobs in the 2MAPPS, and estimated times in the final processing system with identified speed-ups.

CPU Requirements

Process	CPU Fraction Allocated	Sparc 10/41 cpu hrs/hrs of data (Current)	(Estimated)
TAPELOAD	**---**		
EXEC	**---**		
MONITOR	**---**		
DARKS	**0.003**	**0.03**	**0.03**
RDFRAME	**0.05**	**0.56**	**0.50**
DFLAT	**0.09**	**0.92**	**0.90**
FREXAS	**0.04**	**0.36**	**0.40**
POSFRM	**0.007**	**---**	**0.07**
PICMAN	**0.17**	**3.56**	**1.70**
PIXPHOT	**0.20**	**---**	**2.00**
PROPHOT	**0.20**	**29.16**	**2.00**
MAPCOR	**0.02**	**---**	**0.20**
CALMON	**0.002**	**---**	**0.02**
BANDMERGE	**0.006**	**---**	**0.06**
POSPTS	**0.006**	**0.06**	**0.06**
MPCAT	**---**		
GALWORKS	**0.20**	**2.00**	**2.00**
CONMAN	**0.002**	**---**	**0.02**
QUALITY	**0.004**	**---**	**0.04**
TOTAL CPU	**1.00**	**36.65**	**10.00**

The “current” cpu hours in the above table are based on cpu times for equivalent tasks in the prototype pipeline. The “estimated” cpu times for the final processing pipeline are based on identified speed-ups. Major performance improvements are anticipated for the point-source extraction (PROPHOT) and image creation (PICMAN) processors. The estimated times for PROPHOT are based on replacing the current detection step of the prototype software with the equivalent of DAOPHOT FIND, and retaining the prototype point-source extraction processing with minor

changes. DAOPHOT currently can detect detect sources in one-tenth the time taken by the prototype software.

Bandwidth Requirements to Disk:

1 observatory approx. 1900 MB/observing hour = 0.53 MB/observing s

Required throughput = 0.07 hrs-data/hr/processor
= 0.07 * acquisition rate = 0.04 MB/s

Required bandwidth to disk = 25 * data processing rate = 1.0 MB/s

The current Sun Sparc Center 1000 has a sustained backbone rate of 250 MB/s, and multiple disks with sustained transfer rates of 2 MB/s each.

The required bandwidth to disk assumes I/O is minimized by eliminating all unnecessary intermediate image files. PIXCAL will read in raw data and write out flattened frames and the flats themselves. PIXPHOT will read in flattened frames and write out the processed images. The galaxy processor will read the coadded images from disk. This amount of I/O is required to simplify the system design and integration.

Appendix C. Hardware Failure Contingency Plans

The 2MASS processing system is designed to take advantage of the demonstrated reliability of modern workstations. The system uses the inherent granularity of the scan for coarse-grain parallelism. Any scan can be processed by any cpu in an identical manner, so that any combination of cpus may be used, and a failure of any single cpu just decreases the net processing rate by a corresponding fraction. The use of multiple cpu machines makes the common backplane, power supply, etc, single point failures for that group of cpus, but these failures are low probability and recovery is a few days. In addition, the 2MASS analysis machine will be available for emergency supplementary production use in off-hours and as required.

The pipeline does not interact actively with the database machine, so that production may continue with a database machine outage by archiving updates.

We will not archive intermediate results during routine production or complicate the software with (non-rudimentary) restart capabilities as the risk of downtime is low enough that it will be much cheaper to restart a whole night's processing when necessary (the capability to produce intermediate files will be included for integration and debugging, however).

Recovery Plan For Different Failure Modes

Failure	Recovery
Software	**Critical error. System must undergo thorough testing during system integration (starting 1 year prior to full survey), ramping up to full production load. If software problem encountered for unusual data, continue pipeline processing and work off-line. If generic software problem discovered during full production, production may have to be halted until fixed, and then data rerun as appropriate.**
Data Base Machine	**Repair machine & re-load db - can continue & running pipeline archive db updates until repair effected as db machine is de-coupled from pipeline.**
Machine running EXEC	**Restart night with EXEC running on another machine.**
Disks with log data	**Restart night with log files on another drive. Log data will be duplicated on backup drive.**
m/n cpus in PCP farm	**Remove m cpus from active list, continue at (n-m)/n processing rate until repaired. EXEC will utilize only machines in active list.**
m/n disks in PCP farm	**Replace disks with in-house backups (on analysis machine), reload and continue.**
Tape units	**Hot backups will be available for input and output tape units.**
Internal sub-net	**Critical failure. Down until repaired.**
External network	**Can continue running pipeline by running EXEC on local cpu**.

We should have backups for all critical components in-house, or available on a few days notice. This includes tape drives, disks, cpu boards, and memory.

Appendix D. Dependencies on Camera, Telescope and Survey Design

The design contained in this document is contingent on certain assumptions about the design of the telescope, camera, and survey design.

The major assumptions are enumerated below:

Camera and Telescope Assumptions:

1) The telescope is designed with a combination of thermally compensated structure and calibrated thermal corrections such that the diameter encircling 80% energy shall not vary more that 20%(TBD) due to focus with time at any point in the focal plane.

2) The optics meet the requirements specified in the 2MASS Proposal (Reference 1.1-1).

3) The frame-to-frame offsets are consistent to 0.2 pixel.

4) The Observatory/IPAC interface is as described in the Observatory/IPAC interface document (Reference 1.1-3).

5) The position of the first frame of each scan can be determined from the scan header information to an accuracy of 1 arcminute or better with simple offsets.

Survey Design and Execution Assumptions:

1) Calibration scans of at least 2 degrees in length are performed approximately every two hours during survey operations on unconfused fields containing calibration stars, and several nights are devoted to color, extinction, and psf calibration before the survey starts. Also, one night should be devoted to recalibration after each significant event affecting the optics, and every 1-2 months on a regular basis during the survey. Linearity calibration sequences are done at (TBD) intervals.

2) A sequence (approximately 200) of dark frames is taken at the beginning and end of each observing night at a minimum.

3) Scans are of a minimum length of 6 degrees to facilitate calculation of flat fields and associated diagnostics.

4) Scans overlap by 10% and are taken in a primarily contiguous manner to facilitate the extraction of galaxies and the automatic detection of non-photometric scans by comparing the photometry in the overlap region.

5) Sufficient quality control is performed on-site so that at least 75% of the data forwarded

to IPAC is usable for catalog production, and essentially 100% of it can be processed automatically (complete data and header files, the dome was open, reasonably clear, etc).

6) The data formats are predefined, consistent, and stable.

7) IPAC is informed of significant reconfigurations at the observatories that affect instrument performance (recollimation, mirror repolishing, data system changes, etc).

8) IPAC will scan all data tapes for readability and content within 48 hours of receipt from the observatory, and relay problems to the observatory.

9) IPAC will attempt to process data fully through quality assessment and feedback scan quality information to the observatory within 4 weeks (TBD) of data receipt during routine operations; this cannot be guaranteed, however.

Appendix E. Intersubsystem Function RDFRAME

Function: The RDFRAME intersubsystem function will:

1) read an observatory scan file from disk and create corresponding FITS formatted frames;

2) create and maintain an electronic fits log file in the scan's appropriate date directory (date/fits.log).

Called By: DFLAT

Input: The following files will be input from disk:

1) the 2MAPPS Scan_log file (currently created by the “makelog” program);

2) the observatory scan files: (read2-read1) and/or read1;

3) the observatory scan parameters file (date.scan#.SPF);

4) the RDFRAME NAMELIST control and override file.

5) calling list parameters from DFLAT

Output:

1) a memory resident FITS-formatted Read1 and Read2-Read1 frame per call;

2) date/fits.log file created and/or updated on disk with current RDFRAME status.

Processing: The specific purpose of this intersubsystem function is to supply DFLAT with FITS formatted frames. These scan files and associated scan parameters file will have been previously loaded to disk by the TAPELOAD subsystem from tapes shipped to IPAC from the observatory. A detailed description of the scan file and scan parameters file is contained in the Observatory/IPAC Interface Specification.

RDFRAME will be called by the DFLAT subsystem for each frame DFLAT needs, one frame of each read type at a time. Using a fortran direct access read, RDFRAME will extract the requested frames from within the corresponding scan file, which itself consists of a single large record of contiguous frames. RDFRAME will construct a FITS-compatible frame and return it along with a count of the total number of frames in the specified scan.

Included in the FITS format header computed and output by RDFRAME will be UT at the frame center, airmass, RA and Dec in epoch J2000, hour angle and sundry other parameters. Computation of these parameters will be based on data available in the Scan Parameters File, the IPAC log file and the RDFRAME NAMELIST file.

In addition, RDFRAME will create and maintain a log of frames that it has created. This log will be kept in the associated scan date directory.

Appendix F. Subsystem Function CONMAN

Function: To process regions of sky covered by overlapping scans for point-source confirmation.

Called By: QUALITY

Input: CONMAN reads band-merged source lists from the observation period being processed.

Output: CONMAN produces lists of confirmed sources and a processing summary that provides statistical information on position and photometric discrepancies.

Processing: The CONMAN function is invoked once per night by the QUALITY subsystem to process regions of sky covered by overlapping scans on that night for point-source confirmation.

Design Assumptions: Position refinement is not performed for confirmed sources; confirmation processing is applied only to pairs of overlapping survey scans (calibration scans are not processed by CONMAN); when more than two scans overlap, the pairing is done for scans closest to each other in time. Only two coverages are considered at a time, i.e., at scan endpoints, where overlap may produce four coverages, only sources within the area defined by the survey strategy for each scan are considered.

Appendix G. Intersubsystem Function SEEMAN

Function: To compute seeing/PSF information and generate a history file. SEEMAN is maintained as part of the GALWORKS subsystem.

Called By: PIXPHOT

Input: SEEMAN uses the coadded images as they are being developed by PICMAN to estimate the seeing and PSF information needed by PROPHOT and GALWORKS.

Output: SEEMAN writes the time-ordered seeing-model parameter data used by PROPHOT and GALWORKS.

Glossary

2MAPPS	2MASS Production Processing System
2MASS	2-Micron All Sky Survey
band	the spectral range defined by a filter; for 2MASS, J, H, or K_s
BANDMERGE	pipeline procedure to combine point source data for each of the three filters
CALMON	pipeline procedure to analyze calibration data and calibrate sources
coadd	an image produced by coaddition; usually refers to one coadded frame, 512x1024 pixels, a section of the entire coadded scan
coaddition	the process which takes the six samples of data and produces an image with increased resolution relative to the raw data; see PICMAN section
CONMAN	pipeline procedure to confirm and analyze point sources in overlapping scan regions
DBMAN	pipeline procedure to create and maintain databases (public and for production only) with both source and auxiliary information
DARKS	pipeline procedure that creates and analyzes dark frames
DCC	Darks, Calibration, and Catalog tape; a supplemental tape supplied by the observatory each night
DFLAT	pipeline procedure that creates, analyzes, and applies sky flats
DMT	Data Management Team; IPAC section in charge of day-to-day operation of the pipeline
EXEC	pipeline procedure that controls execution of all reduction/analysis/database tasks for the night
frame	a discrete quantity of contiguous pixels within a scan, handled as a separate file; 256x256 pixels for raw data, 512x1024 for coadded data
FREXAS	pipeline procedure to extract single-frame point sources for frame-to-frame offset computation and Read1 detections for photometric merging, compute the seeing parameter, and compute miscellaneous statistical parameters on each frame
GALWORKS	pipeline procedure that detects and measures extended sources
GB	gigabytes (10^9 bytes)
GSC	Guide Star Catalog; astrometric catalog used by POSMAN to position reconstruct point sources
GUI	Graphical User Interface; the “look and feel” of a window session
image	a two-dimensional matrix of ordered pixels (either camera or coadd), usually consisting of intensity (count) information
IPAC	Infrared Processing and Analysis Center; responsible for the processing of survey data and production of catalogs
LSB	Low Surface Brightness (galaxy)
MB	megabytes (10^6 bytes)
MFlops	an indicator of CPU speed (Millions of Floating point operations per second)
MONITOR	procedure that monitors pipeline execution
MPCAT	pipeline processor that identifies minor planets in a scan

NAMELIST	a standard FORTRAN construct to pass variables into a program
PCP	pipeline control program; procedure that controls the execution of subsystems involved in the reduction and analysis of all filters in a single scan
PICMAN	pipeline procedure that coadds processed overlapping raw frames
PIXCAL	pipeline procedure that controls the reduction and analysis of all frames in one scan through one filter for purposes of flattening the frames and computing frame offsets and other statistics
pixel, camera	a single picture element as viewed by one of the HgCdTe devices and imaged through the telescope, currently 2.0 arcsec in size; there are 256x256 in each frame
pixel, coadd	a single picture element in the coadded frames; after many camera pixels are combined, one coadd pixel is 1.0 arcsec in size
PIXPHOT	pipeline procedure that controls the reduction and analysis of all frames in one scan through one filter for purposes of constructing images and extracting point sources
POSFRM	pipeline procedure to reconstruct frame positions from detections in each band and astrometric catalogs
POSPTS	pipeline procedure that reconstructs the positions of point sources into equatorial coordinates
PSF	Point Spread Function; the 2-D character of a point source as imaged by the survey camera, through a given filter at a specific time
PROPHOT	pipeline procedure to detect and measure point sources
QUALITY	pipeline procedure to analyze and monitor data and processing quality; information fed back to observatory
RDFRAME	pipeline procedure to read frames from the offloaded tape data and parse header files to maintain information logs
Read1, Read2	a pair of frames observed consecutively, per step on the sky; the read1 is a quick exposure (50 msec) compared to the read2 frame (1.3 sec)
scan	a set of contiguous frames produced representing a strip of sky
SCSI	Small Computer System Interface; a popular bus structure for connecting disks and other devices to a CPU
SEEMAN	pipeline procedure to evaluate the PSF model as a function of time for other processors needing this information
SPECMARK	a set of standard benchmarks to indicate computer processing speed
SPECfp92	a set of standard benchmarks to indicate computer processing speed
SSCD	Standard Star Calibration Database; a compilation of magnitude and color information on all standards to be used in the survey
Sparc	Type of computer (Sun Computer) under consideration for pipeline
TAPELOAD	pipeline procedure to load and validate incoming tapes of survey data, per night
TBytes	terabytes (10^{12} bytes)
WSDB	Working Survey Data Base; the production database