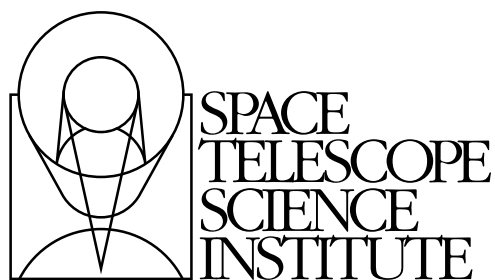

Version 4.0
January, 2002

HST Data Handbook for STIS



Hubble Division
3700 San Martin Drive
Baltimore, Maryland 21218
help@stsci.edu

User Support

For prompt answers to any question, please contact the STScI Help Desk.

- **E-mail:** help@stsci.edu
- **Phone:** (410) 338-1082
(800) 544-8125 (U.S., toll free)

World Wide Web

Information and other resources are available on the Web site:

- **URL:** <http://www.stsci.edu>.

STIS Revision History

Version	Date	Editors
1.0	February 1994	Stefi Baum
2.0	December 1995	Claus Leitherer
3.0, Vol., I	October 1997	Mark Voit
3.0, Vol. II	October 1997	Tony Keyes
3.1	March 1998	Tony Keyes
4.0	January 2002	Bahram Mobasher, Chief Editor, HST Data Handbook Michael Corbin, Jin-chung Hsu, Editors, HST Introduction Thomas M. Brown, Editor, STIS Data Handbook

Contributors:

T. Brown, J. Davies, R. Díaz-Miller, R. Downes, L. Dressel, B. Espey, P. Goudfrooij, P. Hodge, J. Kim Quijano, J. Kriss, C. Leitherer, G. Mitchell, B. Mobasher, M. Potter, C. Proffitt, K. Sahu, D. Stys, J. Valenti and N. Walborn.

Citation:

In publications, refer to this document as:

Brown, T. et al. 2002, in HST STIS Data Handbook, version 4.0, ed.

B. Mobasher, Baltimore, STScI

Send comments or corrections to:
Hubble Division
Space Telescope Science Institute
3700 San Martin Drive
Baltimore, Maryland 21218
E-mail: help@stsci.edu

Table of Contents

Preface	ix
----------------------	----

Part I: Introduction to Reducing HST Data

Chapter 1: Getting HST Data	1-1
1.1 Archive Overview	1-2
1.1.1 Archive Registration.....	1-3
1.1.2 Archive Documentation and Help	1-3
1.2 Getting Data with StarView	1-4
1.2.1 Downloading and Setting Up StarView.....	1-4
1.2.2 Simple Use of StarView	1-4
1.2.3 Marking and Retrieving Data with StarView	1-9
1.2.4 Using StarView to Retrieve Calibration Files and Proposal Information	1-9
1.2.5 Advanced Features of StarView	1-10
1.2.6 StarView and the Visual Target Tuner.....	1-15
1.2.7 Quick Data Retrieval with StarView	1-15
1.3 Getting Data With the World Wide Web	1-16
1.4 Reading HST Data Tapes and Disks	1-17

Chapter 2: HST File Formats	2-1
2.1 Historical Perspective	2-2
2.2 FITS File Format	2-3
2.2.1 Working with FITS Image Extensions	2-4
2.2.2 Working with FITS Table Extensions	2-9
2.3 GEIS File Format	2-12
2.3.1 Converting FITS to GEIS	2-13
2.3.2 GEIS Data Groups	2-14
2.3.3 Working with GEIS Files	2-15
2.3.4 The "waiver" FITS format	2-17
Chapter 3: STSDAS Basics	3-1
3.1 Navigating STSDAS	3-2
3.1.1 STSDAS Structure	3-2
3.1.2 Packages of General Interest	3-2
3.2 Displaying HST Images	3-4
3.2.1 The display Task	3-5
3.2.2 Working with Image Sections	3-8
3.3 Analyzing HST Images	3-9
3.3.1 Basic Astrometry	3-10
3.3.2 Examining and Manipulating Image Data	3-12
3.3.3 Working with STIS, ACS, and NICMOS Imsets	3-14
3.3.4 Photometry	3-17
3.4 Displaying HST Spectra	3-20
3.4.1 FOS and GHRS Spectra	3-20
3.4.2 STIS Spectra	3-21
3.4.3 Producing Hardcopy	3-22
3.5 Analyzing HST Spectra	3-24
3.5.1 Preparing FOS and GHRS Data	3-24
3.5.2 Preparing STIS Spectra for Analysis	3-27
3.5.3 General Tasks for Spectra	3-30
3.5.4 STSDAS fitting Package	3-34
3.5.5 specfit	3-37
3.6 References	3-37
3.6.1 Available from STScI	3-37
3.6.2 Available from NOAO	3-38
3.6.3 Other References Cited in This Chapter	3-38

Part II: STIS Data Handbook

STIS Introduction	1
Chapter 1: STIS Overview	1-1
1.1 Instrument Capabilities and Design	1-1
1.1.1 The STIS Detectors	1-5
1.1.2 STIS Physical Configuration	1-6
1.2 Basic Instrument Operations	1-7
1.2.1 Target Acquisitions and Peakups	1-7
1.2.2 Routine Wavecalcs	1-7
1.2.3 Data Storage and Transfer	1-8
1.2.4 Parallel Operations	1-8
1.3 STIS Operations on Side 2	1-9
Chapter 2: STIS Data Structure	2-1
2.1 Overview	2-1
2.2 Types of STIS Files	2-2
2.2.1 Understanding Associations	2-4
2.3 STIS File Structures	2-5
2.3.1 STIS FITS Image Extension Files	2-6
2.3.2 Storage of STIS Two-Dimensional Data	2-6
2.3.3 Storage of STIS Tabular Data	2-8
2.4 Headers, Keywords, and Relationship to Phase II	2-9
2.5 Error and Data Quality Arrays	2-16
2.5.1 The Error Array	2-16
2.5.2 Data Quality Flagging	2-17
2.6 STIS Coordinate Systems	2-18
2.7 STIS Paper Products	2-20

Chapter 3: STIS Calibration	3-1
3.1 Pipeline Processing Overview	3-1
3.2 Structure of calstis	3-3
3.3 Data Flow Through calstis	3-10
3.4 Descriptions of Calibration Steps	3-16
3.4.1 ATODCORR: analog to digital conversion correction	3-16
3.4.2 BACKCORR: 1D spectral background subtraction	3-17
3.4.3 BIASCORR: small-scale bias subtraction	3-17
3.4.4 BLEVCORR: large-scale bias & overscan subtraction	3-18
3.4.5 CRCORR: cosmic-ray correction	3-19
3.4.6 DARKCORR: dark signal subtraction	3-22
3.4.7 DISPCORR: apply dispersion solution	3-23
3.4.8 DOPPCORR: correct reference files for Doppler shift	3-24
3.4.9 DQICORR: Initialize data quality file	3-25
3.4.10 EXPSCORR: retain individual CR-SPLIT exposures	3-26
3.4.11 FLATCORR: flat-field correction	3-26
3.4.12 FLUXCORR: conversion to cgs flux	3-27
3.4.13 GEOCORR: geometric distortion correction	3-28
3.4.14 GLINCORR and LFLGCORR: nonlinearity corrections	3-29
3.4.15 HELCORR: correction to heliocentric reference frame	3-30
3.4.16 LORSCORR: conversion to native format MAMA pixels	3-31
3.4.17 PHOTCORR: populate photometry header keywords	3-32
3.4.18 RPTCORR: sum repeated MAMA observations	3-32
3.4.19 SC2DCORR: echelle scattered light correction	3-32
3.4.20 SGEOCORR: small-scale distortion correction	3-33
3.4.21 SHADCORR: CCD shading correction	3-33
3.4.22 WAVECORR: wavecal correction	3-33
3.4.23 X1DCORR: locate and extract 1-D spectrum	3-35

3.4.24 X2DCORR: produce rectified 2-D spectral image	3-37
3.5 Recalibration of STIS Data	3-38
3.5.1 Mechanics of Full Recalibration.....	3-39
3.5.2 Rerunning Subsets of the Calibration Pipeline	3-43
3.6 Updates to calstis.....	3-44
Chapter 4: STIS Error Sources	4-1
4.1 Summary of Accuracies.....	4-1
4.2 Flats.....	4-3
4.3 Factors Limiting Flux and Wavelength Accuracy	4-4
4.3.1 Flux Accuracy	4-4
4.3.2 Wavelength and Spatial Accuracies	4-5
Chapter 5: STIS Data Analysis	5-1
5.1 Data Reduction Applications	5-1
5.2 STIS-Specific Reduction and Analysis Tasks	5-3
5.3 Working with Imaging Data.....	5-4
5.3.1 Sensitivity Units and Conversions	5-4
5.3.2 Spatial Information.....	5-4
5.4 Working with Rectified Two Dimensional Spectral Images	5-5
5.4.1 Sensitivity Units and Conversions	5-5
5.4.2 Wavelength and Spatial Information.....	5-8
5.5 Combining Dithered Spectroscopic Data	5-8
5.6 TIME-TAG Data.....	5-9
5.6.1 Heliocentric and Barycentric Time Correction	5-11
5.6.2 Converting TIME-TAG Data to an Image	5-11
5.6.3 Echelle TIME-TAG Data	5-11
5.6.4 Timing Analysis Tasks.....	5-11
5.6.5 Merging TIME-TAG Event Tables.....	5-12
5.6.6 TIME-TAG Data Calibration.....	5-12
5.7 Target Acquisition Data	5-12
5.7.1 Target Acquisition Basics	5-12
5.7.2 ACQ Data	5-15
5.7.3 ACQ/Peak Data	5-16
5.7.4 The tastis Task	5-17

Part III: Appendixes

Appendix A: IRAF Primer	A-1
A.1 Initiating IRAF	A-2
A.1.1 Setting Up IRAF	A-2
A.1.2 Starting and Stopping an IRAF Session	A-4
A.2 IRAF Basics	A-4
A.2.1 Loading Packages	A-5
A.2.2 Running Tasks	A-6
A.2.3 Getting Help	A-7
A.2.4 Setting Parameters	A-9
A.2.5 Setting Environment Variables	A-11
A.2.6 File Management	A-12
A.2.7 Troubleshooting	A-14
A.3 Getting IRAF and STSDAS	A-15
A.3.1 Retrieving the IRAF and STSDAS Software	A-15
A.3.2 Getting the Synphot Database	A-16
A.3.3 Extracting the synphot Unix Tar Files	A-17
Appendix B: HST File Names	B-1
B.1 Rootnames	B-2
B.2 Suffixes of Files Common to all Instruments	B-3
B.3 Associations	B-5
Appendix C: Observation Logs	C-1
C.1 Observation Log Files	C-1
C.1.1 Observation Log File Contents (October 1994 version)	C-3
C.1.2 Observation Log File Contents (February 1997 version)	C-4
C.2 Retrieving Observation Logs	C-9
C.3 Using Observation Logs	C-10
C.3.1 Guiding Mode	C-10
C.3.2 Guide Star Acquisition Failure	C-11
C.3.3 Moving Targets and Spatial Scans	C-12
C.3.4 High Jitter	C-13
Index	1



Preface

Introduction to Reducing HST Data

This data handbook provides an introduction to the process of retrieving and reducing Hubble Space Telescope (HST) data. The reduction procedures, calibrations, and sources of error specific to each active HST instrument (WFPC2, STIS, NICMOS and ACS) are explained in detail in their respective instrument data handbooks. However, we recommend a careful reading of this handbook before proceeding to the instrument data handbooks and before starting to work on your HST data. The present document is an updated version of chapters 1-3 of the HST Data Handbook, v.3.0, and is based on the information available as of December 2001. In particular, it is written under the assumption that the ACS and NICMOS instruments will be fully functional following HST Servicing Mission 3B (SM3B).

Many changes in the HST Data Archive and HST data reduction software have occurred since v. 3.0 of the Hubble Data Handbook. These differences are covered in this document and include, but are not limited to:

1. Expansion of the HST Data Archive into the Multimission Archive at Space Telescope (MAST), which currently includes 14 satellite mission archives as well as ground-based survey data.
2. The ability to retrieve MAST data using the World Wide Web.
3. New capabilities of the StarView program for searching for and retrieving HST and other MAST data.
4. The introduction of PyRAF, a new Python-based alternative to the IRAF cl shell, and
5. A new distinction between waiver FITS format, used to archive data from the older HST instruments such as WFPC and FOS, and the FITS extension format used for the newest instruments (STIS, NICMOS and ACS).

Future changes in this handbook are anticipated as MAST expands to cover additional missions, and as StarView and PyRAF evolve. The reader is advised to consult the STScI web site at <http://resources.stsci.edu> for the latest information. Moreover, as the present revision comes before SM3B, important revisions to the ACS file structure and data handling may be necessary after the installation of this instrument.

Bahram Mobasher (Chief Editor, HST Data Handbook)

Michael Corbin (Editor, Chapter 1)

Jin-chung Hsu (Editor, Chapters 2 and 3)



PART I:

Introduction to Reducing HST Data

The chapters in this part provides an introduction to the process of retrieving and reducing Hubble Space Telescope (HST) data.

■ Part I: Introduction to Reducing HST Data



CHAPTER 1:

Getting HST Data

In this chapter. . .

1.1 Archive Overview / 1-2
1.2 Getting Data with StarView / 1-4
1.3 Getting Data With the World Wide Web / 1-16
1.4 Reading HST Data Tapes and Disks / 1-17

This chapter describes how to obtain Hubble Space Telescope (HST) data files. All HST data files are stored in the Hubble Data Archive (HDA), which forms part of the Multimission Archive at STScI (MAST)¹. HST Guaranteed Time Observers (GTOs), Guest Observers (GOs) and Archival Researchers can retrieve data in either of two ways:

- Electronically over the Internet from the HDA, where data are stored immediately after they pass through HST pipeline processing.
- On data storage media written at STScI from the HDA. The options are Exabyte and DAT tapes, and will include CDs and DVDs in the future.

To retrieve data electronically you must first register as a MAST user². HST Principal Investigators (PIs) are *not* automatically registered. If you have not recently retrieved data, you should register or renew your registration before retrieving data from the HDA. PIs should register before their observations are made. GTO and GO observations normally remain proprietary for a period of one year, which means that during this period

1. MAST currently includes data from HST, FUSE, IUE, EUVE, ASTRO, HUT, UIT, WUPPE, ORFEUS, BEFS, IMAPS, TUES, Copernicus and ROSAT. Data from the FIRST radio survey, Digital Sky Survey (DSS) and Sloan Digital Sky Survey (SDSS) are also available.

2. By 2002, registration will no longer be required for public (non-proprietary) data.

other registered users cannot retrieve them without authorization from the PI. All calibration observations as well as observations made as part of the Public Parallel programs are immediately public. All observations made as part of the Treasury Programs begun in Cycle 11 will either be immediately public or have only a brief proprietary period. The HST section of MAST also contains several Prepared (fully reduced) data sets, including the Hubble Deep Fields, the Hubble Medium Deep Survey Fields, and a composite quasar spectrum, which are also public.

This chapter describes how to search the HDA, how to electronically retrieve files from it, and how to request and read tapes and disks containing HST data. As an aid to retrieving their data, PIs will automatically receive e-mail notification of the status of their observations two times: first, when the first datasets for their proposal are archived, and second, when all the datasets for their proposal and all necessary calibration files have been archived.



Note for Advanced Camera for Surveys (ACS) Users: Calibrated ACS images are approximately 168 MB in size, larger than those of any other HST instrument. Therefore, electronic retrieval of ACS data is enabled only for those with broadband (> 100 KB/s) Internet connections, in order to ensure uninterrupted transmission of individual files. Users retrieving large numbers of ACS files should also consider requesting them on tape or disk.

1.1 Archive Overview

The HDA contains all HST observations ever made. It also contains a database that catalogs and describes these observations. There are currently two ways to search and retrieve data from the HDA. The first is a program called StarView, which acts as an interface to the HDA. StarView currently runs as Java-based, stand-alone application that can be downloaded from the web site <http://starview.stsci.edu>. Previous versions of StarView, such as XStarView, are no longer available. The second search and retrieval method is through the HST section of the MAST web site, <http://archive.stsci.edu>. StarView is the more powerful of the two methods, and in particular allows an examination of the calibration files applied to a given data file. StarView also provides an interface to the Visual Target Tool (VTT) in the Astronomer's Proposal Tool (APT) suite of programs. The VTT interface can display archive observations on a Digital Sky Survey (DSS) image alongside planned observations. StarView is thus

recommended for observation planning, duplication checking, calibration file review, investigation of On-The-Fly Reprocessing flags and proprietary status. It is also recommended for those needing to retrieve large numbers of datasets, and those needing to examine calibration files. The MAST web site interface to the HDA has the same basic capabilities as StarView, and may be preferable for those requiring simple retrievals of datasets. Both StarView and the MAST web site allow simultaneous searches of the other MAST mission archives for all HDA searches. They also offer simple preview of the capabilities of HST datasets when available, as well as links to literature references citing a given dataset, using the Astrophysics Data System (ADS). In later sections we discuss StarView and the MAST web site in more detail.

1.1.1 Archive Registration

The simplest way to register and retrieve HST data is to complete the form on the Web page at: <http://archive.stsci.edu/registration.html>.

Registration requests may also be sent to the HDA hotseat, at: archive@stsci.edu.

The PI of each HST proposal must request proprietary access for their data, and for anyone else whom the PI wants to have access to it. PI retrieval permission is not granted automatically, for security reasons. PIs wishing to allow others access to their proprietary data should make that request to archive@stsci.edu.

When registration is granted, your account will be activated within two working days, and you will receive your username and password via e-mail.

1.1.2 Archive Documentation and Help

The MAST web site provides a wealth of useful information, including an online version of the HST Archive Manual available at <http://archive.stsci.edu/hst/manual/>. Investigators expecting to work regularly with HST and other datasets supported by MAST should also subscribe to the MAST electronic newsletter by sending an e-mail to archive_news-request@stsci.edu and putting the single word *subscribe* in the body of the message. Questions about the HDA can be directed to archive@stsci.edu, or by phone to (410) 338-4547.

1.2 Getting Data with StarView

1.2.1 Downloading and Setting Up StarView

The latest version of StarView runs under versions 1.2.2 and later of Java and may be downloaded from <http://starview.stsci.edu>.

This site also includes a FAQ page and news on releases and updates. StarView will automatically update itself to the latest version, so users do not have to worry about additional installations. Following its installation on computers running Unix and Linux, begin StarView by typing

```
> StarView
```

at the system prompt. Under Windows and Macintosh systems, StarView will appear as an icon. The StarView session then begins, first with an Information window explaining navigation within StarView, and a request for the user to specify an object name resolver (SIMBAD or NED) for use in HDA searches. The first-time user are asked to supply their e-mail information in order to allow StarView to communicate the results of its attempts to retrieve the files requested from the HDA. This e-mail information includes the user's SMTP host, or the computer from which e-mail messages are routed. If unsure of your SMTP host, ask your system administrator. These queries can be turned off for future sessions once this information has been supplied.

1.2.2 Simple Use of StarView

We now proceed to an introduction to the use of StarView. A more detailed description of its capabilities is provided at the web site above, which should also be consulted for more advanced topics such as its Table Exportation and Cross-Qualification functions.

The basic function of StarView is to enable the user to first search the HDA (and the other mission archives in MAST) for data files matching criteria such as object name, position, or proposal number, then allow the user to navigate through the set of files matching those criteria, and finally to let the user retrieve some or all of the files found in the search. Several options for the type of search that can be performed (e.g. by a particular instrument) will be discussed later.

The design of StarView is similar to that of a Web browser. At its top are pull-down menu bars including File, Searches, and Help. The Help menu offers links to documents including the StarView FAQ page. Beneath these menu bars is a row of buttons that run StarView's basic functions, such as searching, marking files for retrieval, and previewing images. A Help

button allows users to display pop-up windows describing the function of the different StarView buttons and windows, by first clicking the Help button, then the item of interest. Beneath the row of buttons is the Qualifications table, which is displayed when a search is begun. It consists of several cells corresponding to the search parameters the user wishes to use, e.g., object name, proposal I.D., or instrument. Below this window will appear the Results table, displaying the datasets found to match a given set of search parameters entered into the Qualifications table. For the purpose of introduction, we will describe the use of the most basic search option, called “Quick Search,” which can be started by clicking the “Quick” button at the top left of StarView.

As an example of the use of the Quick Search option, we will request all available WFPC2 data for the galaxy M87. This is done by typing “WFPC2” and “M87” in the Instrument and Target Name cells of the Qualifications section, then clicking the “Search” button at the top left of the StarView window. The results of the search will then be displayed in the bottom panel of StarView, as shown in figure 1.1. These results include the dataset name, instrument name, R.A. and Dec of the target, and the instrument aperture used. Note that these parameters could also have been specified in the Qualifications section, as can other parameters including proposal I.D. number, proposal P.I. name, and image central wavelength (corresponding to particular instrument filters or gratings).

Figure 1.1: Results of StarView Quick Search for WFPC2 files of M87

File Edit View Searches Comment Window Help

Quick Search Scan Prev Stop Next Scan Mark All Unmark All XQual Export Preview Refs. Overlay DSS Help

Enter qualifications for: Quick Search

Load Qualifications Save Qualifications Clear Qualifications Target Resolver

Label	Qualification (click cell to edit)	Database Field Name	Logical Type
Dataset Name:		sci_data_set_name	datasetname
Radius (degrees):	0:10	(sci_ra,sci_dec)	radius
RA:		sci_ra	ra
Dec :		sci_dec	decl
Instrument:	WFPC2	sci_instrume	instrument
Flag:		sci_expflag	expflag
Apertures:		sci_aper_1234	wildtext

Results for: Quick Search

Proposal ID: 5122 Release Date: 1995-02-27 02:37:53

PI (last name): FORD

Target Name: M87

Target Description: GALAXY;ELLIPTICAL;NUCLEUS;

Instrument: WFPC2 Config: WFPC2

Start Time: 1994-02-26 19:10:17.5 Fla... NORMAL

Dataset Name:	RA:	Dec :	Instrument:	Flag:	Apertures:
U2900101T	+12:30:49	+12:23:28	WFPC2	NORMAL	PC1
U2900102T	+12:30:49	+12:23:28	WFPC2	NORMAL	PC1
U2900103T	+12:30:49	+12:23:28	WFPC2	NORMAL	PC1
U2900104T	+12:30:49	+12:23:28	WFPC2	NORMAL	PC1

V3 Angle: 111.001 sci_fov_config SM-1

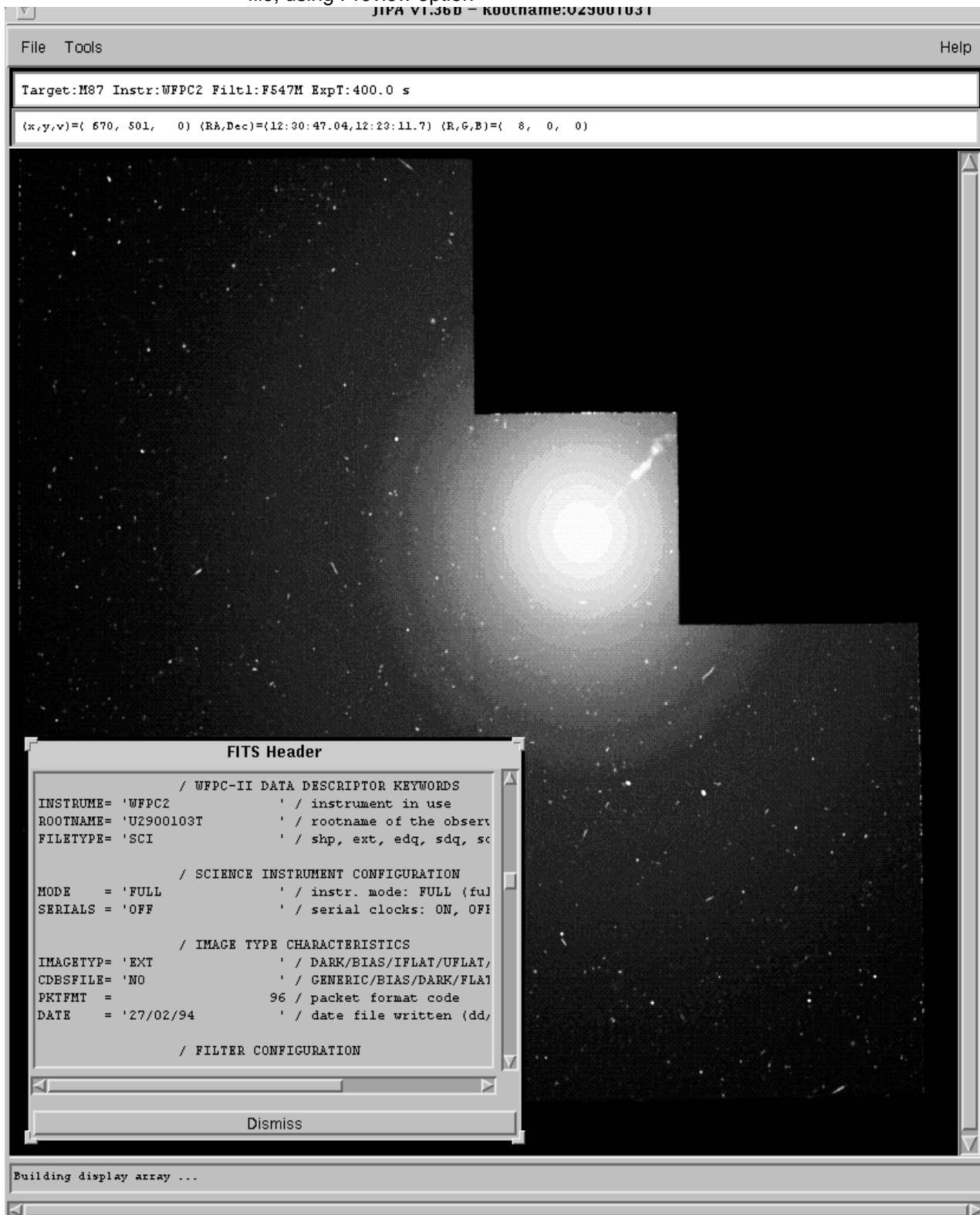
All records retrieved. Static Snap Table No Update 1 of 30

Clicking on a given dataset in the Results table will display the information shown in the cells above it (Proposal ID, Release Date, P.I., etc.). You may browse through the retrieved datasets either by using the mouse and scroll bar, or by using the navigation buttons (Scan, Previous, Next) in the top row of mouse buttons. The Scan option will automatically step through all of the files retrieved in the search, provided that the right most button at the bottom of the Results window is toggled to “Update.” If this button is toggled to “No Update,” the Scan option will go straight to the end of the list of files.

The ability to obtain a preview is available for many, but not all, of the datasets in the HDA (e.g., previews are not available for many FOC datasets). This is done with the “Preview” button, if it is enabled. For images, this will display a re-sampled version of the image using the Java Image Preview Application (JIPA) tool that is part of StarView. For spectra, a simple GIF image of the calibrated spectrum will be displayed. JIPA and VTT can also display an image's FITS header, under the JIPA “Tools” menu. The JIPA preview of the WFPC2 image U2900103T retrieved in the previous search for WFPC2 images of M87 is shown in figure 1.2, along with the window displaying part of the FITS header file of this image.

Other display options with StarView include “DSS,” which will display a 20' x 20' Digital Sky Survey image at the target coordinates, while the “Overlay” button will display the same DSS image with outlines of the HST instrument apertures at the target coordinates superimposed on it, at the orientation of the observation selected. The “References” button provides a link to any known published papers citing the dataset, as listed in ADS. Note that the HST images displayed by the Preview are of reduced quality compared to the actual data files, and cannot be downloaded. They are only meant to provide a quick check that the datasets found by the search met the search criteria, i.e., contained the object(s) of interest, and are of the desired quality.

Figure 1.2: JIPA preview of WFPC2 image U2900103T, along with image header file, using Preview option



1.2.3 Marking and Retrieving Data with StarView

Datasets are marked for retrieval by first clicking on them, then using the “Mark” button at the top of StarView. There is also the “All” button, which will mark all the datasets retrieved in the search. Marked datasets will be displayed in the Retrieval window. Datasets still within their proprietary period will be displayed in yellow, and users other than the proposal PI and those authorized by the PI will not be able to retrieve them. The release date of files still within their proprietary period will also be indicated on the search results form.

If satisfied with the marked datasets, choose “Submit” in the Retrieval window to retrieve them. You will then be queried for both the type of data files associated with the dataset(s) to be retrieved, and for the method of delivery of these files. The options for type of file include files calibrated with the On-The-Fly-Recalibration (OTFR) pipeline for the WFPC2, NICMOS, STIS and ACS instruments. OTFR applies the best available calibration files (i.e., dark current and flat field images taken closest in time to the observations) to the uncalibrated data files. You may also request the uncalibrated (raw) files and calibration files separately. For some of the earlier instruments, e.g., WFPC and FOS, you may request both the calibration files actually applied to the images, as well as those that should provide the best calibration of them, if recalibration is desired. You may also request Data Quality and Observation Log files.

Options for data delivery include ftp transfer by the user from the HDA staging disk, automatic transfer from the HDA via the Internet to a directory specified by the user, and the mailing of tapes or disks. If Internet delivery is specified, you will be queried for the name of the computer and directory in which the files are to be placed, as well as your user name and password on that computer (these requests are encrypted, so there is no danger of your login information being stolen). Upon final submission of the request, you will receive an e-mail message acknowledging its receipt, and another message after all the requested files have been transferred. The status of the request, i.e., how many files have been transferred and any errors that have occurred, can be checked on a Web page that will be given in the acknowledgment message.

1.2.4 Using StarView to Retrieve Calibration Files and Proposal Information

StarView allows several additional types of searches of the HDA besides the Quick Search option described above. These can be selected from the Searches menu bar at the top of the StarView screen. One such search option is by instrument. This is necessary for identifying calibration reference files. As an example, selecting the option “WFPC2 OTFR” under the Instrument and WFPC2 sub-menus of the Searches menu, and then

entering “M87” under Target Name in the qualifications box, brings up the screen shown in figure 1.3. This screen shows all the calibration images and files applied by OTFR to the first file in the set of WFPC2 images of M87, as well as whether the application of these files was performed or omitted in the calibration pipeline. This is the same set of images found by the Quick Search query described above, and the same information for the other datasets from this search can be found using the Previous, Next and Scan buttons. Once these calibration images have been identified, further information on them can be obtained. For example, taking the name of the flat field file found in the above search and entering it into the “WFPC2 Calibration Data” Searches option will retrieve information on when and where this file was taken, and the date after which its use is recommended. This will help users decide if they would prefer to recalibrate their data using different files.

StarView can also be used to search for and view the abstracts of accepted HST proposals. Like the Preview capability of StarView, this provides additional information about a given dataset and whether it may be useful for your science goals. Viewing proposal abstracts is an option under the Searches menu, and an example is shown in figure 1.4. The Qualifications window again offers several parameters by which this search can be constrained, including proposal I.D. number, HST cycle, P.I. name, and combinations thereof. In the example shown only the proposal I.D. number was used.

Finally, StarView can be used during the Phase I proposal process to see whether or not HST observations of a given object or object class have already been made, or else are scheduled for execution. Specifically, the Duplications option under the Searches menu allows users to check a database containing both HDA files and a list of queued observations in order to see if a given object has been or will be observed. Similarly, under Duplications the user may also query the database of proposal abstracts for a given object or object class, to check for archived or scheduled observations.

1.2.5 Advanced Features of StarView

In addition to its basic search and retrieval function, StarView allows users to cross-qualify results from separate searches of the HDA, and to export the results of searches to disk as ASCII files. These operations are performed with the “XQual” and “Export” buttons, respectively. As an example of cross-qualification, a user might want to identify all the spiral galaxies for which both WFPC2 images and STIS spectra have been obtained. This could be accomplished with the Cross-Qualification feature by first doing two separate Quick Searches, in which these respective instruments are specified in the query box, and in which “Galaxy;Spiral” is typed in the Target Description box for both searches. Clicking the XQual

button, specifying “Target Name” as the common field in the two sets of search results (as shown in figure 1.5), and clicking the “X-Qualify” button then identifies the galaxies occurring in both lists. StarView then places these galaxy names in the Target Name box of a new Quick Search window. Clicking the Search button with “WFPC2,STIS” entered for Instrument then gives a list of all the WFPC2 and STIS datasets for these galaxies. The Cross-Qualification function can also be performed on the files produced by the Export feature.

Figure 1.3: Results of StarView search for WFPC2 OTFR calibration files for M87

File Edit View Searches Comment Window

Quick Search Scan Prev Stop Next Scan Mark All Unmark All XQual Export Preview Refs. Overlay DSS Help

Enter qualifications for: WFPC2 OTFR

Load Qualifications Save Qualifications Clear Qualifications Target Resolver

Label	Qualification (click cell to edit)	Database Field Name	Logical Type
PI (last name):		sci_pi_last_name	lastname
Proposal ID:		w2r_proposid	peppropid
Target Name:	m87	sci_targname	targetname
Release Date:		sci_release_date	datetime
Radius (degrees):	0:10	(sci_dec,sci_ra)	radius
Dec :		sci_dec	decl
RA:		sci_ra	ra

Results for: WFPC2 OTFR

PI (last name): FORD Proposal ID: 5122

Target Name: M87 Release Date: 1995-02-27 02:37:53

RA: +12:30:49 Dec : +12:23:28

Dataset Name: U2900101T Filter 1: F658N Serials: OFF Mode: FULL Shutter: A

A-D Gain: 7.0 Filter 2: Exptime: 1400.0

Date of Last Software Change (calwp2): 1994-05-04 00:00:00.0

Date of Last On The Fly Calibration Action Update: 2001-09-21 08:04:50.1

SOFTWARE SWITCH	REFERENCE FILE	OTFR FILE/TABLE	OTFR ACTION
ATODCORR	Atod Correction	DBU1405IU.R1H	PERFORM
BLEV CORR	Engineering File	U2900101T.X0H	PERFORM
BIAS CORR	Bias Correction	E4P1629BU.R2H	PERFORM
DARKCORR	Dark Current	F5O1154MU.R3H	PERFORM
FLATCORR	Flat Field	E391433LU.R4H	PERFORM

Figure 1.4: Results of the StarView search for the abstract of Proposal 8725

Enter qualifications for: Proposal Abstract

Load Qualifications Save Qualifications Clear Qualifications Target Resolver

Label	Qualification (click cell to edit)	Database Field Name	Logical Type
Proposal Type:		type	varchar
Cycle:		cycle	smallint
TAC Panel:		sci_cat	varchar
Title:		title	wildtext
PI (Last name):		pi_lastname	varchar
Abstract:		abstract	wildtext
Proposal ID:	8725	prop_id	smallint

Results for: Proposal Abstract

Proposal ID: 8725

Proposal Type: GO Cycle: 9

TAC Panel: GALAXIES

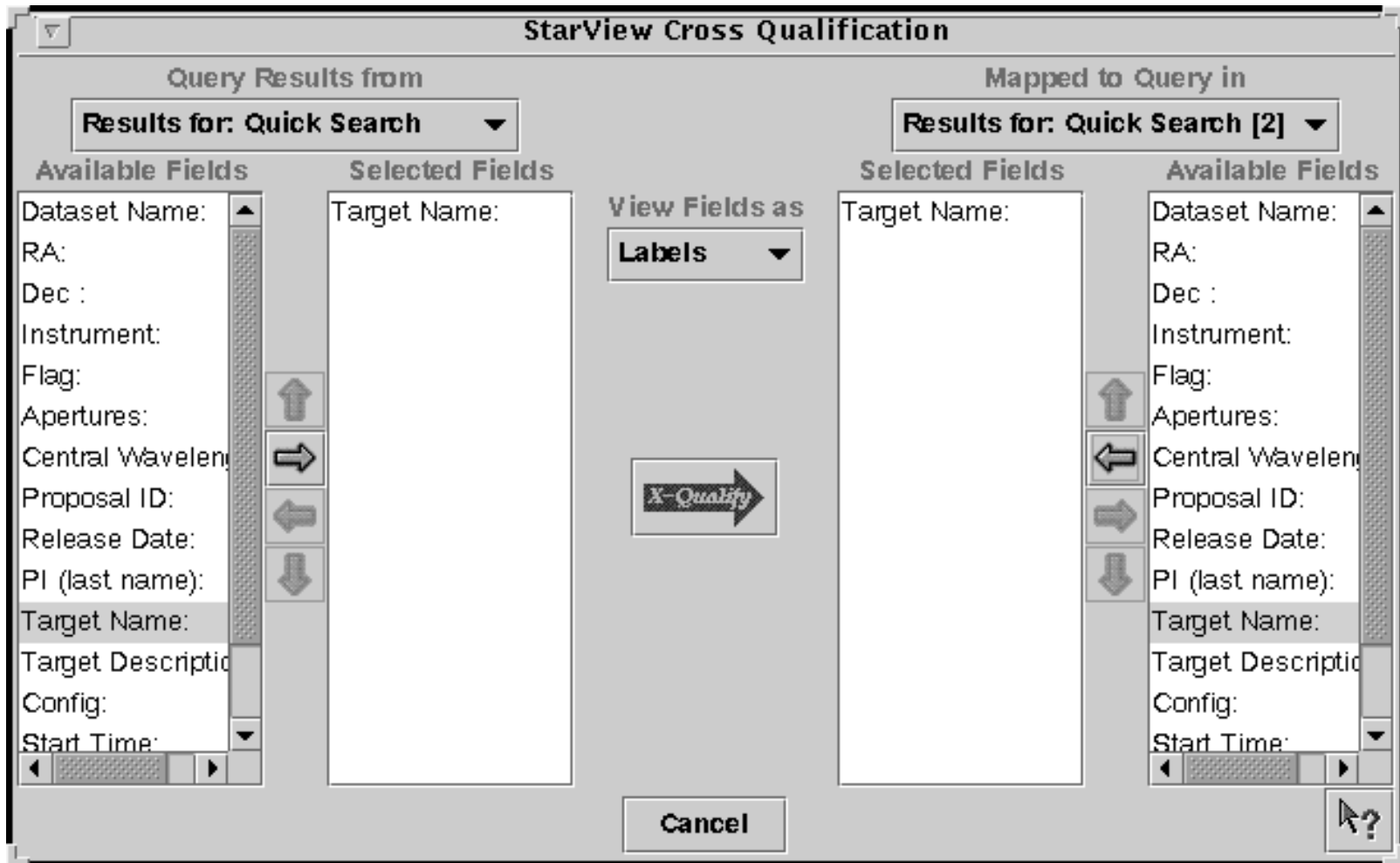
PI (Last name): Zepf

Title: The Formation Epoch(s) of Globular Clusters Around Ellipticals fr

Abstract: We propose to study the formation history of elliptical galaxies by obtaining far-UV photometry of their globular clusters. The far-UV photometry will be used in conjunction with existing optical data to constrain the ages of these clusters, and thus the formation

One record matched search criteria Static Snap Table No Update 1 of 1

Figure 1.5: Example of Cross Correlation Feature, in which Target Name has been chosen as the common feature to search for in two Quick Search result lists.



1.2.6 StarView and the Visual Target Tuner

The Visual Target Tuner (VTT) is part of the Astronomer's Proposal Tool (APT) package, which has been created to aid astronomers in planning their HST observations during the Phase I and Phase II proposal stages (see <http://apt.stsci.edu>). VTT is an image display tool which allows the user to display DSS images or local FITS images with proper World Coordinate System keywords in the headers. It offers more features than JIPA, which is the default StarView display tool. However, for a limited number of operating systems, the VTT can be used with StarView. VTT offers the particular advantage that it can overlay the instrument apertures of multiple observations on a single DSS image. Clicking on these apertures will also highlight the associated datasets in StarView.

Currently, to combine StarView and VTT requires downloading and installing APT from the above Web site. APT is only available for those operating systems with the Java Virtual Machine 1.3 (JVM 1.3). You can download the StarView/VTT package with JVM 1.3 included (a large download), or if you already have JVM 1.3 installed, you can get the smaller APT/VTT package. To make StarView use VTT, you must change your Viewer options from JIPA to VTT. Go to the Environment sub-menu of Edit in StarView, and change JIPA to VTT in the Viewers section. If VTT is not listed here, you should reinstall the two programs. Following this change, the Preview, DSS and Overlay buttons of StarView should all bring up VTT.

Once VTT has been installed, you can also bring up StarView from it. Enter in to StarView mode by clicking on the StarView button in the lower left hand menu of VTT. Clicking on a DSS image will then spawn a Quick StarView screen with the R.A. and Dec of the position you clicked loaded into the search fields. You can enter other constraints into these fields as usual. Search results can be displayed on the VTT screen by selecting the results in StarView, and pressing the Overlay button.

1.2.7 Quick Data Retrieval with StarView

The following steps summarize the basic process that PIs need to go through to retrieve their data with StarView. These steps follow registration as a MAST user, notification from STScI that the observations for a given proposal are complete, and providing StarView with your e-mail information. They are intended as a quick reference for this process.

1. Start StarView.
2. Click the “Quick” button.
3. Enter your PI name and/or proposal ID number in the appropriate cell.
4. Click the “Search” button.

5. Use the “Scan” button to step through the retrieved files, after toggling the right most button at the bottom of the Results window to “Update,” to verify that all datasets have been retrieved.
6. Preview some or all of the datasets if desired, to verify data quality and target acquisition.
7. Click “All” to mark all datasets for retrieval, or “Mark” to mark individual datasets for retrieval.
8. Click “Submit” in the window that will be spawned by marking the files.
9. Enter your MAST username and password and specify the means of data delivery. StarView remembers your name and password from past searches so it does not have to be entered each time.
10. Click “Done”, and your data are on their way. You will receive an e-mail message when your retrieval has been queued, and another when the transfer is complete.

1.3 Getting Data With the World Wide Web

HDA datasets can be searched for, previewed and retrieved via the World Wide Web in very much the same way as with StarView. As noted in section 1.1, StarView offers more capabilities for this process, including cross-qualification, the use of VTT, and more information about instrument calibration files. However, Web retrievals may be preferable in some cases, particularly when information on calibration files is not needed, and the hypertext on the Results pages makes it easy to access all the information they contain. The starting point for Web-based searches of the HDA is the MAST web site at: <http://archive.stsci.edu>³

This web page is shown in figure 1.6. A powerful feature of MAST is that all of its mission archives, including the HDA, can be searched simultaneously. This is done with the Cross-Correlation Target Search option shown on the MAST home page. This search will return all datasets for all missions available for a given object or coordinates, according to the search constraints specified by the user (based on the wavelength region of interest), and will provide hypertext links to these datasets. If only HST datasets are desired, they can be accessed separately by clicking “HST” on the MAST home page.

3. European archive users should generally use the ST-ECF Archive at <http://archive.eso.org>. Canadian users should request public archival data through the CADCF web site at <http://cadcfwww.dao.nrc.ca>. Proprietary data are only available through STScI.

The HST section of MAST offers tutorials about the HDA as well as a FAQ page and HDA news. It also provides links to HST Prepared datasets such as the Hubble Deep Field images. Clicking on the “Main Search Form” option of the HST section brings up the page shown in figure 1.7. Here the user is queried for the same search parameters as requested by StarView, e.g., Object Name, Instrument and Proposal I.D. Once these are entered, clicking the Search button returns a page listing the datasets found, which can then be selectively marked for retrieval. The data type and retrieval options remain the same as those for StarView. Previews of GIF files of the datasets are also available.

1.4 Reading HST Data Tapes and Disks

If you request HDA files on tapes or disks, you will receive them within a few weeks of your request. The tapes will contain **tar** files containing the requested datasets. The datasets will all be in FITS (Flexible Image Transport System) format⁴. You should thus first create a directory where you want your data to reside, e.g., /home/myname/myhstdata, go to that directory, then read the tape or disk using the Unix/Linux **tar** command to read the FITS files into it.

Currently, datasets obtained with HST's original instruments (FGS, FOC, FOS, GHRS, HSP and WFPC) as well as WFPC2 must have their FITS files converted to GEIS (Generic Edited Information Set) format in order to work on them with IRAF/STSDAS. Further information on HST file formats is presented in chapter 2. STSDAS is the package analysis software for HST data, and is discussed further in chapter 3. Datasets obtained with the other current HST instruments (ACS, NICMOS and STIS) should be reduced in FITS format without conversion to GEIS. STSDAS support for the analysis of WFPC2 data in FITS format is currently planned.

The steps for reading and converting FITS files to GEIS files are as follows:

First bring up IRAF/STSDAS in your IRAF home directory by typing

```
> cl
```

4. A description of FITS format and various supporting documents can be found at the Web site http://fits.gsfc.nasa.gov/fits_home.html

This will start an IRAF session. IRAF and STSDAS are organized into *packages*. To load a package, type its name. To begin with, you must load the **stsdas** and **fitsio** (FITS Input/Output) packages:

```
cl> stsdas
st> fitsio
```

The IRAF prompt (such as st>) shows the first two letters of the most recently loaded package. The **fitsio** package contains the STSDAS programs (called *tasks* in the IRAF/STSDAS environment) required to read and write FITS files to and from tapes and disks. The two principle tasks are **strfits** for reading files, and **stwfits** for writing them.

Next, set the IRAF environment variable *imtype* to specify that your data files are to be written in GEIS format. This is done by typing

```
fi> set imtype="hhh"
```

You should then move to the directory containing the FITS files.

The last step is to use **strfits** to read the data. Like most IRAF/STSDAS tasks, **strfits** has several parameters that control its function. You can either edit these tasks using the IRAF “epar” command, or specify them on the command line. For the purpose converting FITS files to GEIS files, the important parameter is *oldirafname*, which needs to be set to “yes” in order to keep the file rootname the same. To convert all the FITS files in a directory to GEIS files, type

```
fi> strfits *.fits " " oldirafname=yes
```

Figure 1.6: MAST Home Page

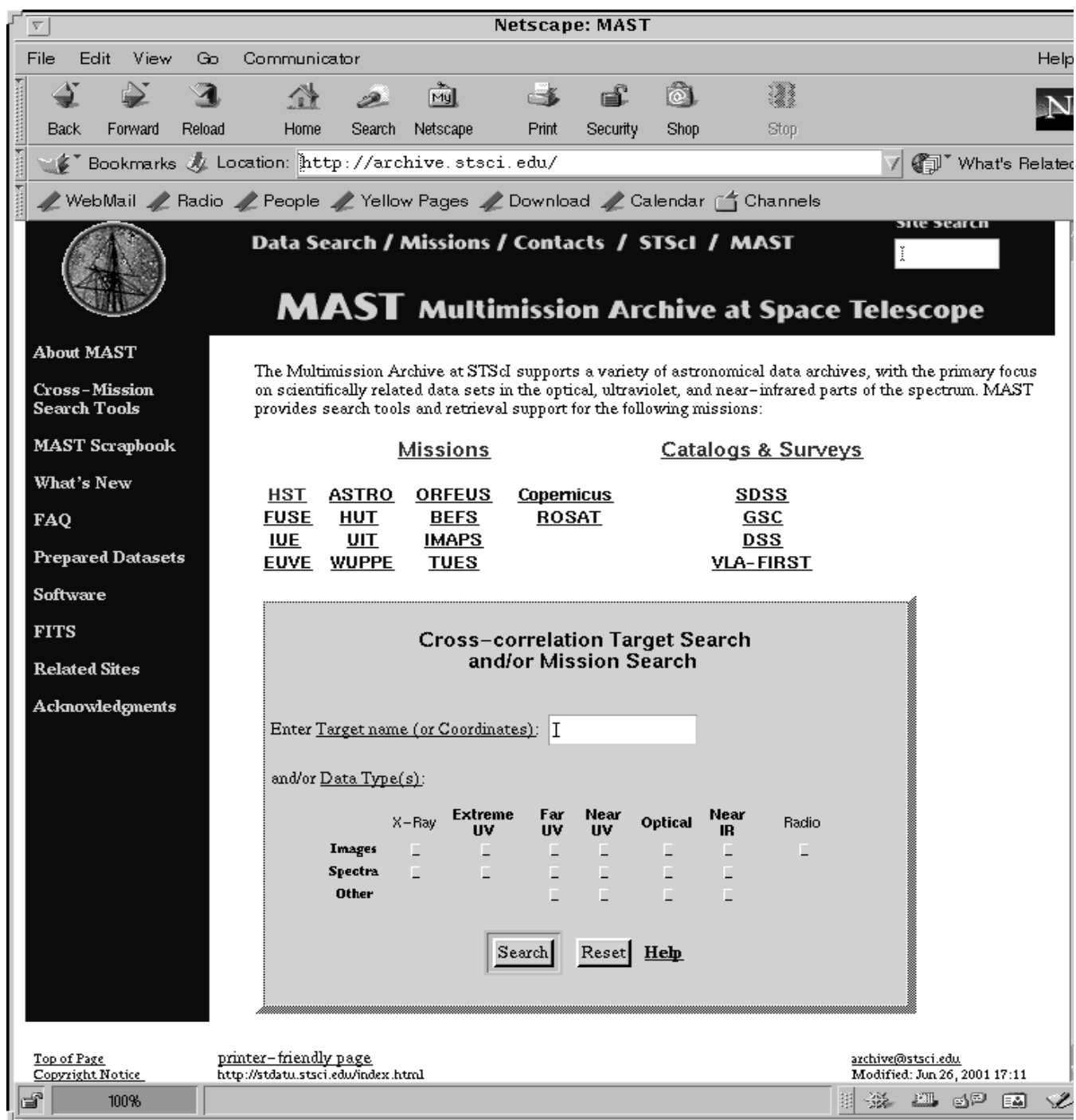


Figure 1.7: HST Archive Web search Form

HST Archive Search
[Archive Status](#) [Other Mission Search Pages](#)
[Important Downtime Message](#)

Object Name **Resolver**
☒ SIMBAD ☐ NED
☐ Don't resolve

RA **Dec** **Radius (arcmin)** **Equinox**
 ☐

Observation Date **Exp Time (sec)** **Observer** **Program ID**

Target Description **Dataset Name**

[Help...](#)

Imagers ☐ ALL ☐ NONE **Spectrographs** ☐ ALL ☐ NONE **Other** ☐ ALL ☐ NONE

☐ WFPC2 ☐ STIS ☐ FOS ☐ STIS ☐ FGS ☐ HSP
☐ WF/PC ☐ NICMOS ☐ GHRS ☐ NICMOS
☐ FOC ☐ FOC

Release Date
Archive Date
☐ Show calibration observations

Observation Band
 Central wavelength falls in this range (Å .. Å):

Output Options

Output columns (Default / Custom...)
☒ Mark
☐ Targname
☐ RA
☐ Dec
☐ Ang Sep
☐ Exp Time

Sort output by:
 1. ☐ reverse
 2. ☐ reverse
 3. ☐ reverse

Maximum number of hits ☐
Output Equinox ☐
☐ Show SQL query

This command will make GEIS format copies (having extension “.hhh”) of all the FITS files in the directory, with the same rootname. Following reduction and analysis of the GEIS files with the IRAF/STSDAS tasks, they may be written back into FITS format, on hard disk or to a tape or other storage media, with the **stwfits** task.

HST File Formats

In this chapter. . .

2.1 Historical Perspective / 2-2

2.2 FITS File Format / 2-3

2.3 GEIS File Format / 2-12

STScI automatically processes and calibrates all the data received from HST. The suite of software programs that performs this processing—part of a system known as OPUS—is frequently called the *pipeline*, and its purpose is to provide data to observers and to the HST Data Archive in a form suitable for most scientific analyses. Pipeline processing assembles data received from HST into *datasets*, calibrates the data according to standard procedures described in the instrument sections of this handbook, and stores both calibrated and uncalibrated datasets in the Archive.

Pipelines of older instruments (FOC, FOS, FGS, GHRS, HSP, WF/PC-1, and WFPC2) generate files in the so-called GEIS (stands for Generic Edited Information Set) format. Since GEIS is a machine dependent format, these files are converted to a specific kind of FITS file format, sometimes referred as “waiver” FITS, before being archived. We’ll explain the structure of this “waiver” FITS format later in this chapter. Since the “waiver” FITS format is only designed for archival purpose, it is necessary to convert it back to the GEIS format before further data processing and analysis using IRAF/STSDAS tasks.

Instruments installed after the 1997 servicing mission (STIS, NICMOS, ACS, and most likely all future instruments) have pipelines which generate FITS files directly. They are ready to be used by relevant IRAF/STSDAS tasks and, unlike the “waiver” FITS files, do NOT need to (and indeed, should not) be converted to GEIS format. Sometimes FITS files for the newer instruments are referred to as “FITS with extension” or “extended”

FITS files. But this can be misleading, since a “waiver” FITS file also has one (ASCII table) extension.

Much confusion has occurred about the two kinds of FITS files been archived at STScI. So we like to repeat one more time:



Older instruments (FOC, FOS, FGS, GHRS, HSP, WF/PC-1, and WFPC2) generate files in GEIS formats, but are stored and delivered as “waiver” FITS format in the archive, and need to be converted back to GEIS format before processing. Newer instruments (STIS, NICMOS, ACS) generate and store files in FITS format and should not be converted to GEIS.

This chapter describes these two HST file formats, first giving some historical perspective on the reasons why they were selected, then explaining the FITS and GEIS formats in more detail. STIS,ACS, and NICMOS observers should pay particular attention to the section on FITS files, which shows how to identify and access the contents of these files and covers some important conventions regarding header keywords. Veteran observers with the other instruments will find little new in the section on GEIS files, but newcomers to the older HST instruments should consult the material on data groups and conversion from FITS to GEIS before proceeding to chapter 3 of the HST Introduction.

2.1 Historical Perspective

In the early 1980’s, when GEIS was selected as the standard format for HST data files, it held several advantages over both FITS and the original IRAF format (OIF):

- GEIS allows floating-point data. The early incarnations of FITS accommodated only integer data, and this restriction to integers would have made data reduction and storage of calibrated data rather cumbersome.
- GEIS files can hold multiple images, each with associated parameters. This feature allowed the packaging of images from the four WF/PC-1 chips into a single unit, as well as the packaging of multiple FOS or GHRS readouts into single files. OIF files and early FITS files could contain only single images.

- GEIS data are stored in two parts, an ASCII header and a binary data file. The separation of these two pieces and the restriction of the header to ASCII made these headers easier to read and print in the days when computers were less powerful and tasks for reading header information were less numerous. OIF headers combine ASCII and binary information, and FITS headers come packaged with the data in a single file.

GEIS was also the standard format for archiving and distribution of HST data until September 1994, when the Space Telescope Data Archive and Distribution Service (ST-DADS) came online. This new system stores and distributes HST data files in machine-independent FITS format, but observers with FOC, FOS, FGS, GHRS, HSP, WF/PC-1, and WFPC2 still must convert their files to machine-dependent GEIS format as described in section 2.3.1 before using IRAF/STSDAS software (see chapter 3 in the HST Introduction) to reduce their data.

Since the selection of GEIS as HST's standard data format, FITS has added features that have dramatically increased its flexibility. In particular, FITS files can now contain multiple *image extensions*, each with its own header, size, and datatype, that allow multiple exposures to be packaged into the same file, along with associated error and data quality information. The FITS image kernel in IRAF version 2.11, released in August 1997, enables users to access FITS image extensions in ways similar to how they would access GEIS data groups.

Because of these advantages, FITS was chosen as the standard reduction and analysis format for STIS and NICMOS. The STSDAS tasks written for these instruments expect FITS files as input and produce FITS files as output. *You cannot convert STIS and NICMOS files to GEIS format.* Observers using these instruments should therefore read the following section, which explains how to work with these new FITS files.

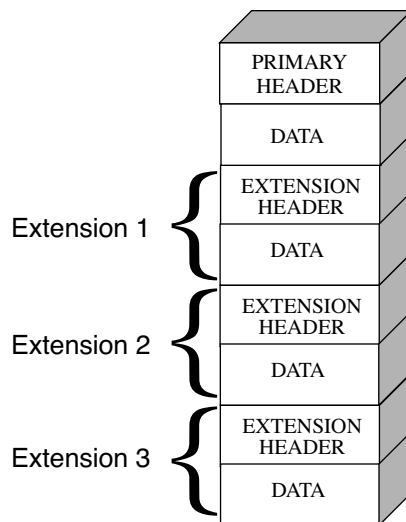
2.2 FITS File Format

Flexible Image Transport System (FITS) is a standard format for exchanging astronomical data between institutions, independent of the hardware platform and software environment. A data file in FITS format consists of a series of Header Data Units (HDUs), each containing two components: an ASCII text header and the binary data. The header contains a series of *header keywords* that describe the data in a particular HDU and the data component immediately follows the header.

The first header in a FITS file is known as the *primary header*, and any number of *extensions* can follow the primary HDU. The data unit following the primary header must contain either an image or no data at all, but each extension can contain one of several different data types, including images,

binary tables, and ASCII text tables. The value of the XTENSION keyword in the extension’s header identifies the type of data the extension contains. Figure 2.1 schematically illustrates the structure of a FITS file and its extensions.

Figure 2.1: FITS File Structure



The three-letter identifier (e.g., d0h) that follows the rootname of an HST data file (see appendix B for more on HST file names) has often been called an “extension” in the past. However, because of the potential for confusion with FITS extensions, this handbook will refer to these three-letter identifiers as “suffixes.”

2.2.1 Working with FITS Image Extensions

The FITS image kernel included in IRAF version 2.11 is designed to read and write the images in FITS extensions and their associated headers. Once IRAF has ingested a FITS image and its header, it treats the header-data pair like any other IRAF image. The following discussion describes how to specify the image extensions in FITS files that you would like to process with IRAF/STSDAS tasks and presumes that you are using IRAF 2.11 or higher. It covers how to:

- List a FITS file’s extensions.
- Access data in particular FITS extensions.

- Inherit keywords from the primary header.
- Append new extensions to existing FITS files.



Retaining the .fits at the end of every FITS file name in your file specifications will ensure that IRAF both reads and writes these images in FITS format.



If you want to work with STIS and NICMOS data, you will need to upgrade to IRAF 2.11 or higher and STSDAS 2.0.

Generating a FITS File Listing

Once you have downloaded STIS, ACS, or NICMOS FITS files from the Archive, you may want an inventory of their contents. To generate a listing of a FITS file's extensions, you can use the **catfits** task in the **tables** package. The following example, in table 2.1, illustrates the first 11 lines generated by **catfits** from a NICMOS MULTIACCUM FITS file containing images only.

The first column of a **catfits** listing gives the extension numbers. Note that the primary HDU is labeled extension number zero. The second column lists the extension type, given by the keyword XTENSION (IMAGE = image, BINTABLE = binary table, TABLE = ASCII table). The third column lists the extension name, given by the keyword EXTNAME. In STIS, ACS, and NICMOS image files, the EXTNAME values SCI, ERR, and DQ indicate science, error, and data quality images, respectively. NICMOS image files contain samples and exposure time images as well, with EXTNAME values SAMP and TIME.

Each STIS or NICMOS readout generates an image set or *imset*. STIS and ACS imsets comprise three images (SCI, ERR, DQ), while NICMOS imsets comprise five (SCI, ERR, DQ, SAMP, TIME). All images belonging to the same imset share the same integer value of the EXTVER keyword, given in the fourth column of a **catfits** listing. Several STSDAS tasks can work with entire imsets (see section 3.3.3), but most operate on individual images. See the Data Structure chapters of STIS, ACS, and NICMOS Data Handbooks for more information on the contents of imsets.

Table 2.1: NICMOS MULTIACCUM Listing from catfits

```
tt> catfits n3t501c2r_raw.fits
```

EXT#	FITSNAME	FILENAME	EXTVE	DIMENS	BITPI	OBJECT
0	n3t501c2r_raw	n3t501c2r_raw.fits			16	n3t501c2r_raw.f
1	IMAGE	SCI	1	256x256	16	n3t501c2r_raw.f
2	IMAGE	ERR	1		-32	
3	IMAGE	DQ	1		16	
4	IMAGE	SAMP	1		16	
5	IMAGE	TIME	1		-32	
6	IMAGE	SCI	2	256x256	16	
7	IMAGE	ERR	2		-32	
8	IMAGE	DQ	2		16	
9	IMAGE	SAMP	2		16	
10	IMAGE	TIME	2		-32	

Accessing FITS Images

After you have identified which FITS image extension you wish to process, you can direct an IRAF/STSDAS task to access that extension using the following syntax:

```
fitsfile.fits[extension number][keyword options][image section]
```

Note that all the bracketed information is optional. However, the only time it is *valid* to provide only a file name without further specification is when the file is a simple FITS file that contains a single image in the primary HDU.

Designation of the extension number is the most basic method of access, but it is not necessarily the most helpful. Referring to an extension's EXTNAME and EXTVER in the [keyword options] is often more convenient. If a number follows an EXTNAME, IRAF interprets the number as an EXTVER. For example, if extension number 6 holds the science image belonging to the imset with EXTVER = 2, as in the **catfits** listing on the previous page, you can specify it in two equivalent ways:

```
fitsfile.fits[6]
fitsfile.fits[sci,2]
```

Designations giving an EXTNAME without an EXTVER refer to the first extension in the file with the specified value of EXTNAME. Thus, fitsfile.fits[sci] is the same as fitsfile.fits[sci,1].

The syntax for designating image sections adheres to the IRAF standard, so in the current example the specifications

```
fitsfile.fits[6][100:199,100:299]
fitsfile.fits[sci,2][100:199,100:299]
```

both extract a 100 by 200 pixel subsection of the same science image in

```
fitsfile.fits.
```

Header Keywords and Inheritance

STIS, ACS, and NICMOS data files use an IRAF image kernel convention regarding the relationship of the primary header keywords to image extensions in the same file. In particular, IRAF allows image extensions to *inherit* keywords from the primary header under certain circumstances. When this inheritance takes place, the primary header keywords are practically indistinguishable from the extension header keywords. This feature circumvents the large scale duplication of keywords that share the same value for all extensions. The primary header keywords effectively become global keywords for all image extensions. The FITS standard does not cover or imply keyword inheritance, and while the idea itself is simple, its consequences are often complex and sometimes surprising to users.

Generally keyword inheritance is the default, and IRAF/STSDAS applications will join the primary and extension headers and treat them as one. For example, using **imheader** as follows on a FITS file will print both primary and extension header keywords to the screen:

```
c1> imheader fitsfile.fits[sci,2] long+ | page
```

Using **imcopy** on such an extension will combine the primary and extension headers in the output HDU, even if the output is going to an extension of another FITS file. Once IRAF has performed the act of inheriting the primary header keywords, it will normally turn the inheritance feature off in any output file it creates unless specifically told to do otherwise.



If you need to change the value of one of the global keywords inherited from the primary header, you must edit the primary header itself (i.e., “extension” [0]).

Keyword inheritance is not always desirable. For example, if you use **imcopy** to copy all the extensions of a FITS file to a separate output file, IRAF will write primary header keywords redundantly into each extension header. You can suppress keyword inheritance by using the **NOINHERIT** keyword in the file specification. For example:

```
im> imcopy fitsfile.fits[6][noinherit] outfile.fits
im> imcopy fitsfile.fits[sci,2,noinherit] outfile.fits
```

Both of the preceding commands will create an output file whose header contains only those keywords that were present in the original extension header. Note that in the second command, the **noinherit** specification is bracketed with the **EXTNAME** and **EXTVER** keywords and not in a separate bracket of its own, as in the first command where an absolute extension number is used. For a complete explanation of FITS file name specifications, see:

<http://iraf.noao.edu/iraf/web/docs/fitsuserguide.html>.

Appending Image Extensions to FITS Files

IRAF/STSDAS tasks that produce FITS images as output can either create new FITS files or append new image extensions to existing FITS files. You may find the following examples useful if you plan to write scripts to reduce STIS, ACS, or NICMOS data:

If the specified output file does not yet exist, a new output file is created containing only a primary HDU if no specification is appended to the output file name. For example, to copy the contents of the primary header of **fitsfile.fits** into the primary HDU of the FITS file **outfile.fits**, type the command:

```
cl> imcopy fitsfile.fits[0] outfile.fits
```

If the specified output file already exists and you want to append a new extension to it, you need to include the **APPEND** option in the output file specification. The following command appends extension **[sci,2]** of **fitsfile.fits** onto the existing file **outfile.fits**, while retaining the original **EXTNAME** and **EXTVER** of the extension—the **noinherit** specification inhibits the copying of the primary header keywords from the input file into the output extension header:

```
cl> imcopy fitsfile.fits[sci,2,noinherit] \
>>> outfile.fits[append]
```

If you want to change the EXTNAME or EXTVER of the appended extension, you can specify the new values of these keywords in the output extension, like this:

```
cl> imcopy fitsfile.fits[sci,2,noinherit] \
>>> outfile.fits[sci,3,append]
```

For obvious reasons, it is not generally advisable for two file extensions in the same FITS file to share the same EXTNAME and EXTVER values. However, if you must append an extension to an output file already containing an extension with the same EXTNAME/EXTVER pair you can do so with the DUPNAME option:

```
cl> imcopy fitsfile.fits[7] \
>>> outfile.fits[append,dupname]
```

If you need to replace an existing extension with a new output extension, you can use the OVERWRITE option as follows. Overwriting can cause a lengthy rewrite of the whole file to insert the new extension, if its size is not the same as the extension it replaces.

```
cl> imcopy fitsfile.fits[sci,2,noinherit] \
>>> outfile.fits[sci,2,overwrite]
```

2.2.2 Working with FITS Table Extensions

STIS and NICMOS use FITS tables in two basic ways. Both instruments produce association tables (see appendix B.3) listing the exposures that go into constructing a given association product. In addition, STIS provides certain spectra, calibration reference files, and time-tagged data in tabular form. Here we describe:

- How to access and read FITS table extensions.
- How to specify data arrays in FITS table cells.

This discussion assumes you are using STSDAS 2.0 or later. (The IRAF FITS kernel deals only with FITS images. The **tables** package installed with STSDAS handles FITS table extensions.)

Accessing FITS Tables

You can access data in FITS table extensions using the same tasks appropriate for any other STSDAS table, and the syntax for accessing a specific FITS table is similar to the syntax for accessing FITS images (see section 2.2.1), with the following exceptions:

- The FITS table interface does not support header keyword inheritance.
- FITS tables cannot reside in the primary HDU of a FITS file. They must reside instead in a FITS table extension, in either ASCII form (XTENSION=TABLE) or binary form (XTENSION=BINTABLE).
- If the first extension in a FITS file is a TABLE or a BINTABLE, you can access it by typing the file name with no extension specified. It is not sufficient for the table to be just the first BINTABLE or TABLE; *it must actually be the first extension.*

For example, running **catfits** on the NICMOS association table `n3tc01010_asn.fits` provides the following output:

```
fi> catfits n3tc01010_asn.fits
```

EXT#	FITSNAME	FILENAME	EXTVE ...
0	n3tc01010_asn	N3TC01010_ASN.FITS	...
1	BINTABLE	ASN	1 ...

Extension number 1 holds the association table, which has EXTNAME=ASN and EXTVER=1. You can use the **tprint** task in the STSDAS **tables** package to print the contents of this table, and the following commands are all equivalent:

```
tt> tprint n3tc01010_asn.fits
tt> tprint n3tc01010_asn.fits[1]
tt> tprint n3tc01010_asn.fits[asn,1]
```

STSDAS **tables** tasks can read both FITS TABLE and BINTABLE extensions, but they can write tabular results only as BINTABLE extensions. Tasks that write to a table in-place (i.e., **tedit**) can modify an existing FITS extension, and tasks that create a new table (i.e., **tcopy**) will create a new extension when writing to an existing FITS file. If the designated output file does not already exist, the task will create a new FITS file with the output table in the first extension. If the output file already exists, your task will append the new table to the end of the existing file; the APPEND option necessary for appending FITS image extensions is not required. As with FITS images, you can specify the EXTNAME and EXTVER of the output extension explicitly, if you want to assign them values different from those in the input HDU. You can also specify the

OVERWRITE option if you want the output table to supplant an existing FITS extension. For example, you could type:

```
tt> tcopy n3tc01010_asn.fits out.fits[3][asn,2,overwrite]
```

This command would copy the table in the first extension of `n3tc01010_asn.fits` into the third extension of `out.fits`, while reassigning it the EXTNAME/EXTVER pair `[asn,2]` and overwriting the previous contents of the extension. Note that overwriting is the only time when it is valid to specify an extension, EXTNAME, and an EXTVER in the output specification.

Specifying Arrays in FITS Table Cells

A standard FITS table consists of columns and rows forming a two-dimensional grid of cells; however, each of these cells can contain a data array, effectively creating a table of higher dimensionality. Tables containing extracted STIS spectra take advantage of this feature. Each column of a STIS spectral table holds data values corresponding to a particular physical attribute, such as wavelength, net flux, or background flux. Each row contains data corresponding to one spectral order, and tables holding echelle spectra can contain many rows. Each cell of such a spectral table can contain a one-dimensional data array corresponding to the physical attribute and spectral order of the cell.

In order to analyze tabular spectral data with STSDAS tasks other than the **sgraph** task and the **igi** package, which have been appropriately modified, you will need to extract the desired arrays from the three-dimensional table. Two new IRAF tasks, named **tximage** and **txtable**, can be used to extract the table-cell arrays. Complementary tasks, named **tiimage** and **titable**, will insert arrays back into table cells. To specify the arrays which should be extracted from or inserted into the table cells, you will need to use the *selectors* syntax to specify the desired row and column. The general syntax for selecting a particular cell is:

```
intable.fits[extension number][c:column_selector][r:row_selector]  
or  
intable.fits[keyword options][c:column_selector][r:row_selector]
```

A *column selector* is a list of column patterns separated by commas. The column pattern is either a column name, a file name containing a list of column names, or a pattern using the IRAF pattern matching syntax (type `help system.match`, for a description of the IRAF pattern matching syntax). If you need a list of the column names, you can run the **tlcol** task (type `tlcol infile.fits`).

Rows are selected according to a *filter*. The filter is evaluated at each table row, and the row is selected if the filter is true. For example, if you specify:

```
infile.fits[3][c:WAVELENGTH,FLUX][r:SPORDER=(68:70)]
```

IRAF will extract data from the table stored in the third extension of the FITS file, `infile.fits`, specifically the data from the columns labelled `WAVELENGTH` and `FLUX`, and will restrict the extraction to the rows where the spectral order (`SPORDER`) is within the range 68–70, inclusive. Alternatively, if you specify:

```
infile.fits[sci,2][c:FLUX][r:row=(20:30)]
```

IRAF will obtain data from the table stored in the FITS file extension with an `EXTNAME` of `SCI` and `EXTVER` of 2. The data will come from the column `FLUX` and be restricted to the row numbers 20–30, inclusive. Eventually, all `STSDAS` and `TABLES` tasks will be able to use row and column selection. For a complete explanation of the table selector syntax, type `help selectors`.

2.3 GEIS File Format

The HST-specific Generic Edited Information Set (GEIS) format¹ is the standard format for reducing data from FOC, FOS, FGS, GHRS, HSP, WF/PC-1, and WFPC2. All HST images in GEIS format consist of two components: a *header file* and a separate *binary data file*, both of which should reside in the same directory. GEIS header files, whose suffixes end in “h” (e.g., `w01o0105t.c1h`), consist entirely of ASCII text in fixed-length records of 80 bytes. These records contain header keywords that specify the properties of the image itself and the parameters used in executing the observation and processing the data. GEIS binary data files, whose suffixes end in “d” (e.g., `w01o0105t.c1d`), contain one or more *groups* of binary data. Each group comprises a data array followed by an associated block of binary parameters called the Group Parameter Block (GPB). The sizes and datatypes of the data arrays and group parameters in

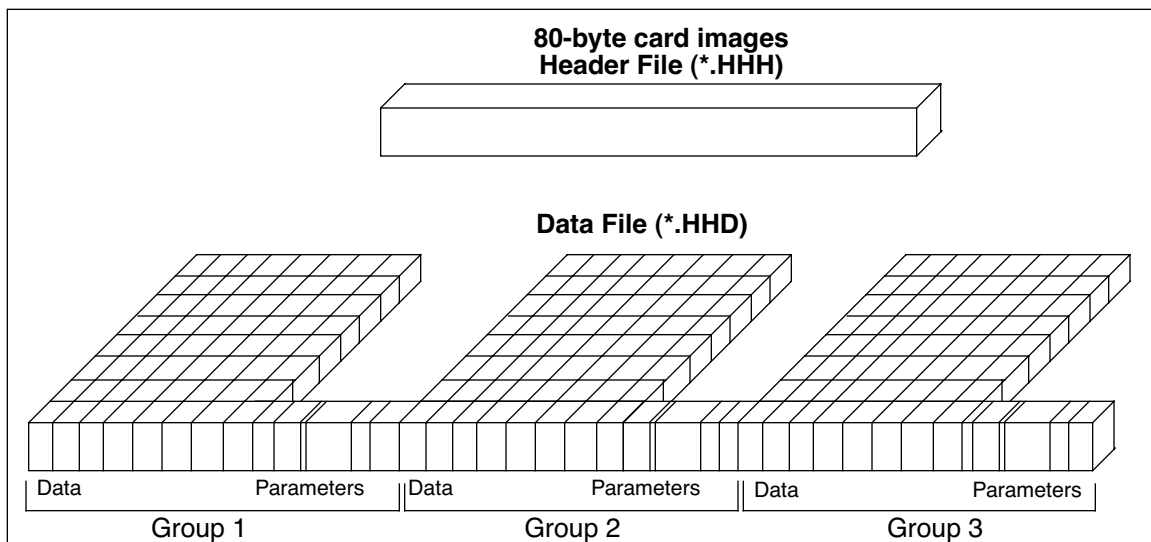
1. GEIS files are also commonly referred to as `STSDAS` images.

each group of a GEIS file are identical. Figure 2.2 depicts the structure of a GEIS data file graphically.



The binary content of GEIS files is machine dependent. Copying GEIS files directly from one platform to another (e.g., from a VAX to a Sun) may result in unreadable data.

Figure 2.2: GEIS File Structure



2.3.1 Converting FITS to GEIS

The STScI archive stores and distributes datasets from FOC, FOS, FGS, GHRS, HSP, WF/PC-1, and WFPC2 in a special archival FITS format. *We highly recommend that users convert these datasets back into their native GEIS format before working with them.* Your data must be in GEIS format for you to use many of the STSDAS software tools developed specifically for analysis of these data. It is important to use the **strfits** task found in **stsdas.fitsio** or in **tables.fitsio** to perform the conversion from archival FITS format to the GEIS format because the data-processing pipeline employs a special convention for mapping GEIS files to FITS format. While other FITS readers may be able to read portions of the data correctly, they are unlikely to reconstruct the entire data file properly.

To recreate the original multigroup GEIS file using strfits, you must first type:

```
c1> set imtype=hhh
```

This command tells IRAF to write output files in GEIS format. You then need to set the **strfits** parameters **xdimtogf** and **oldirafname** both to “yes”. For example, after you have set **imtype = hhh**, you can convert the FITS file ***_hhf.fits** into the GEIS format files ***.hhh** and ***.hhd** by typing:

```
cl> strfits *_hhf.fits "" xdim=yes oldiraf=yes
```

2.3.2 GEIS Data Groups

One of the original advantages of GEIS format noted in Section 2.1 was that it could accommodate multiple images within a single file. This feature is useful because a single HST observation often produces multiple images or spectra. For example, a single WF/PC-1 or WFPC2 exposure generates four simultaneous images, one for each CCD chip. Likewise, the FOS and GHRS obtain data in a time-resolved fashion so that a single FOS or GHRS dataset comprises many spectra—one corresponding to each readout. The data corresponding to each sub-image (for the WF/PC-1 or WFPC2) or each sub-integration (for the FOS or GHRS) are stored sequentially in the groups of a single GEIS binary data file. The header file corresponding to this data file contains the information that applies to the observation as a whole (i.e., to all the groups in the image), and the group-specific keyword information is stored in the group parameter block of each data group in the binary data file.

The *number* of groups produced by a given observation depends upon the instrument configuration, the observing mode, and the observing parameters. Table 2.2 lists the *contents* and the number of groups in the final calibrated image for the most commonly-used modes of each instrument which uses the GEIS data format.

Table 2.2: Groups in Calibrated Images, by Instrument and Mode

Instrument	Mode	Number of Groups	Description
FGS	All	7	FGS data are not reduced with IRAF and STSDAS. Therefore, FGS groups have different meaning than for the other instruments.
FOC	All	1	All FOC images have only a single group.
FOS	ACCUM	<i>n</i>	Group <i>n</i> contains accumulated counts from groups (subintegrations) 1, 2, ... <i>n</i> . The last group is the full exposure.
	RAPID	<i>n</i>	Each group is an independent subintegration with exposure time given by group parameter EXPOSURE.

Instrument	Mode	Number of Groups	Description
HSP	All	1	HSP datasets always have only a single group that represents either digital star (.d0h, .c0h), digital sky (.d1h, .c1h), analog star (.d2h, .c2h), or analog sky (.d3h, .c3h).
GHRS	ACCUM	<i>n</i>	Each group is an independent subintegration with exposure time given by group parameter EXPOSURE. If FP-SPLIT mode was used, the groups will be shifted in wavelength space. The independent subintegrations should be coadded prior to analysis.
	RAPID	<i>n</i>	Each group is a separate subintegration with exposure time given by group parameter EXPOSURE.
WF/PC-1	WF	4	Group <i>n</i> represents CCD chip <i>n</i> , e.g., group 1 is chip 1 (unless not all chips were used). Group parameter DETECTOR always gives chip used.
	PC	4	Group <i>n</i> is chip <i>n</i> + 4, e.g., group 1 is chip 5. If not all chips were used, see the DETECTOR parameter which always gives the chip used.
WFPC2	All	4	Planetary chip is group 1, detector 1. Wide Field chips are groups 2–4 for detectors 2–4. If not all chips were used, see the DETECTOR keyword.

2.3.3 Working with GEIS Files

This section briefly explains how to work with information in GEIS header and data files.

GEIS Headers

Header keyword information relevant to each group of a GEIS file resides in two places, the header file itself and the parameter block associated with the group. Because GEIS header files are composed solely of ASCII text, they are easy to print using standard Unix or VMS text-handling facilities. However, the group parameters are stored in the binary data file. To access them you need to use a task such as **imheader**, as shown in section “Printing Header Information”.

You can use the IRAF **hedit** task to edit the keywords in GEIS headers. While it is possible to edit GEIS header files using standard Unix and VMS text editors, you must maintain their standard 80-character line length. The **hedit** task automatically preserves this line length. If you need to add or delete group parameters, you can use the STSDAS **groupmod** task in the **stsdas.hst_calib.ctools** package. The STSDAS **chcalpar** task, described in more detail in the Calibration chapters for each instrument’s data

handbook, is useful for updating header keywords containing calibration switches and calibration reference files.



*Always edit headers using tasks like **hedit**, **eheader**, and **chcalpar**. Editing headers with a standard text editor may corrupt the files by creating incorrect line lengths.*

GEIS Data Files

Numerous IRAF/STSDAS tasks exist for working with GEIS images (see chapter 3 of the HST Introduction). Most of these tasks operate on only one image at a time, so you usually need to specify which group of a GEIS file is to be processed. If you do not specify a group, your task will choose the first group by default.

Specifying a Group

To specify a particular group in a GEIS file, append the desired group number in square brackets to the file name (e.g., `z2bd010ft.d0h[10]`). For example, to apply the **imarith** task to group 10 of a GEIS image, type the following (always refer to a GEIS file by its header file name, i.e. `*.??h`, even though mathematically you are operating on the data portion):

```
cl> imarith indata.hhh[10] + 77.0 outdata.hhh
```

This command will add 77.0 to the data in group 10 of the file `indata.hhh`, and will write the output to a new single-group file called `outdata.hhh`. Any operation performed on a single group of a multigroup GEIS file results in an output file containing a single group.

Specifying an Image Section

If you wish to process only a portion of an image, you can specify the image section after the group specification in the following manner:

```
cl> imarith indata.hhh[2][100:199,200:399] * 32.0 outdata.hhh
```

This command extracts a 100 by 200 pixel subsection of the image in the second group of the file `indata.hhh`, multiplies this data by a factor of 32.0, and stores the result in a new output file, `outdata.hhh`, which is a 100 by 200 pixel single group GEIS file.

Printing Header Information

As discussed in the previous section, the task **imheader** extracts and prints information about the GEIS image. This task reports the image

name, dimensions (including the number of groups), pixel type, and title of the image when it is run in default mode. For example:

```
cl> imhead indata.hhh
      indata.hhh[1/64][500][real]: INDATA[1/64]
```

The output line indicates that `indata.hhh` is a multigroup GEIS file which contains 64 groups of images, each consisting of a spectral array 500 pixels in length. The data type of the values is real (floating point). Note that since no group designation was provided, the task defaulted to the first group. To reveal more information regarding group 10, you can type:

```
cl> imhead indata.hhh[10] long+ | page
```

which will generate a long listing of both the ASCII header parameters in the `*.hhh` file and the specific group parameters for group 10 from the `*.hhd` file.

Other Group-Related Tasks

Currently, IRAF or STSDAS tasks cannot process all the groups in an input image and write the results to corresponding groups in an output image. However, there are several STSDAS tasks, particularly in the **toolbox.imgttools** and **hst_calib.ctools** packages, that simplify working with group format data. Please refer to chapter 3 and the [STSDAS User's Guide](#) for more details about working with GEIS images.

2.3.4 The "waiver" FITS format

Although “waiver” is not quite the accurate or good word for the intended purpose, for historic reasons it has stuck and will be reluctantly adopted. However, in the past, a grammatically incorrect word “waivered” had been used.

The “waiver” FITS format was developed when the HST archive needed a format to store and distribute the data products in a machine-independent medium for the community, at a time before FITS image extension was standardized. As a result, the “waiver” FITS format was adopted as a compromise.

Since, at the time, FITS could only have a single image while the HST data (in GEIS format) may have several images as multiple groups in one file, the idea is to stack the images of different groups together as a new dimension in the FITS image. As for group parameters, they are put in an ASCII table and the table becomes the first (and only) extension of the FITS file.

For example, the WFPC2 pipeline generates the science data as a GEIS file of 4 groups, each is an 800x800 image corresponding to one of the 4

detectors. When this GEIS file is converted to the “waiver” FITS file, the FITS file has an image of 800x800x4 (a three-dimensional image!) at its primary HDU. Similarly, an FOS GEIS file may have 40 groups, each group is a 1-D image (spectrum) of the size 2064. The waiver FITS file then will have one 2-D image of the size 2064x40, at its primary HDU. In the case of WFPC2, the first extension of the waiver FITS file will be an ASCII table containing 4 rows; each row corresponds to a group. The value of each group parameter is under a column named after the group parameter, i. e. the value of the group parameter CRVAL1 of the 2nd group will be at the 2nd row, under the column named “CRVAL1”. In other words, the ASCII table has as many rows as there are groups in the original GEIS file, and as many columns as group parameters.

Although, *in theory, certain* IRAF/STSDAS tasks can directly access the data in the “waiver” FITS file, e.g. to display the 2nd “group” of a WFPC2 image:

```
st.> display u67m0206r_c0f.fits[0][*,*,2]
```

will work, while *most tasks, especially those specific to HST instruments, can not*. It is therefore HIGHLY recommended that all waiver FITS files are converted back to the GEIS format, by using the task **strfits**, before further processing and analysis with IRAF/STSDAS tasks.

STSDAS Basics

In this chapter. . .

3.1 Navigating STSDAS / 3-2
3.2 Displaying HST Images / 3-4
3.3 Analyzing HST Images / 3-9
3.4 Displaying HST Spectra / 3-20
3.5 Analyzing HST Spectra / 3-24
3.6 References / 3-37

The Space Telescope Science Data Analysis System (STSDA) is the software system for calibrating and analyzing data from the Hubble Space Telescope. The package contains programs—called *tasks*—that perform a wide range of functions supporting the entire data analysis process, from reading tapes, through reduction and analysis, to producing final plots and images. This chapter introduces the basics of STSDAS, showing you how to display your data, leading you through some simple data manipulations, and pointing you towards more sophisticated tasks, some of which are described in the instrument data handbooks.

STSDAS is layered on top of the [Image Reduction and Analysis Facility \(IRAF\)](#) software developed at the National Optical Astronomy Observatory (NOAO). Any task in IRAF can be used in STSDAS, and the software is portable across a number of platforms and operating systems. To exploit the power of STSDAS effectively, you need to know the basics of IRAF. If you are not already familiar with IRAF, consult the IRAF Primer in Appendix A before reading further.

3.1 Navigating STSDAS

The tasks in STSDAS are far too numerous and complicated to describe comprehensively in this volume. Instead, we will show you where to find the STSDAS tasks appropriate for handling certain jobs. You can refer to online help or the *STSDAS User's Guide* for details on how to use these tasks. Some useful online help commands are:

- `help task` - provides detailed descriptions and examples of each task.
- `help package` - lists the tasks in a given package and their functions.
- `describe task` - provides a detailed description of each task.
- `example task` - provides examples of each task.
- `apropos word` - searches the online help database for tasks relating to the specified word (see figure A.4).

3.1.1 STSDAS Structure

STSDAS is structured so that related tasks are grouped together as packages. For example, tasks used in the calibration process can be found in the **hst_calib** package and tasks used for image display and plotting can be found in the **graphics** pack. Figure 3.1 shows the STSDAS package structure. Note that IRAF version 2.11 must be installed on your system in order for you to use STSDAS 2.0 and TABLES version 2.0 or higher

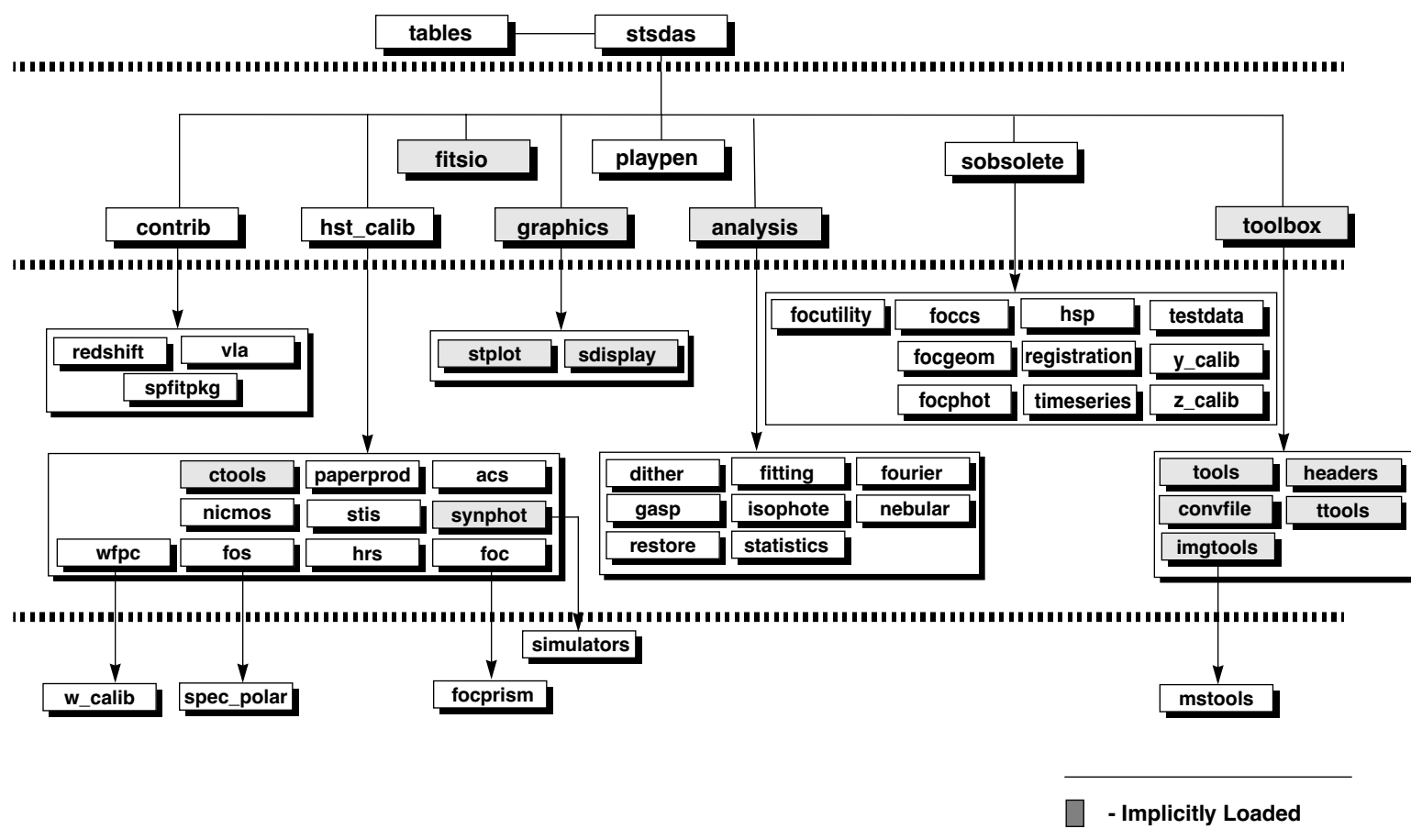
3.1.2 Packages of General Interest

Images

Both IRAF and STSDAS contain a large number of tasks that work with HST images. Some of the packages you should investigate are:

- **images**: This package includes general tasks for copying (**imcopy**), moving (**imrename**), and deleting (**imdelete**) image files. These tasks operate on both the header and data portions of the image. The package also contains a number of general purpose tasks for operations such as rotating and magnifying images.
- **stdas.toolbox.imgtools**: This package contains general tools for working with multigroup GEIS images, including tasks for working with masks, and general purpose tasks for working with the pixel data, such as an interactive pixel editor (**pixedit**).

Figure 3.1: STSDAS Version 2.3 Package Structure



- **stdas.toolbox.imgtools.mstools**: This package contains tools for working with FITS image extensions, in particular STIS and NICMOS image sets (imsets).
- **stdas.analysis**: This package contains general tasks for image analysis, such as Fourier analysis and dither.

Tables

Several of the analysis packages in STSDAS, including calibration pipeline tasks, create output files in STSDAS table format, which is a binary row-column format, or in FITS binary table format. (ASCII-format tables are also supported, for input only.) The *STSDAS User's Guide* describes the STSDAS table format in detail. Tasks in the **ttools** package or in the external **tables** package can be used to read, edit, create, and manipulate tables. For example:

- **tread** displays a table, allowing you to move through it with the arrow keys.
- **tprint** displays a table.
- **tcopy** copies tables.
- **tedit** allows you to edit a table.

Many other tasks in **ttools** perform a variety of other functions. See the online help for details.

3.2 Displaying HST Images

This section will be of interest primarily to observers whose datasets contain two-dimensional images, as it explains:

- How to display images in IRAF using the **display** task.
- How to display subsections of images.

Observers viewing WF/PC-1 and WFPC2 data may wish to remove cosmic rays before displaying their data. The FOC photon-counting hardware does not detect cosmic rays as easily as CCDs, the NICMOS pipeline automatically removes cosmic rays from MULTIACCUM observations, and the STIS pipeline automatically removes cosmic rays from CR-SPLIT association products.

3.2.1 The display Task

The most general IRAF task for displaying image data is the **display** task, the best choice for a first look at HST imaging data. To display an image, you need to:

1. Start an image display server, such as SAOimage, in a separate window from your IRAF session, either from a different xterm window or as a background job before starting IRAF. To start SAOimage, type the following:

```
saoimage &
```

2. Load the **images.tv** package from the window where you're running IRAF:

```
cl> images
im> tv
```



Several different display servers, including SAOimage, ds9 (the next generation of SAOimage), and Ximtool, can be used with IRAF. ds9 may be retrieved from <http://hea-www.harvard.edu/IRAF/ds9/>. Ximtool may be retrieved from <ftp://iraf.noao.edu/iraf/x11iraf/>.

3. Display the image with the IRAF **display** task, using the syntax appropriate for the file format (Chapter 2 explains how to specify GEIS groups and FITS extensions):

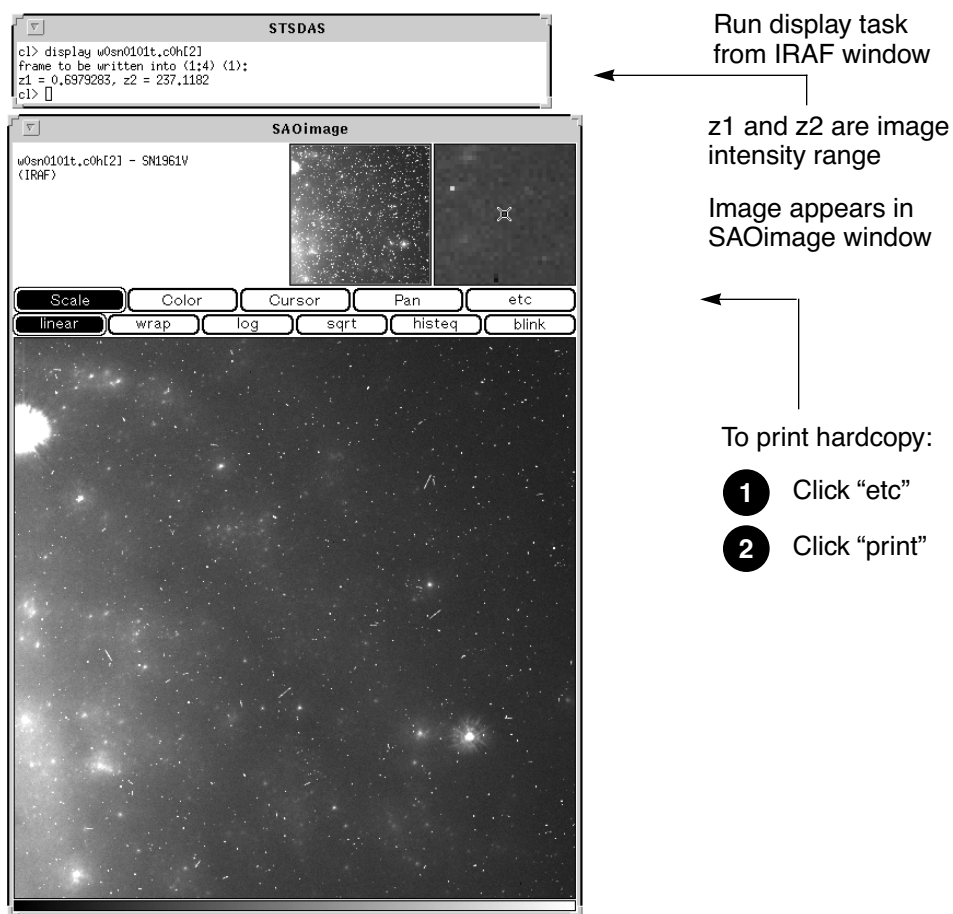
```
tv> display fname.c0h[2] 1 (GEIS group 2)
tv> display fname.fits[11] 1 (FITS extension 11)
tv> display fname.fits[sci,3] 1 (FITS extension sci,3)
```

Note that when using **display** or any other task on GEIS images, you do not need to specify a group; the first group is the default. However, when working with FITS files you must specify an extension, unless the FITS file contains only a single image in the primary data unit and has no extensions. figure 3.2 shows how to display group two of a WF/PC-1 image.



If you want to display all four chips of a WF/PC-1 or WFPC2 image simultaneously, you can create a mosaic with the STSDAS `wmosaic` task in the `hst_calib.wfpc` package. Type `help wmosaic` for details.

Figure 3.2: Displaying an Image



Modifying the Display

There are two ways to adjust how your image is displayed:

- Use the SAOimage command buttons that control zooming, panning, etc.
- Reset the **display** task parameters.

Once an image appears in your SAOimage window, you can use the SAOimage commands displayed near the top of the image window to manipulate or print your image. The *SAOimage Users Guide* describes

these commands, although most are fairly intuitive. Just click on the buttons to scale, pan, or print the image, or to perform other commonly-used functions. On-line help is also available at the system level: type `man saimage` in Unix or `help saimage` in VMS.

The example in figure 3.2 shows how you should display an image for a first look. By default, **display** automatically scales the image intensity using a sampling of pixels throughout the image. During your first look, you may want to experiment with the scaling using the `zscale`, `zrange`, `z1` and `z2` parameters. The `zscale` parameter toggles the autoscaling. Setting `zscale-` and `zrange+` tells the task to use minimum and maximum values from the image as the minimum and maximum intensity values. To customize your minimum and maximum intensity display values, set `zscale-`, `zrange-`, `z1` to the minimum value and `z2` to the maximum value that you want displayed. For example:

```
im> disp w0mw0507v.c0h 1 zrange- zscale- z1=2.78 z2=15.27
```

Notice in figure 3.2 that when you run **display**, the task shows you the `z1` and `z2` values that it calculates. You can use these starting points in estimating reasonable values for the minimum and maximum intensity display parameters.¹

If you want to display an image with greater dynamic range, you may prefer to use logarithmic scaling. However, the log scaling function in SAOimage divides the selected intensity range into 200 linearly spaced levels before taking the log. The resulting intensity levels are rendered in a linear rather than logarithmic sense. You can often obtain better results if you create a separate logarithmic image to display. One way to create a logarithmic image is with the **imcalc** task:

```
im> imcalc x2ce0502t.c1h x2ce0502t.hhh "log10(im1+1.0)"
```

If the peak pixel in your original image contained 2000 counts, for example, you would then display the logarithmic image with `z1=0` and `z2=3.3`.

Otherwise, the user can simply do:

```
im> display x2ce0502t.c1h ztrans=log
```

1. Type `help display` within IRAF to obtain more information about these parameters.

The image display buffer can also be adjusted in IRAF by setting the `stdimage` parameter. For example,

```
im> set stdimage = imt 2048
```

will allow a larger image to be displayed without losing the borders.

3.2.2 Working with Image Sections

Sometimes you may want to display only a portion of an image, using the syntax for specifying image sections discussed in chapter 2. Your specified pixel range should give the starting point and ending point, with a colon separating the two. List the horizontal (x axis) range first, followed by the vertical (y axis) range. For example, to specify a pixel range from 101 to 200 in the x direction and all pixels in the y direction from group three of a GEIS format image:

```
tv> display image.hhh[3][101:200,*] 1
```

To specify the same pixel range in the second SCI extension of a NICMOS FITS image:

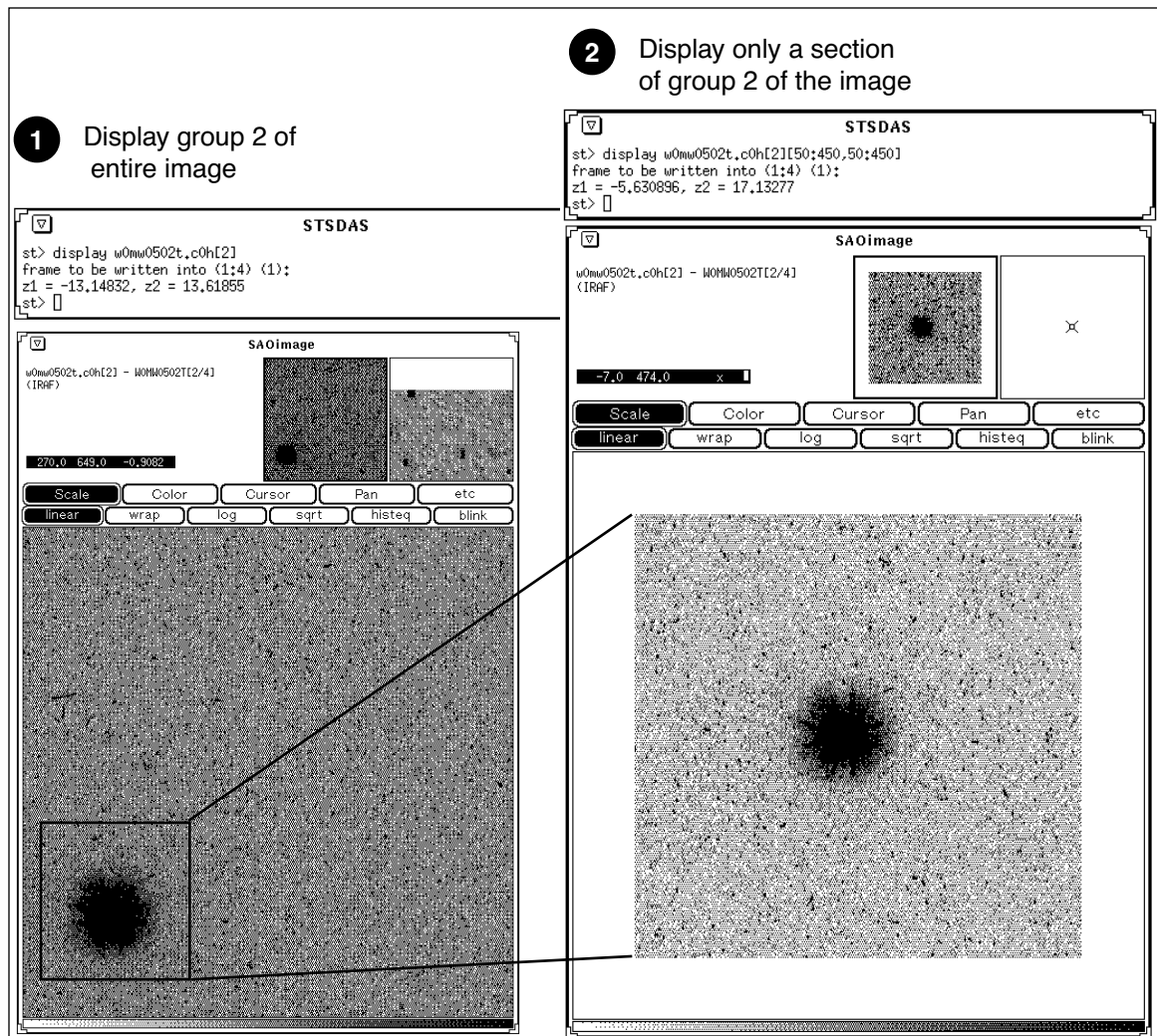
```
tv> display image.fits[sci,2][101:200,*] 1
```



If you specify both a group and an image section of a GEIS file, the group number must come first. When displaying sections of FITS image extensions, you must specify the extension, which also comes before the image section

Figure 3.3 shows examples of displaying an image and an image section.

Figure 3.3: Displaying Sections and Groups of an Image



3.3 Analyzing HST Images

This section describes methods for using STSDAS and IRAF to work with two-dimensional image data from HST. Subjects include:

- Relating your image to sky coordinates.
- Examining and manipulating your image.
- Working with STIS, ACS, and NICMOS imsets.
- Converting counts to fluxes.

3.3.1 Basic Astrometry

This section describes how to determine the orientation of an HST image and the RA and Dec of any pixel or source within it, including:

- Tasks that supply positional information about HST images.
- Methods for improving your absolute astrometric accuracy.

Positional Information

The header of every calibrated HST two-dimensional image contains a linear astrometric plate solution, written in terms of the standard FITS astrometry header keywords: CRPIX1, CRPIX2, CRVAL1, CRVAL2, and the CD matrix—CD1_1, CD1_2, CD2_1, and CD2_2. IRAF/STSDAS tasks can use this information to convert between pixel coordinates and RA and Dec. Two simple tasks that draw on these keywords to relate your image to sky coordinates are:

- **disconlab**: Displays your image with a superimposed RA and Dec grid. Simply open an SAOimage window and type, for example:

```
sd> disconlab n3tc01a5r_cal.fits[1]
```

- **xy2rd**: Translates x and y pixel coordinates to RA and Dec. (The task **rd2xy** inverts this operation.) SAOimage displays the current x,y pixel location of the cursor in the upper-left corner of the window. To find the RA and Dec of the current pixel, you supply these coordinates to **xy2rd** by typing

```
sd> xy2rd n3tc01a5r_cal.fits[1] x y
```

Table 3.1 lists some additional tasks that draw on the standard astrometry keywords.

Observers should be aware that these tasks do not correct for geometric distortion. Only FOC images currently undergo geometric correction during standard pipeline processing (the .c0h/.c0d and .c1h/.c1d FOC images have been geometrically corrected); STIS images will be geometrically corrected in the pipeline once suitable calibration files are in hand. If you need precise relative astrometry, you should use an instrument-specific task that accounts for image distortion, such as the **metric** task for WF/PC-1 and WFPC2 images.



Do not use tasks like `rimcursor` or `xy2rd` directly on WF/PC-1 or WFPC2 images if you require accurate relative positions. WF/PC-1 and WFPC2 pipelines do not correct for geometric distortions which will affect the accuracy of relative positions. Both `wmosaic` and `metric`, found in the `stsdas.hst_calib.wfpc` package, correct for this distortion.

Table 3.1: Additional IRAF and STSDAS Astrometry Tasks

Task	Purpose
compass	Plot north and east arrows on an image.
north	Display the orientation of an image based on keywords.
rimcursor	Determine RA and Dec of a pixel in an image.
wcscoords	Use WCS ¹ to convert between IRAF coordinate systems.
wcslab	Produce sky projection grids for images.

1. World Coordinate System (WCS). Type “help specwcs” at the IRAF prompt for details.

Improving Astrometric Accuracy

Differential astrometry (measuring a position of one object relative to another in an image) is easy and relatively accurate for HST images, while absolute astrometry is more difficult, owing to uncertainties in the locations of the instrument apertures relative to the Optical Telescope Assembly (OTA or V1) axis and the inherent uncertainty in Guide Star positions. However, if you can determine an accurate position for any single star in your HST image, then your absolute astrometric accuracy will be limited only by the accuracy with which you know that star’s location and the image orientation.

If there is a star on your image suitable for astrometry, you may wish to extract an image of the sky around this star from the Digitized Sky Survey and measure the position of that star using, for example, the GASP software (described in the *STSDAS User’s Guide*). These tools provide an absolute positional accuracy of approximately 0″.7. Contact the Help Desk for assistance (send E-mail to help@stsci.edu).

3.3.2 Examining and Manipulating Image Data

This section describes **implot** and **imexamine**, two basic IRAF tools for studying the characteristics of an image, and table 3.3 lists some useful IRAF/STSDAS tasks for manipulating images.

implot

The IRAF **implot** task (in the **plot** package) allows you to examine an image interactively by plotting data along a given *line* (*x* axis) or *column* (*y* axis). When you run the task, a large number of commands are available in addition to the usual cursor mode commands common to most IRAF plotting tasks. A complete listing of commands is found in the on-line help, but the most commonly used are listed in table 3.2. Figure 3.4 shows an example of how to use the **implot** task.

Table 3.2: Basic implot Commands








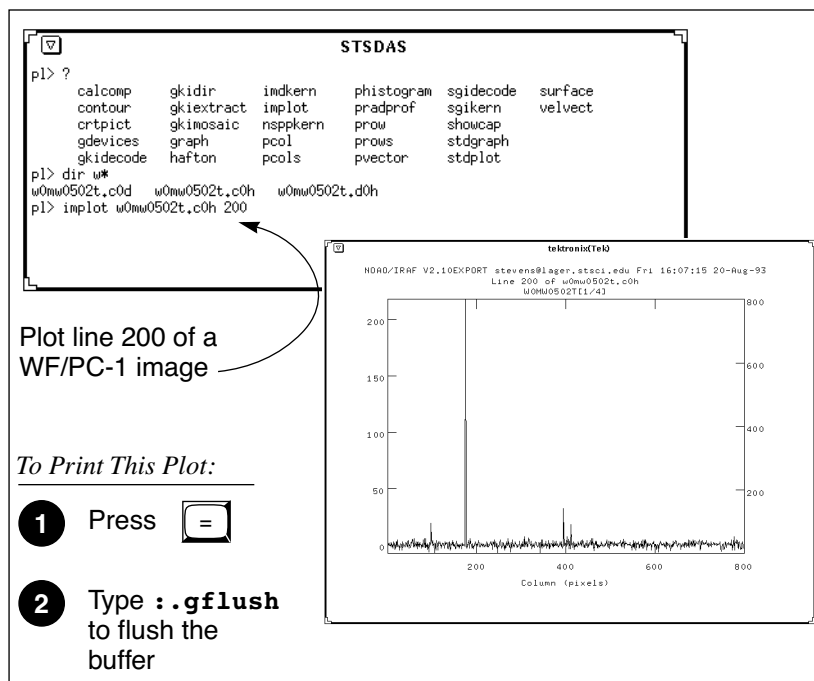
Keystroke	Command
 ?	Display on-line help.
 L	Plot a line.
 C	Plot a column.
 Q	Quit implot.
 J	Move down.
 K	Move up.
 Space	Display coordinates and pixel values.

Figure 3.4: Plotting Image Data with implot



imexamine

The IRAF **imexamine** task (in the **images.tv** package) is a powerful task that integrates image display with various types of plotting capabilities. Commands can be passed to the task using the image display cursor and the graphics cursor. A complete description of the task and its usage are provided in the online help, available from within the IRAF environment by typing `help imexamine`.

Table 3.3: Image Manipulation Tasks

Task	Package	Purpose
boxcar	images.imfilter	Boxcar smooth a list of images
gcombine	stdas.toolbox.imgtools	Combine images using various algorithms and rejection schemes
gcopy	stdas.toolbox.imgtools	Copy GEIS multigroup images
geomap	images.immatch	Compute a coordinate transformation
geotran	images.immatch	Resample an image based on geomap output
grlist	stdas.graphics.stplot	List of file names of all groups of a GEIS image (to make @lists)
gstatistics	stdas.toolbox.imgtools	Compute image statistics ¹
imcalc	stdas.toolbox.imgtools	Perform general arithmetic on GEIS images ^a
imedit	images.tv	Fill in regions of an image by interpolation
imexamine	images.tv	Examine images using display, plots, and text (see “imexamine” on page 3-13)
implot	plot	Plot lines and columns of images (see “implot” on page 3-12)
magnify	images.imgeom	Magnify an image
msarith	stdas.toolbox.mstools	Performs basic arithmetic on STIS and NICMOS imsets
mscombine	stdas.toolbox.mstools	Extension of gcombine for STIS and NICMOS imsets
msstatistics	stdas.toolbox.mstools	Extension of gstatistics for STIS and NICMOS imsets
newcont	stdas.graphics.stplot	Draw contours of two-dimensional data
pixcoord	stdas.hst_calib.wfpc	Compute pixel coordinates of stars in a GEIS image
plcreate	xray.ximages	Create a pixel list from a region file (e.g., from SAOimage)
rotate	images.imgeom	Rotate an image
saodump	stdas.graphics.sdisplay	Make image and colormap files from SAOimage display
siaper	stdas.graphics.stplot	Plot science instrument apertures of HST

1. Will process all groups of a multigroup GEIS file.

3.3.3 Working with STIS, ACS, and NICMOS Imsets

STIS, ACS, and NICMOS data files contain groups of images, called imsets, associated with each individual exposure. A STIS or ACS imset comprises SCI, ERR, and DQ images, which hold science, error, and data quality information. A NICMOS imset, in addition to its SCI, ERR, and DQ images, also contains TIME and SAMP images recording the integration time and number of samples corresponding to each pixel of the SCI image. See the STIS, ACS, and NICMOS Data Structures chapters for more details on imsets.

Here we describe several STSDAS tasks, located in the **stsdas.toolbox.imgtools.mstools** package, that have been designed to work with imsets as units and to deconstruct and rebuild them.

msarith

This tool is an extension of the IRAF task **imarith** to include error and data quality propagation. The **msarith** task supports the four basic arithmetic operations (+, -, *, /) and can operate on individual or multiple imsets. The input operands can be either files or numerical constants; the latter can appear with an associated error, which will propagate into the error array(s) of the output file. Table 3.4 below shows how this task operates on the SCI, ERR, and DQ images in a STIS, ACS, or NICMOS imset, as well as the additional TIME and SAMP images belonging to NICMOS imsets:

Table 3.4: Task msarith Operations

Operation	Operand2	SCI	ERR	DQ	TIME	SAMP
ADD	file	op1+op2	$\sqrt{\sigma_1^2 + \sigma_2^2}$	OR	T1+T2	S1+S2
SUB	file	op1-op2	$\sqrt{\sigma_1^2 + \sigma_2^2}$	OR	T1	S1
MULT	file	op1*op2	$(op1 \times op2) \sqrt{(\sigma_1/op1)^2 + (\sigma_2/op2)^2}$	OR	T1	S1
DIV	file	op1/op2	$(op1/op2) \sqrt{(\sigma_1/op1)^2 + (\sigma_2/op2)^2}$	OR	T1	S1
ADD	constant	op1+op2	$\sqrt{\sigma_1^2 + \sigma_2^2}$
SUB	constant	op1-op2	$\sqrt{\sigma_1^2 + \sigma_2^2}$
MULT	constant	op1*op2	$(op1 \times op2) \sqrt{(\sigma_1/op1)^2 + (\sigma_2/op2)^2}$...	T1*op2	...
DIV	constant	op1/op2	$(op1/op2) \sqrt{(\sigma_1/op1)^2 + (\sigma_2/op2)^2}$...	T1*op2	...

In table 3.4, the first operand (op1) is always a file, and the second operand (op2) can be either a constant or a file. The ERR arrays of the input files (σ_1 and σ_2) are added in quadrature. If the constant is given with an error (σ_2), the latter is added in quadrature to the input ERR array. Note that in table 3.4 the pixels in the SCI images are in counts, but **msarith** can also operate on count rates.

mscombine

This task allows you to run the STSDAS task **gcombine** on STIS, ACS, and NICMOS data files. It divides each imset into its basic components (SCI, ERR, and DQ, plus SAMP and TIME for NICMOS) to make them digestible for **gcombine**. The SCI extensions become the inputs proper to the underlying **gcombine** task, and the ERR extensions become the error maps. The DQ extensions are first combined with a user-specified Boolean mask allowing selective pixel masking and then fed into the data quality maps. If scaling by exposure time is requested, the exposure times of each

imset are read from the header keyword PIXVALUE in the TIME extensions.

Once **gcombine** has finished, **mscombine** then reassembles the individual output images into imsets and outputs them as one STIS, ACS, or NICMOS data file. The output images and error maps from **gcombine** form the SCI and ERR extensions of the output imset. The DQ extension will be a combination of the masking operations and the rejection algorithms executed in **gcombine**. For NICMOS, the TIME extension will be the sum of the TIME values from the input files minus the rejected values, divided on a pixel-by-pixel basis by the number of valid pixels in the output image. The final TIME array will be consistent with the output SCI image (average or median of the science data). The SAMP extension for NICMOS is built from all the input SAMP values, minus the values discarded by masking or rejection.

msstatistics

This tool is an extension of **gstatistics** in the STSDAS package, which is in turn an extension of **imstatistics**. The main novelty is the inclusion of the error and data quality information included with STIS, ACS, and NICMOS images in computing statistical quantities. In addition to the standard statistical quantities (min, max, sum, mean, standard deviation, median, mode, skewness, kurtosis), two additional quantities have been added to take advantage of the error information: the weighted mean and the weighted variance of the pixel distribution. If x_i is the value at the i -th pixel, with associated error σ_i , the weighted mean and variance used in the task are:

$$\langle x \rangle_w = \frac{\sum_i \frac{x_i}{\sigma_i \times \sigma_i}}{\sum_i \frac{1}{\sigma_i \times \sigma_i}}$$

and:

$$\langle \sigma \rangle_w^2 = \frac{1}{\sum_i \frac{1}{\sigma_i \times \sigma_i}}$$

The data quality information carried by the STIS, ACS, or NICMOS file is used to reject pixels in the statistical computation. Users can supply additional masks to reject objects or regions from the science arrays.

mssplit and msjoin

The **mssplit** task extracts user-specified imsets from a STIS, ACS, or NICMOS data file and copies them into separate files. Each output file contains a single imset along with the primary header of the original file. You might find this task useful for reducing the size of a STIS, ACS, or

NICMOS file containing many imsets or for performing analysis on a specific imset. The **msjoin** task inverts the operation of **mssplit**: it assembles separate imsets into a single data file.

There are additional tasks in this package for deleting and sorting imsets, as well as tasks for addressing a specific image class within an imset.

3.3.4 Photometry

Included in this section are:

- A list of IRAF/STSDAS tasks useful for determining source counts.
- Instructions on how to use header keyword information to convert HST counts to fluxes or magnitudes.
- A brief description of **synphot**, the STSDAS synthetic photometry package.

IRAF and STSDAS Photometry Tasks

The following are some useful IRAF/STSDAS packages and tasks for performing photometry on HST images:

- **apphot**: aperture photometry package.
- **daophot**: stellar photometry package useful for crowded fields.
- **isophote**: package for fitting elliptical isophotes.
- **imexamine**: performs simple photometry measurements.
- **imstat**: computes image pixel statistics.
- **imcnts**: sums counts over a specified region, subtracting background.
- **plcreate**: creates pixel masks.

Consult the online help for more details on these tasks and packages. The document “Photometry using IRAF” by Lisa A. Wells, provides a general guide to performing photometry with IRAF; it is available through the IRAF web page:

<http://iraf.noao.edu/docs/photom.html>



*The **apphot** package allows you to measure fluxes within a series of concentric apertures. This technique can be used to determine the flux in the wings of the PSF, which is useful if you wish to estimate the flux of a saturated star by scaling the flux in the wings of the PSF to an unsaturated PSF.*

Converting Counts to Flux or Magnitude

All calibrated HST images record signal in units of counts or Data Numbers (DN)²—NICMOS data is DN s⁻¹. The pipeline calibration tasks do not alter the units of the pixels in the image. Instead they calculate and write the inverse sensitivity conversion factor (PHOTFLAM) and the ST magnitude scale zero point (PHOTZPT) into header keywords in the calibrated data. WF/PC-1 and WFPC2 observers should note that the four chips are calibrated individually, so these photometry keywords belong to the group parameters for each chip.

For all instruments other than NICMOS, PHOTFLAM is defined to be the *mean* flux density F_λ in units of erg cm⁻² s⁻¹ Å⁻¹ that produces 1 count per second in the HST observing mode (PHOTMODE) used for the observation. If the F_λ spectrum of your source is significantly sloped across the bandpass or contains prominent features, such as strong emission lines, you may wish to recalculate the inverse sensitivity using **synphot**, described below. WF/PC-1 observers should note that the PHOTFLAM value calculated during pipeline processing does not include a correction for temporal variations in throughput owing to contamination buildup. Likewise, FOC observers should note that PHOTFLAM values determined by the pipeline before May 18, 1994 do not account for sensitivity differences in formats other than 512 x 512.

To convert from counts or DN to flux in units of erg cm⁻² s⁻¹ Å⁻¹, multiply the total number of counts by the value of the PHOTFLAM header keyword and divide by the value of the EXPTIME keyword (exposure time). You can use the STSDAS task **imcalc** to convert an entire image from counts to flux units. For example, to create a flux-calibrated output image `outimg.fits` from an input image `inimg.fits[1]` with header keywords PHOTFLAM = 2.5E-18 and EXPTIME = 1000.0, you could type:

```
st> imcalc inimg.fits[1] outimg.fits "im1*2.5E-18/1000.0"
```

Calibrated NICMOS data are in units of DN s⁻¹, so the PHOTFLAM values in their headers are in units of erg cm⁻² Å⁻¹. You can simply multiply these images by the value of PHOTFLAM to obtain fluxes in units of erg cm⁻² s⁻¹ Å⁻¹. NICMOS headers also contain the keyword PHOTFNU in units of Jy s. Multiplying your image by the PHOTFNU value will therefore yield fluxes in Janskys.

2. Except for 2-D rectified STIS images, which are in units of I_λ .



If your HST image contains a source whose flux you know from ground based measurements, you may choose to determine the final photometry of your HST image from the counts observed for this source.

To convert a measured flux F , in units of $\text{erg cm}^{-2} \text{s}^{-1} \text{\AA}^{-1}$, to an ST magnitude, plug it into the following equation:

$$m = -2.5 \times \log_{10}(F) + \text{PHOTZPT}$$

where the value of the PHOTZPT keyword is the zero point of the ST magnitude scale. The zero point of the ST magnitude system has always been and probably always will be equal to -21.10 , a value chosen so that Vega has an ST magnitude of zero for the Johnson V passband (see Koornneef et al., 1986; Horne, 1988; and the *Synphot Users Guide*).

synphot

The STSDAS synthetic photometry package, called **synphot**, can simulate HST observations of astronomical targets with known spectra. It contains throughput curves of all HST optical components, such as mirrors, filters, gratings, apertures, and detectors, and can generate passband shapes for any combination of these elements. It can also generate synthetic spectra of many different types, including stellar, blackbody, power-law and H II region spectra, and can convolve these spectra with the throughputs of HST's instruments. You can therefore use it to compare results in many different bands, to cross-calibrate one instrument with another, or to relate your observations to theoretical models.

One useful application of **synphot** is to recalculate the value of PHOTFLAM for a given observation using the latest calibration files. For example, to recalculate PHOTFLAM for an FOC observation, you could use the **calcphot** task in **synphot** as follows:

```
sy> calcphot foc,f/96,x96z1rg,f501n 'unit(1,flam)' counts
```

The first argument to **calcphot** gives the instrument and its configuration, in this case the FOC $f/96$ camera in full zoomed format with the F501 filter. (See the **obsmode** task in **synphot** and the *Synphot User's Guide* for help with these observation-mode keywords.) The second tells the task to model a flat F_{λ} spectrum having unit flux, and the third tells the task to produce output in units of counts per second. After you run **calcphot**, its **result** parameter will contain the count rate expected from the FOC, given this configuration and spectrum. The PHOTFLAM

keyword, defined to be the flux required to produce one count per second, simply equals the reciprocal of this value, which you can print to the screen by typing `=1./calcpHOT.result` at the IRAF prompt.

Please see the *Synphot User's Guide* for more details on this package, and see appendix A for information on getting the **synphot** dataset, which is not included with STSDAS.

3.4 Displaying HST Spectra

This section shows how to plot your HST spectra for a quick first look and how to generate hardcopies of your plots. Because the STIS data format differs from that of FOS and GHRS, we will discuss STIS data separately.

3.4.1 FOS and GHRS Spectra

Before you work with FOS and GHRS data within STSDAS, you will want to convert the FITS files you received from the Archive into GEIS format (see section 2.3.1 for instructions). After conversion, the `.c1h` file will hold the calibrated flux values for each pixel, the `.c0h` file will hold the corresponding wavelengths, and the `.c2h` file will hold the propagated statistical errors.

Each group of an FOS or GHRS GEIS file contains the results of a separate subintegration. FOS readouts taken in ACCUM mode are cumulative, so the last group contains the results of the entire integration. In contrast, GHRS readouts and FOS readouts in RAPID mode are independent. If you want to see the results of an entire GHRS FP-SPLIT integration, you will need to align and coadd the spectra in the groups of the GHRS file. You can also combine all the groups in an FOS or GHRS data file, without wavelength alignment, using the **rcombine** task in the **hst_calib.ctools** package. See online help for details.

The STSDAS task **sgraph** (in the **graphics.stplot** package) can plot the contents of a single GEIS group. For example, if you want to see group 19 of the calibrated FOS spectrum with rootname `y3b10104t` you can type

```
st> sgraph y3b10104t.c1h[19]
```

Given an input flux image (`.c1h`), the task **fwplot** (in the **hst_calib.ctools** package) will look for the corresponding wavelength (`.c0h`) file and plot flux versus wavelength. If requested, it will also look

for the error (.c2h) file and plot the error bars. To see a plot of the same spectrum as above, but with a wavelength scale and error bars, type

```
st> fwplot y3b10104t.c1h[19] plterr+
```

If you ever need to plot the contents of multiple groups offset from one another on the same graph, you can use the **grspec** task in the **graphics.stplot** package. For example, to plot groups 1, 10, and 19 of a given flux file, you can type

```
st> grspec y3b10104t.c1h 1,10,19
```

Note that **grspec** expects group numbers to be listed as a separate parameter, rather than enclosed in the standard square brackets.

3.4.2 STIS Spectra

STIS data files retrieved from [the Archive](#) can contain spectra in two different forms: as long-slit spectral images in FITS IMAGE extensions or as extracted echelle spectra in FITS BINTABLE extensions.

You can use **sgraph** to plot STIS long-slit spectra by specifying the image section that contains the spectrum. For example, to plot the entire *x* range of the calibrated two-dimensional spectrum in the first extension of the file `o43ba1bnm_x2d.fits`, averaging rows 100 through 1000, you would type

```
st> sgraph o43ba1bnm_x2d.fits[1][*,100:1000]
```

Displaying the long-slit spectral image using the **display** task (see section 3.2.1 in the HST Introduction) allows you to see the range of your spectrum in *x* and *y* pixel space, so you can choose a suitable image section for plotting.

To plot STIS spectra in BINTABLE extensions, you first need to understand how STIS spectra are stored as binary arrays in FITS table cells. Chapter 2 (section 2.2.2) discusses this format and describes the *selectors* syntax used to specify these data arrays. Each row of a STIS echelle table contains a separate spectral order, and each column contains data of a certain type, such as WAVELENGTH data or FLUX data. To specify a particular array, you must first type the file name, then the extension containing the BINTABLE, then the column selector, then the row selector. For example, to select the WAVELENGTH array

corresponding to spectral order 80 of the echelle spectrum in extension 4 of `stis.fits`, you would specify the file as:

```
stis.fits[4][c:WAVELENGTH][r:sporder=80]
```

The **sgraph** task and the **igi** plotting package, to be discussed below, both understand the *selectors* syntax. In particular, if you wanted to plot the flux versus wavelength in STIS echelle order 80, you could type:

```
st> sgraph "stis.fits[4][r:sporder=80] WAVELENGTH FLUX"
```

Remember to include the quotation marks. Otherwise, **sgraph** will complain about too many positional arguments. Note also that **sgraph** understands only row selector syntax; columns are chosen by name.

The STIS-specific **echplot** task is particularly useful for browsing STIS echelle spectra. It can plot single spectral orders, overplot multiple orders on a single plot, or plot up to four orders in separate panels on the same page. For example, to overplot the orders contained in rows two through four and row six on a single page:

```
cl> echplot "stis_xld.fits[1][r:row=(2:4,6)]" output.igi \
>>> plot_style=m
```


Note that the `plot_style` parameter governs how the spectral orders are plotted. The `plot_style` values **s**, **m**, and **p** plot one order per page, several orders on a single plot, and one order per panel, respectively. The default brightness unit is calibrated FLUX, although you can specify other quantities (e.g., NET counts) using the `flux_col` parameter. See the online help for details.

3.4.3 Producing Hardcopy

This section shows how to generate hardcopies of plots directly and describes **igi**, the Interactive Graphics Interpreter available in STSDAS.

Direct Hardcopies

To print a quick copy of the displayed plot:

1. Type `=gcur` in the command window (where your CL prompt is located).
2. Move the cursor to any location in the graphics window.
3. Press  to write the plot to the graphics buffer.
4. Type `q` to exit graphics mode.
5. At the cl prompt, type `gflush`.



Plots will be printed on the printer defined by the IRAF environment variable `stdplot`. Type `show stdplot` to see the current default printer; use `set stdplot = printer_name` to set the default printer.

The PostScript kernel **psikern** allows you to create PostScript files of your IRAF/STSDAS plots. For example, setting the `device` parameter in a plotting task equal to `psi_port` or `psi_land` invokes **psikern** and directs your plot to either a portrait-mode or a landscape mode PostScript file. For example:

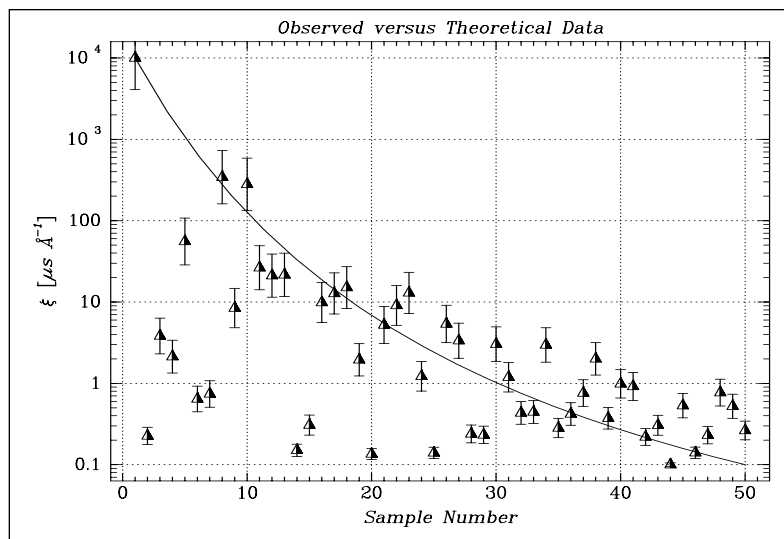
```
st> fwplot y3b10104t.c1h[19] device=psi_land
st> gflush
/tmp/pskxxx
```

The above commands would write a plot of flux vs. wavelength in landscape-mode into a temporary PostScript file, named `/tmp/pskxxx` by a UNIX system. See the online help for more about **psikern**, including plotting in color and incorporating PostScript fonts into your plots.

igi

As your plotting needs grow more sophisticated—and especially as you try preparing presentations or publication-quality plots—you should investigate the Interactive Graphics Interpreter, or **igi**. This task, in the STSDAS **stplot** package, can be used with images as well as two- and three-dimensional tables and can draw axes, error bars, labels, and a variety of other features on plots. Different line weights, font styles, and feature shapes are available, enabling you to create complex plots. Figure 3.5 shows a sample plot created in **igi**, however, because **igi** is a complete graphics environment in itself, it is well beyond the scope of this document. You can learn more about **igi** in the *IGI Reference Manual*, available through the *STSDAS Web pages*.

Figure 3.5: Sample igi Plot.



3.5 Analyzing HST Spectra

This section describes some IRAF/STSDAS tasks that can be used for analyzing and manipulating spectral data. Some of these tasks operate directly on HST data files created by the pipeline. However, a number of the most useful IRAF tasks, such as **splot**, require special preparations of data other than STIS two-dimensional spectra. Before discussing these tasks we will first show how to recast your data into forms that are more generally accessible.

3.5.1 Preparing FOS and GHRS Data

The FOS and GHRS data reduction pipelines store fluxes and wavelengths in separate files. In GEIS format, the **.c1h** file contains the flux information and the **.c0h** file contains the wavelength information. Because IRAF tasks generally require both the flux and wavelength information to reside in the same file, you will probably want to create a new file that combines these quantities.

Several options for combining flux and wavelength information are available:

- **resample**: This simple task resamples your flux data onto a linear wavelength scale, creating a new flux file containing the starting wavelength of the new grid in the **CRVAL1** keyword and the wavelength increment per pixel in the **CD1_1** keyword. Encoding the wavelength information into these standard FITS header keywords

makes this format quite portable, but the resampling process loses some of the original flux information. In addition, the error (.c2h) and data quality (.cqh) files cannot be similarly resampled, limiting the usefulness of this technique.

- **mkmultispec**: This task writes wavelength information into the header of a flux file while preserving all the original information. It is therefore a better choice than **resample** for most applications, and we describe it in more detail below.
- **imtab**: An alternative to writing wavelength information into the header is to use the **imtab** task to create a table recording the wavelength, flux, and if desired, the error data corresponding to each pixel. Many STSDAS tasks, such as those in the STSDAS **fitting** package, can access data in tabular form, so we describe this approach in more detail as well.

mkmultispec

The most convenient method of combining wavelength and flux information, and one that has no effect on the flux data at all, is to use the **mkmultispec** task. This task places wavelength information into the headers of your flux files according to the IRAF multispec-format World Coordinate System (WCS). The multispec coordinate system is intended to be used with spectra having nonlinear dispersions or with images containing multiple spectra, and the format is recognized by many tasks in IRAF V2.10 or later. For a detailed discussion of the multispec WCS, type `help specwcs` at the IRAF prompt.

The **mkmultispec** task can put wavelength information into the flux header files in two different ways. The first involves reading the wavelength data from the .c0h file, fitting the wavelength array with a polynomial function, and then storing the derived function coefficients in the flux header file (.c1h) in multispec format. Legendre, Chebyshev, or cubic spline (spline3) fitting functions of fourth order or larger produce essentially identical results, all having rms residuals less than 10^{-4} Å, much smaller than the uncertainty of the original wavelength information. Because these fits are so accurate, it is usually unnecessary to run the task in interactive mode to examine them.



*If there are discontinuities in the wavelengths, which could arise due to the splicing of different gratings, you should run **mkmultispec** in interactive mode to verify the fits.*



*Because **mkmultispec** can fit only simple types of polynomial functions to wavelength data, this method will not work well with FOS prism data, because of the different functional form of the prism-mode dispersion solution. For prism spectra, use the header table mode of **mkmultispec** (see below) or create an STSDAS table using **imtab**.*

The other method by which **mkmultispec** can incorporate wavelength information into a flux file is simply to read the wavelength data from the .c0h file and place the entire data array directly into the header of the flux (.c1h) file. This method simply dumps the wavelength value associated with each pixel in the spectrum into the flux header and is selected by setting the parameter `function=table`. To minimize header size, set the parameter `format` to a suitable value. For example, using `format=%8.7g` will retain the original seven digits of precision of the wavelength values, while not consuming too much space in the flux header file.



Be aware that there is a physical limit to the number of header lines that can be used to store the wavelength array (approximately 1000 lines). This limit cannot be overridden. Under ordinary circumstances this limitation is not an issue. However, if many spectral orders have been spliced together, it may not be possible to store the actual wavelength array in the header, and a fit must be done instead

imtab

Another way to combine wavelengths with fluxes is to create an STSDAS table from your spectrum. The **imtab** task in the STSDAS **ttools** package reads a GEIS format spectral image and writes the list of data values to a column of an STSDAS table, creating a new output table if necessary. The following example shows how to create a flux, wavelength, and error table from group eight of a GEIS-format FOS dataset:

```
c1> imtab y0cy0108t.c0h[8] y0cy0108t.tab wavelength
c1> imtab y0cy0108t.c1h[8] y0cy0108t.tab flux
c1> imtab y0cy0108t.c2h[8] y0cy0108t.tab error
```

The last word on each command line labels the three columns “wavelength”, “flux”, and “error”.

Constructing tables is necessary if you plan to use certain tasks—such as those in the STSDAS **fitting** package—that do not currently recognize the multispec format WCS header information. Tabulating your spectra is also the best option if you want to join two or more spectra taken with different gratings into a single spectrum covering the complete wavelength range. Because the data are stored as individual wavelength-flux pairs, you do not need to resample, and therefore degrade the individual spectra to a common, linear dispersion scale before joining them. Instead, you could create separate tables for spectra from different gratings, and then combine the two tables using, for example, the **tmerge** task:

```
c1> tmerge n5548_h13.tab,n5548_h19.tab n5548.tab append
```

Note that you will first have to edit out any regions of overlapping wavelength from one or the other of the input tables so that the output table will be monotonically increasing (or decreasing) in wavelength.

3.5.2 Preparing STIS Spectra for Analysis

Calibrated STIS spectra emerge from the pipeline either as two-dimensional images (**_x2d** files) or as one-dimensional spectra in tabular form (**_x1d** files.) You can analyze calibrated two-dimensional STIS spectra in IRAF as you would with any other long-slit spectral image, because their headers already contain the necessary wavelength information. Tabulated STIS spectra can be analyzed directly using STSDAS tasks that understand the *selectors* syntax described in section 2.2.2. However, to use IRAF tasks, such as **splot**, that rely on the multispec WCS or to use STSDAS tasks that do not understand three-dimensional tables, you will have to prepare your data appropriately. This section describes two useful tasks for putting your data in the proper form:

- **tomultispec**: This task is the STIS analog to **mkmultispec**, described above. It extracts STIS spectra from tables and writes them as IRAF spectral images with wavelength information in the header.
- **txtable**: This task extracts specified data arrays from STIS table cells and places them in conventional two-dimensional tables for easier access.
- **tximage**: Extracts specified data arrays from STIS table cells and places them into 1-D images. This task can write single group GEIS files.

tomultispec

The **tomultispec** task in the **stsdas.hst_calib.ctools** package extracts one or more spectral orders from a STIS table, fits a polynomial dispersion

solution to each wavelength array, and stores the spectra in an output file in original IRAF format (OIF), using the multispec WCS. This task is layered upon the **mkmultispec** task, which performs a similar operation for FOS and GHRS calibrated spectra (see “mkmultispec” on page 3-25). Most of the parameters for **tomultispec** echo those for **mkmultispec**. As a helpful navigational aid, the STIS spectral order numbers are written to the corresponding *beam* numbers in the multispec image; the aperture numbers are indexed sequentially starting from one. You can choose to fit the dispersion solution interactively, but the default fourth-order Chebyshev polynomial will likely suffice for all STIS spectral orders, except for prism-dispersed spectra. However, you cannot use the interactive option if you are selecting more than one order from the input file.

For example, if you want to write all spectral orders from the STIS file `myfile_xld.fits` to a multispec file:

```
cl> tomultispec myfile_xld.fits new_ms.imh
```

Note that the `.imh` suffix on the output file specifies that the output file is to be an OIF file. This format is similar to GEIS format, in that it consists of two files: a header file (`.imh`) and a binary data file (`.pix`). The output format for **tomultispec** will always be OIF.

If you want to select particular spectral orders, rather than writing all the orders to the multispec file, you will need to use the **selectors** syntax. To select only the spectrum stored in row nine of the input table, the previous example would change to:

```
cl> tomultispec "myfile_xld.fits[r:row=9]" new_ms.imh
```

Note that the double quote marks around the file name and row selector are necessary to avoid syntax errors. To select a range of rows, say rows nine through eleven:

```
cl> tomultispec "myfile_xld.fits[r:row=(9:11)]" new_ms.imh
```

You can also select rows based upon values in some other column. For example, to select all rows whose spectral order lies in the range 270 to 272, type:

```
cl> tomultispec "myfile_xld.fits[r:sporder=(270:272)]" \
>>> new_ms.imh
```


The calibrated flux is extracted by default. However, other intensity data can be specified by setting the `flux_col` parameter.



Be careful not to restrict the search for matching rows too heavily



Column selectors cannot be used with `tomultispec`



Choose the type of fitting function for the `tomultispec` dispersion solution with care. Using the `table` option, which writes the entire wavelength array to the image header for each order, will fail if more than about three orders are selected. This restriction results from a limit to the number of keywords that can be used to store the dispersion relation.

txtable

Tabulated STIS spectra are stored as data arrays within individual cells of FITS binary tables (see section 2.2.2). These tables are effectively three-dimensional, with each column holding a particular type of quantity (e.g., wavelengths, fluxes), each row holding a different spectral order, and each cell holding a one-dimensional array of values spanning the wavelength space of the order. The **txtable** in the **tables.tools** package extracts these data arrays from the cells specified with the selectors syntax and stores them in the columns of conventional two-dimensional binary tables.

For example, suppose the first extension of the FITS file `data.fits` contains a STIS echelle spectrum and you want to extract only the wavelength and flux arrays corresponding to spectral order 68. You could then type:

```
tt> txtable "data.fits[1][c:WAVELENGTH,FLUX][r:sporder=68]" \
>>> out_table
```

This command would write the wavelength and flux arrays to the columns of the output table `out_table`. To specify multiple rows in a tabulated echelle spectrum, you would type:

```
tt> txtable "data.fits[1][c:WAVELENGTH,FLUX][r:row=(10:12)]" \
>>> ech1
```

This command would generate three separate output files named `ech1_r0010.tab`, `ech1_r0011.tab`, and `ech1_r0012.tab`.

See the online help for more details on **txtable** and the selectors syntax, and remember to include the double quotation marks.

The similar **tximage** task can be used to generate single-group GEIS files from STIS data, which can then be used as input to tasks such as **resample**.

```
tt> tximage "data.fits[1][c:WAVELENGTH][r:row=4]" wave.hhh
tt> tximage "data.fits[1][c:FLUX][r:row=4]" flux.hhh
```

3.5.3 General Tasks for Spectra

IRAF has many tasks for analyzing both one- and two-dimensional spectral data. Many observers will already be familiar with **noao.onedspec** and **noao.twodspec** packages, and those who are not should consult the online help. Table 3.5 lists some of the more commonly used IRAF/STSDAS spectral analysis tasks, and below we briefly describe **splot**, one of the most versatile and useful. Remember that many of these tasks expect to find WCS wavelength information in the header, so you should first run **mkmultispec** or **tomultispec** on your data, if necessary.

Table 3.5: Tasks for Working with Spectra

Task	Package	Input Format	Purpose
boxcar	images.imfilter	Image	Boxcar smooth a list of images
bplot	noao.onedspec	Multispec image	Plot spectra non-interactively
continuum	noao.onedspec	Image	Continuum normalize spectra
fitprofs	noao.onedspec	Image	Non-interactive Gaussian profile fitting to features in spectra and image lines
gcopy	stsdas.toolbox.imgtools	GEIS image	Copy multigroup images
grlist	stsdas.graphics.stplot	GEIS image	List file names for all groups in a GEIS image; used to make lists for tasks that do not use group syntax
grplot	stsdas.graphics.stplot	GEIS image	Plot arbitrary lines from 1-D image; overplots multiple GEIS groups; no error or wavelength information is used
grspec	stsdas.graphics.stplot	Multispec GEIS image	Plot arbitrary lines from 1-D image; stack GEIS groups
magnify	images.imgeom	Image	Interpolate spectrum on finer (or coarser) pixel scale
nfit1d	stsdas.analysis.fitting	Image, table	Interactive 1-D non-linear curve fitting (see section 3.5.4)
ngaussfit	stsdas.analysis.fitting	Image, table	Interactive 1-D multiple Gaussian fitting (see section 3.5.4)
poffsets	stsdas.hst_calib.ctools	GEIS image	Determine pixel offsets between shifted spectra
rapidlook	stsdas.hst_calib.ctools	GEIS image	Create and display a 2-D image of stacked 1-D images
rcombine	stsdas.hst_calib.ctools	GEIS image	Combine (sum or average) GEIS groups in a 1-D image with option of propagating errors and data quality values
resample	stsdas.hst_calib.ctools	GEIS image	Resample FOS and GHRS data to a linear wavelength scale (see section 3.5.1)
sarith	noao.onedspec	Multispec image	Spectrum arithmetic
scombine	noao.onedspec	Multispec image	Combine spectra
sfit	noao.onedspec	Multispec image	Fit spectra with polynomial function
sgraph	stsdas.graphics.stplot	Image, table	Plot spectra and image lines; allows overplotting of error bars and access to wavelength array (see section 3.4.1)
specalign	stsdas.hst_calib.ctools	GEIS image	Align and combine shifted spectra (see poffsets)
specplot	noao.onedspec	Multispec image	Stack and plot multiple spectra
splot	noao.onedspec	Multispec image	Plot and analyze spectra & image lines (see “splot” on page 3-31)

splot

The **splot** task in the IRAF **noao.onedspec** package is a good general analysis tool that can be used to examine, smooth, fit, and perform simple arithmetic operations on spectra. Because it looks in the header for WCS

wavelength information, your file must be suitably prepared. Like all IRAF tasks, **splot** can work on only one group at a time from a multigroup GEIS file. You can specify which GEIS group you want to operate on by using the square bracket notation, for example:

```
cl> splot y0cy0108t.c1h[8]
```

If you don't specify a group in brackets, **splot** will assume you want the first group. In order to use **splot** to analyze your FOS or GHRS spectrum, you will first need to write the wavelength information from your .c0h file to the header of your .c1h files in WCS, using the **mkmultispec** task (see "mkmultispec" on page 3-25).

The **splot** task has *many* available options described in detail in the online help. Table 3.6 summarizes a few of the more useful cursor commands for quick reference. When you are using **splot**, a log file saves results produced by the equivalent width or de-blending functions. To specify a file name for this log file, you can set the **save_file** parameter by typing, for example:

```
cl> splot y0cy0108t.c1h[8] save_file=results.log
```

If you have used **tomultispec** to transform a STIS echelle spectrum into .imh/.pix OIF files with WCS wavelength information (see "tomultispec" on page 3-27), you can step through the spectral orders stored in image lines using the ")", "(", and "#" keys. To start with the first entry in your OIF file, type:

```
cl> splot new_ms.imh 1
```

You can then switch to any order for analysis using the ")" key to increment the line number, the "(" key to decrement, and the "#" key to switch to a specified image line. Note the beam label that gives the spectral order cannot be used for navigation. See the online help for details.

Table 3.6: Useful splot Cursor Commands

Command	Purpose
<i>Manipulating spectra</i>	
f	Arithmetic mode; add and subtract spectra
l	Convert spectrum from f_v to f_λ (invert transformation with “n”)
n	Convert spectrum from f_λ to f_v
s	Smooth with a boxcar
u	Define linear wavelength scale using two cursor markings
<i>Fitting spectra</i>	
d	Mark two continuum points & de-blend multiple Gaussian line profiles
e	Measure equivalent width by marking points around target line
h	Measure equivalent width assuming Gaussian profile
k	Mark two continuum points and fit a single Gaussian line profile
m	Compute the mean, RMS, and S/N over marked region
t	Enter interactive curve fit function (usually used for continuum fitting)
<i>Displaying and redrawing spectra</i>	
a	Expand and autoscale data range between cursor positions
b	Set plot base level to zero
c	Clear all windowing and redraw full current spectrum
r	Redraw spectrum with current windowing
w	Window the graph
x	Etch-a-sketch mode; connects two cursor positions
y	Overplot standard star values from calibration file
z	Zoom graph by a factor of two in X direction
\$	Switch between physical pixel coordinates and world coordinates
<i>General file manipulation commands</i>	
?	Display help
g	Get another spectrum
i	Write current spectrum to new or existing image
q	Quit and go on to next input spectrum

3.5.4 STSDAS fitting Package

The STSDAS **fitting** package contains several tasks, as listed in table 3.7, for fitting and analyzing spectra and images. The **ngaussfit** and **nfit1d** tasks, in particular, are very good for interactively fitting multiple Gaussians and nonlinear functions, respectively, to spectral data. These tasks do not currently recognize the multispec WCS method of storing wavelength information. They recognize the simple sets of dispersion keywords such as W0, WPC and CRPIX, CRVAL, and CDELTA, but these forms apply only to linear coordinate systems and therefore would require resampling of your data onto a linear wavelength scale first. However, these tasks do accept input from STSDAS tables, in which you can store the wavelength and flux data value pairs or wavelength, flux, error value triples (see “imtab” on page 3-26).

Table 3.7: Tasks in the STSDAS fitting Package

Task	Purpose
function	Generate functions as images, tables, or lists
gfit1d	Interactive 1-d linear curve fit to images, tables, or lists
i2gaussfit	Iterative 2-d Gaussian fit to noisy images (script)
nfit1d	Interactive 1-d non-linear curve fit to images, tables, or lists
ngaussfit	Interactive 1-d multiple Gaussian fit to images, tables, or lists
n2gaussfit	2-d Gaussian fit to images
prfit	Print contents of fit tables created by fitting task

When using tasks such as **ngaussfit** and **nfit1d**, you must provide initial guesses for the function coefficients as input to the fitting algorithms. You can either specify these initial guesses via parameter settings in the task’s parameter sets (psets) or enter them interactively. For example, suppose you want to fit several features using the **ngaussfit** task. Using the default parameter settings, you can start the task by typing:

```
fi> ngaussfit n4449.hhh linefits.tab
```

This command reads spectral data from the image `n4449.hhh` and stores the results of the line fits in the STSDAS table `linefits.tab`. After you start the task, your spectrum should appear in a plot window and the task will be left in cursor input mode. You can use the standard IRAF cursor mode commands to rewindow the plot, restricting your display to the region around a particular feature or features that you want to fit. You may then want to:

- Define a sample region (using the cursor mode **S** command) over which the fit will be computed so that the task will not try to fit the entire spectrum.
- Define an initial guess for the baseline coefficients by placing the cursor at two baseline locations (one on either side of the feature to be fitted) using the **B** keystroke.
- Use the **R** keystroke to redraw the screen and see the baseline that you've just defined.
- Set the initial guesses for the Gaussian centers and heights by placing the cursor at the peak of each feature and typing **P**.
- Press **F** to compute the fit once you've marked all the features you want to fit.

The results will automatically be displayed. You can use the `:show` command to see the coefficient values.

Note that when the **ngaussfit** task is used in this way (i.e., starting with all default values), the initial guess for the FWHM of the features will be set to a value of one. Furthermore, this coefficient and the coefficients defining the baseline are held fixed by default during the computation of the fit, unless you explicitly tell the task through cursor *colon* commands³ to allow these coefficients to vary. It is sometimes best to leave these coefficients fixed during an initial fit, and then to allow them to vary during a second iteration. This rule of thumb also applies to the setting of the **errors** parameter which controls whether or not the task will estimate error values for the derived coefficients. Because the process of error estimation is very CPU-intensive, it is most efficient to leave the error estimation turned off until you've got a good fit, and then turn the error estimation on for one last iteration.

Figure 3.6 and figure 3.7 shows the results of fitting the H β (4861Å) and [OIII] (4959 and 5007 Å) emission features in the spectrum of NGC 4449. The resulting coefficients and error estimates (in parentheses) are shown in figure 3.7.

3. To see the online help for details and a complete listing of cursor mode colon commands: type `help cursor`.

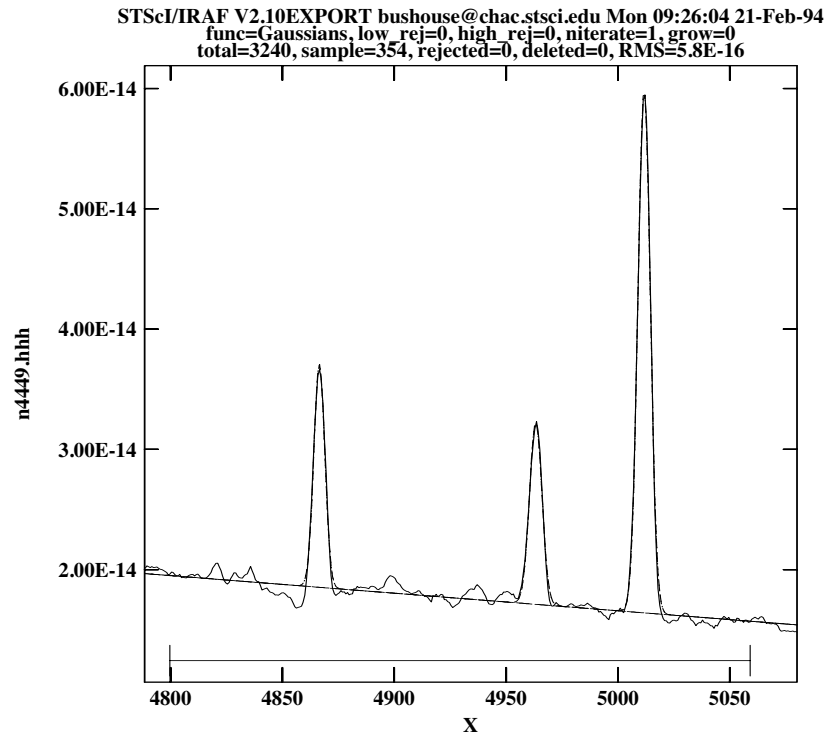
Figure 3.6: Fitting H β and [OIII] Emission Features in NGC 4449

Figure 3.7: Coefficients and Error Estimates

```

function = Gaussians
coeff1 = 8.838438E-14   (0.)           - Baseline zeropoint (fix)
coeff2 = -1.435682E-17 (0.)           - Baseline slope (fix)
coeff3 = 1.854658E-14   (2.513048E-16) - Feature 1: amplitude (var)
coeff4 = 4866.511       (0.03789007)   - Feature 1: center (var)
coeff5 = 5.725897       (0.0905327)    - Feature 1: FWHM (var)
coeff6 = 1.516265E-14   (2.740680E-16) - Feature 2: amplitude (var)
coeff7 = 4963.262       (0.06048062)    - Feature 2: center (var)
coeff8 = 6.448922       (0.116878)     - Feature 2: FWHM (var)
coeff9 = 4.350271E-14   (2.903318E-16) - Feature 3: amplitude (var)
coeff10 = 5011.731      (0.01856957)    - Feature 3: center (var)
coeff11 = 6.415922      (0.03769293)    - Feature 3: FWHM (var)
rms      = 5.837914E-16
grow     = 0.
naverage = 1
low_reject = 0.
high_reject = 0.
niterate = 1
sample   = 4800.132:5061.308

```


3.5.5 specfit

The **specfit** task, in the STSDAS **contrib** package, is another powerful interactive facility for fitting a wide variety of emission-line, absorption-line, and continuum models to a spectrum. This task was written by Gerard Kriss. Extensive online help is available to guide you through the task,⁴ although because it is a contributed task, little to no support is provided by the STSDAS group.

The input spectrum to **specfit** can be either an IRAF image file or an ASCII file with a simple three-column (wavelength, flux, and error) format. If the input file is an IRAF image, the wavelength scale is set using values of W0 and WPC or CRVAL1 and CDELTA1. Hence, for image input, the spectral data must be on a linear wavelength scale. In order to retain data on a non-linear wavelength scale, it is necessary to provide the input spectrum in an ASCII file, so that you can explicitly specify the wavelength values associated with each data value. The online help explains a few pieces of additional information that must be included as header lines in an input text file.

By selecting a combination of functional forms for various components, you can fit complex spectra with multiple continuum components, blended emission and absorption lines, absorption edges, and extinction. Available functional forms include linear, power-law, broken power-law, blackbody, and optically thin recombination continua, various forms of Gaussian emission and absorption lines, absorption-edge models, Lorentzian line profiles, damped absorption-line profiles, and mean galactic extinction.

3.6 References

3.6.1 Available from STScI

(<http://stdas.stsci.edu/STSDAS.html>)

- *STSDAS Users Guide*, version 1.3, September 1994.
- *STSDAS Installation and Site Managers Guide*, version 2.3, June 2001.
- *Synphot Users Guide*, December 1998.
- *IGI Reference Manual*, version 1.3, October 1992.

4. Additional information is available in the *Astronomical Data Analysis Software and Systems III*, ASP Conference Series, Vol. 61, page 437, 1994.

3.6.2 Available from NOAO

(<http://iraf.noao.edu/docs/docmain.html>)

- *A Beginners Guide to Using IRAF*, 1993, J. Barnes.
- *Photometry Using IRAF*, 1994, L. Wells.
- *A User's Guide to Stellar CCD Photometry with IRAF*, 1992, P. Massey and L. Davis.

3.6.3 Other References Cited in This Chapter

- Horne, K., 1988, in *New Directions in Spectrophotometry*, A.G.D. Philip, D.S. Hayes, and S.J. Adelman, eds., L. Davis Press, Schenectady NY, p. 145.
- Koorneef, J., R. Bohlin, R. Buser, K. Horne, and D. Turnshek, 1986, in *Highlights of Astronomy*, Vol. 7, J.-P. Swinds, ed., Reidel, Dordrecht, p. 833.
- Kriss, G., 1994, in *Astronomical Data Analysis Software and Systems III*, PASP Conference Series, Vol. 61, p. 437.



PART II:

STIS Data Handbook

This handbook describes data from the Space Telescope Imaging Spectrograph (STIS) on board the Hubble Space Telescope (HST), and how to manipulate, calibrate, and analyze those data.

STIS Introduction

How to Use this Handbook

This handbook describes data from the Space Telescope Imaging Spectrograph (STIS) on board the Hubble Space Telescope (HST), and how to manipulate, calibrate, and analyze those data. The current version of the *STIS Data Handbook* is presented as an independent and self-contained document, extensively built on the contents of version 3 of the HST data handbook. Users are referred to a companion volume, *Introduction to HST Data Handbook*, for more general information about the details of acquiring data from the HST archive, HST file formats, and general purpose software for displaying and processing HST data. For detailed information on the capabilities of the STIS, and how to plan observations, users should refer to the *STIS Instrument Handbook*.

The current edition of the *STIS Data Handbook* was produced in late 2001; in early 2002, servicing mission 3b will install the Advanced Camera for Surveys (ACS) and a new cooling system for the Near-Infrared Camera and MultiObject Spectrograph (NICMOS), but no direct modifications to the STIS. This is the first major revision to the handbook since the STIS was launched in 1997, and thus includes many updates to the calibration pipeline, such as the echelle scattered light subtraction and a correction to TIMETAG data processing. Besides improvements to the pipeline, there have been a few other events that affect STIS data. On-the-Fly-Reprocessing was implemented in April 2001, which now ensures that data retrieval from the archive will always use the latest pipeline software and reference files. The STIS primary (Side-1) electronics failed on 2001-May-16, and operations had to be resumed on the backup (Side-2) electronics; CCD performance is somewhat degraded on Side 2, with a variable dark signal (dependent on the detector temperature) and increased read noise. Overall, the STIS continues to perform well, and our understanding and characterization of the instrument have progressed tremendously since launch. While this document provides extensive information for reducing and analyzing STIS data, for the latest information and updates, users should consult the STIS web page

(<http://www.stsci.edu/instruments/stis>). The *STScI Analysis Newsletters* (STANs) and *Instrument Science Reports* (ISRs), available under the Documents link, are particularly helpful in this regard.

Bahram Mobasher, Chief Editor, HST Data Handbook

Thomas M. Brown, Editor, STIS Data Handbook

Handbook Structure

The *STIS Data Handbook* is organized in five chapters, which discuss the following topics:

- *Chapter 1: STIS Overview*, provides a brief overview of the instrument and its operational capabilities. If you are not already familiar with the details of STIS, you should begin here.
- *Chapter 2: STIS Data Structure*, describes the contents of STIS data files, the meanings of selected header keywords, and the relationship of the data products to the original Phase II proposal. If you are not familiar with the filenames, header keywords, or contents of the data files from STIS, you should read this chapter next.
- *Chapter 3: STIS Calibration*, describes how the calibration pipeline processes your observation, how to determine if your data files need recalibration, and how to recalibrate them. If you are not familiar with the important characteristics of STIS data and the standard procedures for reducing them, or do not know how your data have been calibrated, you should read this chapter.
- *Chapter 4: STIS Error Sources*, describes the sources of uncertainty and limiting accuracies of STIS data. STIS observers should read this chapter to acquaint themselves with the limitations of the data that remain after pipeline calibration.
- *Chapter 5: STIS Data Analysis*, describes certain methods and IRAF/STSDAS tasks for analyzing the data. In particular, it discusses STIS software tools, calibrated imaging and spectral formats, TIME-TAG data, and acquisition data. Most observers will find this chapter useful when determining how they should reduce and analyze their data.

There are some important pieces of general information about HST data, the HST Archive, and the IRAF and STSDAS analysis software that are not specific to the STIS, and which are therefore not discussed here in the *STIS Data Handbook*. We refer the reader to the most recent version of the companion *Introduction to HST Data Handbook* for this information. In particular, chapters 1, 2 and 3 of the *Introduction to HST Data Handbook* describe how to retrieve and read HST data, HST file formats, and the

basics of the STSDAS software package. Appendix A offers an IRAF primer. Appendix B describes HST file name conventions and image “associations.” Appendix C describes HST Observation Logs, and Appendix E discusses HST resources on the Internet. Additional help with HST data is always available via email to the STScI Help Desk at help@stsci.edu.

Typographic Conventions

To help you understand the material in the *STIS Data Handbook*, we will use a few consistent typographic conventions.






Visual Cues

The following typographic cues are used:

- **bold** words identify a STSDAS or IRAF task or package name.
- `typewriter-like` words identify a file name, system command, or response that is typed or displayed as shown.
- *italic* type indicates a new term, an important point, or a mathematical variable.
- ALL CAPS identifies a header keyword or a table column.

Keystrokes

Keystroke commands and sequences are identified by the following formats.

-   - When two keys are linked by a *dash*, both keys should be pressed at the same time.
-   - When a *space* separates two keys, a sequence is indicated. Press one key, release it, then press the other.
-  - Press *only* the key. If we meant that you should press shift with the key, we would say so, such as in the first example, above.

Comments

Occasional side comments point out three types of information, each identified by an icon in the left margin.



Warning: You could corrupt data, produce incorrect results, or create some other kind of severe problem



Heads Up: Here is something that is often done incorrectly or that is not obvious.



Tip: No problems...just another way to do something or a suggestion that might make your life easier.

STIS Overview

In this chapter. . .

1.1 Instrument Capabilities and Design / 1-1
1.2 Basic Instrument Operations / 1-7
1.3 STIS Operations on Side 2 / 1-9

This chapter provides an overview of the capabilities and design of the STIS and describes the basic instrument operations. The material presented here is excerpted from the more complete information provided in the *STIS Instrument Handbook*, and we refer you there for detailed information about the properties of the STIS as an instrument.

1.1 Instrument Capabilities and Design

The [Space Telescope Imaging Spectrograph](#) (STIS) was built by Ball Aerospace Corporation for the Goddard Space Flight Center (GSFC) Laboratory for Astronomy and Solar Physics, under the direction of Bruce Woodgate, the Principal Investigator (PI). The STIS has been performing very well since its installation during the second HST servicing mission in February 1997. A basic description of the instrument and of its on-orbit performance through the Servicing Mission Orbital Verification (SMOV) program is provided by Kimble et al. (1998, *ApJL*, 492, L83). We encourage all STIS users to reference this paper, and to review the related papers in this special *ApJ Letters* that describe the Early Release Observations and demonstrate the realized scientific capabilities of the STIS.

The STIS is a versatile instrument providing both imaging and spectroscopic capabilities with three two-dimensional detectors operating

from the ultraviolet to the near-infrared. The optics and detectors have been designed to exploit the high spatial resolution of the HST. The STIS has first order gratings, designed for spatially resolved long-slit spectroscopy over its entire spectral range, and echelle gratings, available only in the ultraviolet, that maximize the wavelength range covered in a single spectral observation of a point source. The STIS Flight Software supports on-board target acquisitions and peakups to place science targets on slits and coronagraphic bars.

The STIS can be used to obtain:

- Spatially resolved, long-slit or slitless spectroscopy from 1150–10,300 Å at low to medium spectral resolution ($R \sim 500$ –17,000) in first order.
- Echelle spectroscopy at medium to high spectral resolution ($R \sim 30,000$ –110,000), covering a broad instantaneous spectral range ($\Delta\lambda \sim 800$ or 200 Å, respectively) in the ultraviolet (1150–3100 Å).

In addition to these two prime capabilities, the STIS also provides:

- A modest imaging capability using: the solar-blind CsI far-ultraviolet Multi-Anode Microchannel Array (MAMA) detector (1150–1700 Å); the solar-insensitive Cs₂Te near-ultraviolet MAMA detector (1650–3100 Å); and the optical CCD (2000–11,000 Å) through a small complement of narrow- and broad-band filters.
- Objective prism spectroscopy ($R \sim 500$ –10) in the vacuum ultraviolet (1200–3100 Å).
- High time resolution ($\Delta\tau = 125$ microseconds) imaging and spectroscopy in the ultraviolet (1150–3100 Å) and moderate time resolution ($\Delta\tau \sim 20$ seconds) CCD imaging and spectroscopy in the optical and near IR (2000–10,300 Å).
- Coronagraphic imaging in the optical and near IR (2000–10,300 Å), and bar-occulted spectroscopy over the entire spectral range (1150–10,300 Å).

A complete list of gratings and filters are given in table 1.1 and table 1.2, respectively, with references to the *STIS Instrument Handbook* for more details.

Table 1.1: STIS Spectroscopic Capabilities

Grating	Spectral Range (Å)		Spectral Resolution		No. Prime Tilts ¹	Detector	Page in STIS Instr. Handbook	Slits (apertures) ^{2,3,4,5}	
	Complete	Per Tilt	Scale $\Delta\lambda$ (Å per pixel)	Resolving Power ($\lambda/2\Delta\lambda$)					
CCD First-Order Spectroscopy									
G750L	5240–10,270	5030	4.92	530–1040	1	CCD	278	52X0.05 52X0.1 52X0.2 52X0.5 52X2 52X0.2F1	
G750M	5450–10,140	570	0.56	4870–9050	9	CCD	281		
G430L	2900–5700	2800	2.73	530–1040	1	CCD	284		
G430M	3020–5610	286	0.28	5390–10,020	10	CCD	287		
G230LB	1680–3060	1380	1.35	620–1130	1	CCD	290		
G230MB	1640–3190	155	0.15	5470–10,630	11	CCD	294		
MAMA First-Order Spectroscopy									
G230L	1570–3180	1610	1.58	500–1010	1	NUV-MAMA	298		
G230M	1640–3100	90	0.09	9110–17,220	18	NUV-MAMA	301		
G140L	1150–1730	610	0.60	960–1440	1	FUV-MAMA	304		
G140M	1140–1740	55	0.05	11,400–17,400	12	FUV-MAMA	307		
MAMA Echelle Spectroscopy									
E230M	1570–3110	800	$\lambda/60,000$	30,000	2	NUV-MAMA	310	0.2X0.2, 0.2X0.06	
E230H	1620–3150	267	$\lambda/228,000$	114,000	6	NUV-MAMA	313	0.2X0.2, 0.2X0.09	
E140M	1150–1710	620	$\lambda/91,700$	45,800	1	FUV-MAMA	316	0.2X0.2, 0.2X0.06	
E140H	1150–1700	210	$\lambda/228,000$	114,000	3	FUV-MAMA	319	0.2X0.2, 0.2X0.09	
MAMA Prism Spectroscopy									
PRISM	1150–3620	2470	0.2 - 72	2900 – 25	1	NUV-MAMA	322	52X0.05, 52x0.1, 52X0.2, 52X0.5, 52X2	

1. Number of exposures at distinct tilts needed to cover spectral range of grating with 10% wavelength overlap between adjacent settings.
2. Naming convention gives dimensions of slit in arcseconds. For example, 52X0.1 indicates the slit is 52 arcsec long perpendicular to the dispersion direction and 0.1 arcsec wide in the dispersion direction. The F (e.g., in 52X0.2F1) indicates that it is the fiducial bar to be used for coronagraphic spectroscopy. F25MGII is supported with all NUV-MAMA gratings and the prism.
3. For the MAMA first-order modes, only ~ 25 arcseconds of the long slit projects on the detector. (See also "Slits for First-Order Spectroscopy" on page 45 of the *STIS Instrument Handbook*.)
4. Full-aperture clear (50CCD or 25MAMA), longpass-filtered (F25QTZ or F25SRF2 in UV), and neutral-density-filtered slitless spectroscopy are also supported with the appropriate first-order and echelle gratings, as well as the prism.
5. The 6X0.2 and 52X0.05 long slits are also supported for use with the echelle gratings, but with order overlap. The high S/N multislits 0.2X0.2FP (A-E) and 0.2X0.06FP (A-E) (see Chapter 12), as well as the very narrow 0.1X0.03 for maximum spectral resolution, are likewise supported with the echelles. For the echelles, the filtered slits 0.2X0.05ND and 0.3X0.05ND are also supported. The 0.1X0.09 and 0.1X0.2 slits are supported for E230H only.

Table 1.2: STIS Imaging Capabilities

Aperture Name	Filter	Central Wavelength (λ_c in Å)	FWHM ($\Delta\lambda$ in Å)	Field of View (arcsec)	Detector	Page in STIS Instr. Handbook
<i>Visible - plate scale 0."05071 \pm 0."00007 per pixel</i>						
50CCD	Clear	5850	4410	52 x 52	STIS/CCD	378
F28X50LP	Optical longpass	7230	2720	28 x 52 ¹	STIS/CCD	381
F28X50OIII	[O III]	5007	5	28 x 52 ¹	STIS/CCD	384
F28X50OII	[O II]	3740	80	28 x 52 ¹	STIS/CCD	387
50CORON	Clear + coronographic fingers	5850	4410	52 x 52	STIS/CCD	390
<i>Ultraviolet - plate scale ~ 0.0246 arcseconds per pixel²</i>						
25MAMA	Clear	2220	1200	25 x 25	STIS/NUV-MAMA	392
		1370	320		STIS/FUV-MAMA	419
F25QTZ	UV near longpass	2320	1010	25 x 25	STIS/NUV-MAMA	397
		1590	220		STIS/FUV-MAMA	424
F25SRF2	UV far longpass	2270	1110	25 x 25	STIS/NUV-MAMA	404
		1480	280		STIS/FUV-MAMA	431
F25MGII	Mg II	2800	70	25 x 25	STIS/NUV-MAMA	406
F25CN270	Continuum near 2700 Å	2700	350	25 x 25	STIS/NUV-MAMA	409
F25CIII	C III]	1909	70	25 x 25	STIS/NUV-MAMA	412
F25CN182	Continuum near 1800 Å	1820	350	25 x 25	STIS/NUV-MAMA	415
F25LYA	Lyman- α	1216	85	25 x 25	STIS/FUV-MAMA	434
<i>Neutral-Density-Filtered Imaging</i>						
F25NDQ1	ND=10 ⁻¹	1150–10,300 Å		12 x 12	STIS/CCD	76
F25NDQ2	ND=10 ⁻²			12 x 12	STIS/NUV-MAMA	
F25NDQ3	ND=10 ⁻³			12 x 12	STIS/FUV-MAMA	
F25NDQ4	ND=10 ⁻⁴			12 x 12		
F25ND3	Neutral-density filter, ND=10 ⁻³	1150–10,300 Å		25 x 25	STIS/NUV-MAMA	395
					STIS/FUV-MAMA	422
F25ND5	ND=10 ⁻⁵	1150–10,300 Å		25 x 25	STIS/NUV-MAMA	397
					STIS/FUV-MAMA	424

1. The dimensions are 28 arcsec on AXIS2=Y and 52 arcsec on AXIS1=X. See Figure 3.2 and Figure 11.1 in the *STIS Instrument Handbook*.

2. The MAMA plate scales differ by about 1% in the AXIS1 and AXIS2 directions, a factor that must be taken into account when trying to add together rotated images. Also, the FUV-MAMA uses a different mirror in the filtered and unfiltered modes. In the filtered mode, the plate scale is 0.3% larger (more arcsec/pixel). Further information on geometric distortions can be found in Chapter 14 of the *STIS Instrument Handbook*.

1.1.1 The STIS Detectors

The STIS uses three large format (1024 x 1024 pixel) detectors:

- A Scientific Image Technologies (SITE) CCD, called the **STIS/CCD**, with 0.05 arcsec square pixels, covering a nominal 51 x 51 arcsec square field of view (FOV), operating from ~2000 to 10,300 Å.
- A Cs₂Te MAMA detector, called the **STIS/NUV-MAMA**, with 0.024 arcsec square pixels, and a nominal 25 x 25 arcsec square field of view (FOV), operating in the near ultraviolet from 1650 to 3100 Å.
- A solar blind CsI MAMA, called the **STIS/FUV-MAMA**, with 0.024 arcsec pixels, and a nominal 25 x 25 arcsec square FOV, operating in the far-ultraviolet from 1150–1700 Å.

The basic observational parameters of these detectors are summarized in table 1.1 and table 1.2.

The CCD: The CCD provides high quantum efficiency and good dynamic range in the near-ultraviolet through near-infrared. It produces a time-integrated image in the so-called **ACCUM** data taking mode. As with all CCDs, there is noise (*read noise*) and time (*read time*) associated with reading out the detector. Time-resolved work with this detector is done by taking a series of multiple short exposures. The minimum exposure time is 0.1 sec, and the minimum time between successive identical exposures is 45 sec for full-frame readouts and 20 sec for subarray readouts. CCD detectors are capable of high dynamic range observations; for a single exposure taken with GAIN=4, the depth is limited by the CCD full well (roughly ~144,000 e⁻), while for a single exposure taken with GAIN=1, the depth is limited by the gain amplifier saturation (~33,000 e⁻). This is the maximum amount of charge (or counts) that can accumulate in any one pixel during any one exposure, without saturation. Cosmic rays affect all CCD exposures, and observers will generally want to **CR-SPLIT** their observations by breaking the observations into a number of multiple exposures of less than 1000 sec each. This allows cosmic-ray removal in post-observation data processing.

The MAMAs: The two MAMA detectors are *photon counting* detectors that provide a two-dimensional ultraviolet imaging capability. They can be operated either in **ACCUM** mode, to produce a time-integrated image, or in **TIMETAG** mode, to produce an event stream with high (125 μsec) time resolution. Doppler correction for the spacecraft motion is applied automatically on-board for data taken in **ACCUM** high spectral resolution modes.

The STIS MAMA detectors are subject to both *performance* and *safety* brightness limits. At high local (>50 cts sec⁻¹ pixel⁻¹) and global (>285,000

cts sec⁻¹) illumination rates, counting becomes nonlinear in a way that is not correctable. At only slightly higher illumination rates, the MAMA detectors are subject to damage. Specifically, charge is extracted from the microchannel plate during UV observations, and overillumination can cause decreased quantum efficiency (due to gain decline in the overexposed region) or catastrophic failure (high voltage arcing within the sealed tube due to excess gas generation from the plate). Thus, MAMA observations are subject to bright object checks.

Current information indicates that the pixel-to-pixel flat-fields are stable at the 1% level, which is the signal-to-noise of the flats. Furthermore, these flats show no signs of DQE loss in regions where the detector has been heavily exposed.

A signal-to-noise of 50:1 per spectral resolution element is routinely obtained for extracted spectra of point sources when integrated over the observed aperture. Higher signal-to-noise values of 100-300 can be obtained by stepping the target along the slit in the first order modes, or by use of FP-SPLIT slits with the echelles.

1.1.2 STIS Physical Configuration

The STIS optical design includes corrective optics to compensate for the spherical aberration of the HST, a focal plane slit wheel assembly, collimating optics, a grating selection mechanism, fixed optics, and focal plane detectors. An independent calibration lamp assembly can illuminate the focal plane with a range of continuum and emission line lamps.

The *slit wheel* contains apertures and slits for spectroscopic use and the clear, filtered, and coronagraphic apertures for imaging. The slit wheel positioning is repeatable to very high precision: ± 7.5 and 2.5 milli-arcsec in the spatial and spectral directions, respectively.

The *grating wheel*, or Mode Selection Mechanism (MSM), contains the first-order gratings, the cross-disperser gratings used with the echelles, the prism, and the mirrors used for imaging. The MSM is a nutating wheel that can orient optical elements in three dimensions. It permits the selection of one of its 21 optical elements as well as adjustment of the tip and tilt angles of the selected grating or mirror. The grating wheel exhibits non-repeatability that is corrected in post-observation data processing using contemporaneously obtained comparison lamp exposures (i.e., wavecal).

For some gratings, only a portion of the spectral range of the grating falls on the detector in any one exposure. These gratings can be scanned (tilted by the MSM) so that different segments of the spectral format are moved onto the detector for different exposures. For these gratings a set of pre-specified central wavelengths, corresponding to specific MSM positions (i.e., grating tilts) have been defined.

The STIS has two independent calibration subsystems, the HITM (Hole in the Mirror) system and the Insert Mechanism (IM) system. The HITM system contains two Pt-Cr/Ne line lamps, used to obtain wavelength comparison exposures and to illuminate the slit during target acquisitions. Light from the HITM lamps is projected through a hole in the second correction mirror (CM2); for wavecal data taken before 1998-Nov-9, light from the external sky fell on the detector when the HITM lamps were used, but for subsequent wavecal data, an external shutter was closed to block external sky light. The IM system contains flatfielding lamps (a tungsten lamp for CCD flats, a deuterium lamp for NUV-MAMA flats, and a Krypton lamp for FUV-MAMA flats) and a single Pt-Cr/Ne line comparison lamp. When the IM lamps are used, the Calibration Insert Mechanism (CIM) is inserted into the light path, blocking all external light. Observers will be relieved to know that the ground system will *automatically* choose the right subsystem and provide the necessary calibration exposures.

1.2 Basic Instrument Operations

1.2.1 Target Acquisitions and Peakups

Once the telescope acquires its guide stars, your target will be within ~1–2 arcsec of the aperture center. For science observations taken through apertures less than three arcsec in either dimension, and for observations involving the coronagraphic bars, a target acquisition exposure should be taken to center the target in your chosen science aperture. Furthermore, if either dimension of the aperture is less than or equal to 0.1 arcsec, the acquisition exposure should be followed by one or more peakup exposures to refine the target centering of point or point-like sources. The nominal accuracy of STIS point-source ($V < 21$ mag) target acquisition is 0.01 arcsec, with a peak-up accuracy of 5% of the slit width used. Acquisition exposures always use the CCD, one of the filtered or unfiltered apertures for CCD imaging, and a mirror as the optical element in the grating wheel (as opposed to a dispersive element). Peakup exposures use a science slit or coronagraphic aperture, the CCD, and either a mirror or a spectroscopic element in the grating wheel.

1.2.2 Routine Wavecals

Each time the MSM is moved to select a new optical element or to tilt a grating, the resulting spectrum is projected onto the detector with an uncertainty (lack of repeatability) of roughly plus or minus 3 pixels. In

addition, thermal effects cause the spectrum to drift slowly with time (typical drifts are 0.1 pixels per orbit, with extreme cases of forced large temperature swings as high as 0.35 pixels per orbit). An internal calibration lamp observation (WAVECAL) is automatically taken, following each use of a new grating element or new scan position (grating tilt) and every 40 minutes thereafter, in order to allow calibration of the zero point of the wavelength (dispersion) and spatial (cross dispersion) axes in the spectroscopic science data during post-observation data processing. These routine, automatically-occurring, wavecal observations are expected to provide sufficient wavelength zeropoint accuracy for the large majority of GO science. Only if your science requires particularly accurate tracking of the wavelength zeropoints, you need to insert additional “wavecal” observations in your exposure sequence.

1.2.3 Data Storage and Transfer

At the conclusion of each exposure, the science data are read out from the detector in use and placed in the STIS internal buffer memory, where they are stored until they can be transferred to the HST data recorder (and thereafter to the ground). This design makes for more efficient use of the instrument, as up to seven CCD or four MAMA full frame images can be stored in the internal buffer at any time. The frames can be transferred out of the internal buffer to the data recorder during subsequent exposures, as long as those exposures are longer than three minutes.

The STIS internal buffer stores the data in a 16-bit per pixel format. This format imposes a maximum of 65,536 data numbers per pixel. For the MAMA detectors this number is equivalent to a limit on the total number of *photons* per pixel that can be accumulated in a single exposure. For the a single exposure with the CCD, the gain amplifier saturation level (33,000 e^-) limits the total counts per pixel that can be sustained at GAIN=1, while the CCD full well (144,000 e^- or 36,000 DN) limits the total counts per pixel that can be sustained at GAIN=4.

1.2.4 Parallel Operations

The three STIS detectors do *not* operate in parallel—only one detector can be used at any time. Exposures with different STIS detectors can, however, be freely interleaved in an observing sequence, which incurs the overheads associated with changes in grating and aperture. The three detectors, sharing the bulk of their optical paths, also share a common field of view of the sky. While the STIS CCD *can always* be used in parallel with any of the other science instruments on the HST, there are restrictions in the parallel use of the MAMA detectors.

1.3 STIS Operations on Side 2

The STIS was built with two redundant sets of electronics. On 2001-May-16, the primary (Side-1) set of STIS electronics failed, and operations had to be resumed on the backup (Side-2) set. Although most operations on Side 2 are identical to those on Side 1, there are two important differences, both in regard to the CCD. First, the effective read noise through each of the four available CCD amplifiers has increased by approximately $1 \text{ e}^- \text{ pix}^{-1}$ in the gain=1 setting (e.g., from $\sim 4.5 \text{ e}^- \text{ pix}^{-1}$ to $5.5 \text{ e}^- \text{ pix}^{-1}$ for the default D amplifier, which has the lowest read noise). This increase in noise is spatially correlated (i.e., pattern noise), although the pattern can sometimes be mitigated through filtering. See [ISR STIS 2001-005](#) for full details on the CCD pattern noise and filtering techniques. The second ramification of the switch to Side 2 is that some of the thermal control of the CCD has been lost. This causes variations in the CCD dark rate.

On Side 1, a temperature sensor mounted on the CCD carrier provided closed-loop control of the current provided to the thermoelectric cooler (TEC), thus ensuring a stable detector temperature at the commanded set point (-83°C). Side 2 does not have a functioning temperature sensor, and so the TEC is run at a constant current. Thus, under Side-2 operations, the CCD temperature varies with that of the spacecraft environment, and these temperature changes are accompanied by changes in detector dark rate. Because the current to the TEC is now fixed at a higher value than the typical value required to hold the -83°C set point on Side 1 (i.e., 3 A vs. 2.7 A), the detector is often actually cooler than it was during Side-1 operations. The result is that the median dark rate varies from 4 to 5 e^- per 1000 s on Side 2, as opposed to 4.6 to 5 e^- per 1000 s on Side 1. Details of the temperature dependence of the STIS CCD dark rate are somewhat complicated, however, and they are fully explained in [ISR STIS 2001-003](#).

Although no sensor is available to measure the temperature of the CCD itself, there is a sensor for the CCD housing temperature. The hot side of the TEC is bonded to the CCD housing baseplate; hence with fixed TEC current, the CCD housing should track closely the detector temperature under Side-2 operations, and this can be seen by the excellent correlation between the dark rate and the housing temperature ([ISR STIS 2001-003](#)). Note that the CCD housing is far hotter than the detector itself: the housing temperature is approximately 18°C during normal operations, while the detector runs at approximately -83°C . In order to enable a temperature-dependent dark subtraction for STIS CCD data, the CCD housing temperature for each observation is planned to be included in the science header by Feb. 2002, under keyword CCDHTAV. A temperature-dependent dark correction will be possible within the STIS calibration pipeline shortly thereafter. As always, any news on calibration enhancements will be distributed within the STScI Analysis Newsletters (STANs).

STIS Data Structure

In this chapter. . .

2.1 Overview / 2-1
2.2 Types of STIS Files / 2-2
2.3 STIS File Structures / 2-5
2.4 Headers, Keywords, and Relationship to Phase II / 2-9
2.5 Error and Data Quality Arrays / 2-16
2.6 STIS Coordinate Systems / 2-18
2.7 STIS Paper Products / 2-20

This chapter explains how STIS observations are stored in STIS data files. STIS data files received from the Archive will be in FITS format, and they should remain in FITS format (i.e., do not convert the format with **strfits**). Chapter 2 of the HST Introduction describes the structure of FITS files, explains how images and tables are stored in FITS extensions, and shows how to access data in these extensions. If you are not familiar with FITS extensions, please read Chapter 2:HST File Formats.

2.1 Overview

Raw STIS data are calibrated through the STScI calstis pipeline. The pipeline unpacks the databits from individual exposures, combines them into files containing raw, uncalibrated data, and performs image and spectroscopic reduction to produce output files that can be used directly for scientific analysis (see chapter 3 for a more detailed description of the STIS pipeline). Unlike previous HST pipelines, the STIS pipeline calibrates data from multiple science exposures and any contemporaneously obtained line

lamp calibration exposures through the pipeline as a single unit. These multiple *associated* STIS exposures that are processed through the pipeline as a unit are combined into a single dataset, to allow easy identification and compact storage. See appendix B for a general explanation of HST data associations.

To work effectively with your data you will need to understand:

- The nature of the individual files in your dataset. To understand the contents of each file, see section 2.2.
- The basic format in which the STIS data are stored, the information and nature of the data stored for each observation- see section 2.3.
- How to use the header keyword information to identify the principal parameters of your observation and to determine the calibration processing steps that were performed on your dataset—see section 2.4.
- The meanings of the error and data quality arrays, which are propagated through the pipeline for each STIS science observation—see section 2.5.

2.2 Types of STIS Files

The naming convention for STIS files is *rootname_XXX.fits*, where *XXX* is a three-character file suffix. The suffix identifies the type of data within the file. Table 2.1 lists the file suffixes for both uncalibrated and calibrated data files. Depending on the type of observation you have obtained, and therefore on the path it has taken through the calibration pipeline, you will find an appropriate subset of these files in your particular data set. Table 2.2 gives examples of typical STIS datasets for different types of observations.

Table 2.1: Data File Naming Conventions

Suffix	Type	Contents
Uncalibrated		
_raw	image	Raw science ¹
_tag	table	Timetag event list
_spt	image	Support, planning & telemetry information
_wav	image	Associate wavecal exposure
_wsp	image	The _spt file for _wav (wavecal)
_asn	table	Association file
_trl	table	Trailer file (input); historical record of generic conversion
_jit	table	Contains spacecraft pointing data averaged over 3 s intervals
_jif	image	Two-Dimensional histogram of the _jit file
_pdq	table	Post observation summary and Data Quality
Calibrated		
_flt	image	Flatfielded science
_crj	image	Cosmic ray-rejected, flatfielded science
_sfl	image	Summed flatfielded science
_x1d	table	<i>1-D extracted spectra:</i> <i>* aperture extracted, background subtracted, flux and wavelength calibrated spectra</i>
_x2d	image	<i>2-D extracted data:</i> <i>* rectified, wavelength and flux calibrated spectra or</i> <i>* geometrically corrected imaging data.</i>
_sx1	table	Summed 1-D extracted spectra
_sx2	image	Summed 2-D extracted spectra
_trl	table	Trailer file (output); historical record of pipeline processing

1. Raw data from isolated wavecal, biases, darks, and flats, as well as from ACQs and ACQ/PEAKs, also have the _raw suffix.

Table 2.2: Typical STIS Output Products by Observation Type

Observation Type	Uncalibrated Files	Calibrated Files
ACQ, ACQ/PEAK	_raw	none
IMAGING, ACCUM MODE, ASSOCIATED SET (crsplit or repeatobs)	_raw, _spt, _asn, _trl	_flt, _sfl (MAMA only), _crj (CCD only)
IMAGING, ACCUM MODE, Single Exposure	_raw, _spt, _asn, _trl	_flt
FIRST ORDER SPECTROSCOPY, ACCUM MODE ASSOCIATED SET (crsplit or repeatobs)	_raw, _wav, _asn, _spt, _wsp, _trl	_flt, _sx2, _sx1 _crj (CCD only)
FIRST ORDER SPECTROSCOPY, ACCUM MODE Single Exposure	_raw, _wav, _asn, _spt, _wsp, _trl	_flt, _x2d, _x1d
ECHELLE SPECTROSCOPY, ACCUM MODE single exposure or ASSOCIATED SET	_raw, _wav, _asn, _spt, _wsp, _trl	_flt, _x2d, _x1d
TIMETAG IMAGING and SPECTROSCOPIC	_tag + ACCUM extensions	ACCUM extensions

2.2.1 Understanding Associations

An *association* is created when repeated exposures are obtained through CR-SPLITS or REPEATOBS, and when wavecal exposures are linked to science exposures. The repeated exposures in an association will appear in a single FITS file. You can recognize a file as part of an association because there will be a zero in the last position of the rootname (e.g., o3tt01010_raw.fits). The rootnames of the individual exposures in an association are contained in the association file, which has suffix _asn (e.g., o3tt01010_asn.fits). An association file holds a single binary table extension, which can be displayed with the IRAF tasks **tprint** or **tread**. The information within an association table shows how the associated exposures are related. Figure 2.1 illustrates the contents of the association table for a CRSPLIT=2 observation, with an associated wavecal.

Figure 2.1: Contents of Association Table

```

To display the association table for o3tt01010_asn.fits:
>tread o3tt01010_asn.fits
# row MEMNAME          MEMTYPE          MEMPRSNT
#
1   O3TT01AVR          CRSPLIT          yes
2   O3TT01AWR          CRSPLIT          yes
3   O3TT01AXR          WAVECAL          yes
4   O3TT01010          PRODUCT          yes

```

The association table above tells the user that the product, or data set, will have the rootname `o3tt01010`, that there will be two science exposures contained in the `o3tt01010_raw.fits` file that are CRSPLITS, and that a `o3tt01010_wav.fits` file should exist containing the contemporaneously obtained automatic wavecal. The `o3tt01010_raw.fits` file will contain six image extensions, one triplet of {SCI, ERR, DQ} for each exposure (see section 2.3.1). The pipeline will calibrate these data as a unit, producing a single cosmic-ray rejected image (*rootname_crj.fits*), its data quality and error images, and rectified spectra. Similarly, for REPEATOBS observations, in which many identical exposures are taken to obtain a time series, all the science data will be stored in sequential triplet extensions of a single FITS file. These will be processed through the **calstis** pipeline as a unit, with each image extension individually calibrated. The set of images will also be combined to produce a total time-integrated calibrated image. Chapter 3 gives more information about the pipeline processing.

2.3 STIS File Structures

All STIS data products are FITS files. Images and two-dimensional spectroscopic data are stored in FITS image extension files, which can be directly manipulated, without conversion, in the IRAF/STSDAS environment. These FITS image extension files allow an *associated* set of STIS science exposures, processed through calibration as a single unit, to be packaged into a single file. Accessing the images in the FITS image extension files in IRAF follows a simple convention explained in detail in chapter 2 of the HST Introduction. The **catfits** task can be used to display the complete contents of the primary and extension headers of the data files.

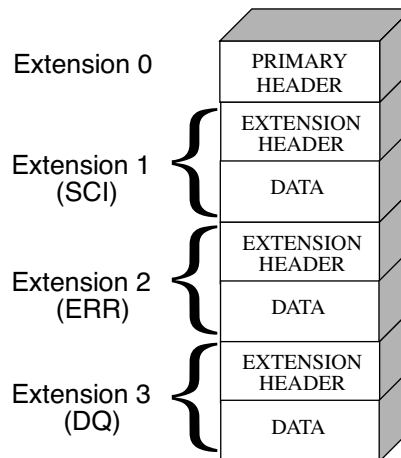
Tabular STIS information, such as extracted one-dimensional spectra or the TIMETAG mode event series, are stored as three-dimensional FITS binary tables. The tables can be accessed directly in the IRAF/STSDAS environment using tasks in the **tables.ttools** package as described in chapter 2 and chapter 3 of the HST Introduction.

2.3.1 STIS FITS Image Extension Files

Figure 2.1 illustrates the structure of a STIS FITS image extension file, which contains:

- A primary header that stores keyword information describing the global properties of all of the exposures in the file (e.g., the target name, target coordinates, total summed exposure time of all exposures in the file, optical element, aperture, detector).
- A series of image extensions, each containing header keywords with information specific to the given exposure (e.g., exposure time, world coordinate system) and a data array.

Figure 2.1: FITS Image Extension File for STIS



2.3.2 Storage of STIS Two-Dimensional Data

All uncalibrated and calibrated ACCUM mode science data (with the exception of the extracted one-dimensional spectra, see below) are stored in FITS image extension files with the particular format shown in figure 2.1. Each STIS readout generates three FITS images. SCI, ERR and DQ, as explained below:

- The first, of extension type SCI, stores the science values.
- The second, of extension type ERR, contains the statistical errors, which are propagated through the calibration process.
- The third, of extension type DQ, stores the data quality values, which flag suspect pixels in the corresponding SCI data.

The error arrays and data quality values are described in more detail in section 2.5. Each of these extensions can contain one of several different data types, including images, binary tables and ASCII text tables. The value

of the `XTENSION` keyword in the extension's header identifies the type of data the extension contains.

Two-Dimensional Rectified Spectral Images: The *rootname_sx2.fits* and *rootname_x2d.fits* files, which hold the flux and wavelength-calibrated two-dimensional spectra for long-slit first-order observations, are stored as FITS images, as are the raw and calibrated imaging data. The units of the data in the extracted two dimensional spectra are $\text{ergs sec}^{-1} \text{cm}^{-2} \text{\AA}^{-1} \text{arcsec}^{-2}$. The procedure to derive flux information from these data is described in section 5.4.1. Discussion of the one dimensional extracted spectra is presented in section 2.3.3.

Imaging Data: The final calibrated output product for CCD imaging data is the *rootname_crj.fits* file, and the final calibrated product for MAMA data is either *rootname_flt.fits* or *rootname_sfl.fits*. The units of the data in these files are counts per pixel. The conversion of the counts to flux (or magnitude) is explained in section 5.3.1.

Storage of Acquisition Images: Almost all STIS spectroscopic science exposures will have been preceded by an acquisition (and possibly an acquisition/peakup) exposure to place the target in the slit. Keywords in the header of spectroscopic data identify the dataset name of the acquisition (in the `ACQNAME` keyword).

An acquisition exposure produces a raw data file (*rootname_raw.fits*) containing three science image extensions corresponding to the three stages of the acquisition procedure:

- `[SCI,1]` is a subarray image (100 x 100 for point source acquisitions; larger for diffuse acquisitions) of the sky obtained after the initial blind pointing.
- `[SCI,2]` is an image of the same subarray after the coarse locate phase of the acquisition.
- `[SCI,3]` is an image of the 32 x 32 subarray taken during the slit-illumination phase of the target acquisition.

Storage of Acquisition/Peakup Images: An acquisition/peakup exposure will produce a single raw data file for a spiral search peakup, and one for each linear search peakup (that is, if you have performed a peakup that requires `SEARCH=LINEARAXIS1` and `SEARCH=LINEARAXIS2` scans, then two data sets will be produced—one for each scan). Keywords in the header of spectroscopic data identify the data set name of the acquisition/peakup images (in the `ACQPEAK1` and `ACQPEAK2` keywords). The *rootname_raw.fits* data file produced for an acq/peak exposure contains one science image extension:

- `[SCI,1]` is the confirmation image, taken at the end of the peakup, after the final move which places the target in the slit.

- To examine the flux values of the individual steps in the ACQ/PEAK, list the pixels (using the **listpix** task) of the fourth extension, i.e., `rootname_raw.fits[4]`.

2.3.3 Storage of STIS Tabular Data

All the time-tag and one-dimensional extracted STIS spectra are stored in binary tables, as described below.

Time-Tag: Time-Tag mode is used for high time resolution spectroscopy and imaging in the UV (with the MAMA detectors only). Time-Tag event data are contained in a binary table extension in which each row of the table corresponds to a single event in the data stream and the columns of the table contain scalar quantities that describe the event, as shown in table 2.1.

Table 2.1: Columns of a Time-Tag Data Table

Column Name	Units	Description
TIME	s	Elapsed time in seconds since the exposure start time
AXIS1	pixel	Pixel coordinate along the spectral axis, with Doppler-correction
AXIS2	pixel	Pixel coordinate along the spatial axis, (no Doppler correction)
DETAXIS1	pixel	Pixel coordinate along the spectral axis, prior to Doppler correction

The STIS pipeline collapses a time-tag event series into a single time-integrated image and processes it as if it were an ACCUM mode image. Outside of the pipeline the raw time-tag event stream can be manipulated to produce two-dimensional images which are integrated over user-specified times or manipulated directly (see section 5.6).

One-Dimensional Extracted Spectra: In the STIS pipeline, two-dimensional STIS echelle spectra are aperture extracted, order by order, and each extracted spectral order from a single spectral image is stored in a single table, one order per row. Each column of the table contains a particular quantity, such as wavelength or flux. Table 2.2 shows the contents of the different columns in a STIS extracted spectrum table. Each table cell, corresponding to a particular spectral order and type of quantity, can contain either a scalar value or an array of values. For example, each cell in the **WAVELENGTH** column contains a one-dimensional array of wavelengths corresponding to a spectral order given by the scalar in the **SPORDER** column on the same row.

There will be a separate table extension for each associated exposure in an associated set. For example, if you specified NEXP=3 in your Phase II instructions, you will find the extracted spectrum from the second exposure in the second table extension.

Table 2.2: Columns of a STIS Extracted Spectrum Table

Column Name	Contents	Units	Description
SPORDER	scalar		Spectral order number
NELEM	scalar		Number of valid elements in each array
WAVELENGTH	array	Å	Wavelengths corresponding to fluxes
GROSS	array	counts s ⁻¹	Extracted spectrum before subtracting BACKGROUND
BACKGROUND	array	counts s ⁻¹	Background that was subtracted to obtain NET
NET	array	counts s ⁻¹	Difference of GROSS and BACKGROUND arrays
FLUX	array	erg s ⁻¹ cm ⁻² Å ⁻¹	Flux calibrated NET spectrum
ERROR	array	erg s ⁻¹ cm ⁻² Å ⁻¹	Internal error estimate
DQ	array		Data quality flags

2.4 Headers, Keywords, and Relationship to Phase II

As with previous HST instruments, the FITS header keywords in STIS data files store important information characterizing the observations and telemetry received during the observations, and describe the post-observation processing of your dataset. Each keyword follows FITS conventions and is no longer than eight characters. Values of keywords can be integer, real (floating-point), or character string. Many are HST and STIS specific. Knowledge of the keywords and where to find them is an important first step in understanding your data. By examining your file headers, using either **infostis**, **catfits**, **imhead**, **hselect** or **hedit**, you will find detailed information about your data including:

- Target name, coordinates, proposal ID, and other proposal level information.
- Observation and exposure time information such as observation start and duration.
- Instrument configuration information such as detector, grating, central wavelength setting, and filter.
- Readout definition parameters such as binning, gain, and subarray parameters.

- Exposure-specific information such as more detailed timing, world coordinate system information, and fine guidance sensor identification.
- Calibration information such as the calibration switches and reference files used by the pipeline and parameters derived from the calibration, such as image statistics and wavelength shifts.

The easiest way to quickly identify the observational parameters of a given dataset is to run the task **infostis** (see figure 2.1 below), which prints selected header information for STIS FITS images. Wildcard characters or a file list may be used for input (e.g., *.fits or @fitslist).

Figure 2.1: Using infostis to Display Header Keywords

```

cl> infostis o3xi03alm_raw.fits
-----
                                S T I S
-----

      Rootname: O3XI03A1M          Detector: CCD
      Proposal ID: 7071            Obs Type: IMAGING
      Exposure ID: 3.031          Obs Mode: ACQ
                                   Lamp: NONE
      Target Name: GD153-1         Aperture: F28X50LP
      Right Ascension: 12:57:02.3  Filter: Long_Pass
      Declination: +22:01:53.2     Opt Element: MIRVIS
      Equinox: 2000.0              CCD amp: D
                                   Gain: 4
      Axis 1 binning: 1            CR-split: 1
      Axis 2 binning: 1
      Subarray: yes

      Total Exp. Time: 0.3 sec
      Number of imsets: 3

```

The STIS takes CCD and MAMA spectroscopic and imaging data, as well as acquisitions and ACQ/peakups. The keywords relevant for one of these data types will not necessarily be relevant to another. Accordingly, you will find that the header on your particular file contains a unique combination of keywords appropriate for your type of observation. Long definitions for the keywords can also be accessed from the following Web page, which provides detailed explanations of the contents and algorithm for populating the keywords. This site also provides sample headers for different STIS file types:

<http://www.dpt.stsci.edu/keyword>

Keywords that deal with a particular topic, such as the instrument configuration, are grouped together logically throughout the headers. Table 2.1 lists a useful subset of these groups of keywords, indicates the name of the grouping, and where applicable, shows their relationship to the corresponding information from the Phase II proposal.

Table 2.2 summarizes the possible calibration switch keywords, and indicates whether they are present for a particular observation; it also indicates the reference file keyword corresponding to the particular

calibration step. A calibration switch keyword is populated with either OMIT, COMPLETE, or PERFORM. Similarly, table 2.3 summarizes the reference file group of keywords that identify the files used by the pipeline during calibration (see chapter 3 for a detailed description of pipeline processing).

Table 2.1: Selected Keywords and Relationship to Phase II

Keyword	Phase II Equivalent	Description
NEXTEND		Number of image extensions in the file
Target Information (Primary Header)		
TARGNAME	Target_Name	Name of target.
RA_TARG	Position: RA	Right ascension of the target (deg) (J2000).
DEC_TARG	DEC	Declination of the target (deg) (J2000).
Proposal Information (Primary Header)		
PROPOSID		4 digit proposal number
LINENUM	Visit_Number Exposure_Number	Indicates the visit and exposure number from the phase II proposal; format vv-nnnn, visit number, exposure num.
Summary Exposure Information (Primary Header)		
TDATEOBS		UT date of start of first exposure in file (a character string)
TTIMEOBS		UT start time of first exposure in file (a character string).
TEXPSTRT		Start time (MJD) of 1st exposure in file (a real number).
TEXPEND		End time (MJD) of last exposure in the file (a real number).
TEXPTIME	Number_of_Iterations x Time_per_Exposure	Total of science exposure times in an association.
Science Instrument Configuration (Primary Header)		
OBSTYPE		Observation type (IMAGING or SPECTROSCOPIC).
OBSMODE	Opmode	Operating mode (ACQ,ACQ/PEAK,ACCUM,TIME-TAG).
DETECTOR	Config	Detector in use: NUV-MAMA, FUV-MAMA, or CCD.
OPT_ELEM	Sp_Element	Optical element in use (grating name or mirror).
CENWAVE	Wavelength	Central wavelength for grating settings.
APERTURE	Aperture	Aperture name.
FILTER	Aperture	Filter in use.
APER_FOV		Aperture field of view.

Keyword	Phase II Equivalent	Description
PLATESC		Plate scale (arcsec/pixel).
SCLAMP		Lamp status, NONE or name of lamp which is on.
LAMPSET		Spectral cal lamp current value (milliamps).
NRPTEXP	Number_of_Iterations	Number of repeat exposures in set: default 1.
SUBARRAY		Data from a subarray (T) or full frame (F).
CRSPLIT	CR-SPLIT (optional parameter)	Number of cosmic ray split exposures.
Readout Definition Parameters		
SIZAXIS1	SIZEAXIS1	Subarray axis 1 size in unbinned detector pixel.
SIZAXIS2	SIZEAXIS2	Subarray axis 2 size in unbinned detector pixel.
BINAXIS1	BINAXIS1	Axis 1 data bin size in unbinned detector pixel.
BINAXIS2	BINAXIS2	Axis 2 data bin size in unbinned detector pixel.
Engineering Parameters		
CCDAMP	AMP (optional engineering parameter)	CCD amplifier read out (A,B,C,D).
CCDGAIN	GAIN (optional parameter)	Commanded gain of CCD.
Exposure Information (Extension header only)		
DATE-OBS		UT date of start of observation (yyyy-mm-dd).
TIME-OBS		UT time of start of observation (hh:mm:ss).
PA_APER		Position angle of reference aperture.
ORIENTAT		Position angle of image y-axis.
CRPIX1, CRPIX2		x and y coordinates of reference pixel
DGESTAR		FGS ID(F1,F2,F3) concat. with dominant guide star ID.
EXPTIME	Time_per_Exposure/CR-SPLIT or Time_per_Exposure (if not CR-SPLIT)	Exposure time of an individual exposure in a CR-SPLIT or REPEAT OBS series, in seconds (a real number).
EXPSTART		Exposure start time (Modified Julian Date)
EXPEND		exposure end time (Modified Julian Date)

Table 2.2: Calibration Switch Keywords

Keyword	Ref. File Keywords	Explanation	Spectra	Images	CCD	MAMA
STATFLAG	N/A	Calculate image statistics	•	•	•	•
DQICORR	BPIXTAB	Initialize data quality image	•	•	•	•
LORSCORR	N/A	Convert MAMA image to low-res	•	•		•
DARKCORR	DARKFILE	Dark image correction	•	•	•	•
FLATCORR	LFLTFILE	Flatfield corrections	•	•	•	•
	PFLTFILE					
	DFLTFILE					
SGEOCORR	SDSTFILE	Small-scale distortion correction	•	•		•
RPTCORR	N/A	Add individual repeat observations	•	•	•	•
ATODCORR	ATODTAB	A-to-D correction	•	•	•	
BLEVCORR	N/A	Correct for CCD bias level (overscan)	•	•	•	
BIASCORR	BIASFILE	Bias image (structure) correction	•	•	•	
CRCORR	CRREJTAB	Cosmic ray rejection	•	•	•	
EXPSCORR	N/A	Process individual observations after CRCORR	•	•	•	
SHADCORR	SHADFILE	Shutter shading correction	•	•	•	
PHOTCORR	PHOTTAB	Populate header photometric keywords		•	•	•
GEOCORR	IDCTAB	Geometric correction		•	•	•
X2DCORR	SDCTAB	Rectify 2-D spectral image	•		•	•
	DISPTAB					
	INANGTAB					
	APDESTAB					
	MOFFTAB					
	SPTRCTAB					

Keyword	Ref. File Keywords	Explanation	Spectra	Images	CCD	MAMA
X1DCORR	SPTRCTAB	Extract 1-D Spectrum	•		•	•
	XTRACTAB					
SC2DCORR	CDSTAB	2-D scattered light correction	•			•
	ECHSCTAB					
	EXSTAB					
	RIPTAB					
	HALOTAB					
	TELTAB					
	SRWTAB					
BACKCORR	XTRACTAB	Subtract spectral background	•		•	•
DISPCORR	DISPTAB	Apply dispersion solution	•		•	•
	INANGTAB					
	APDESTAB					
	MOFFTAB					
WAVECORR	WAVECAL	Use wavecal to adjust wavelength zeropoint	•		•	•
FLUXCORR	APERTAB	Convert to absolute flux	•		•	•
	PHOTTAB					
HELCORR	N/A	Convert to heliocentric wavelengths	•		•	•
DOPPCORR	N/A	Doppler correction	•			•
GLINCORR	N/A	Global detector non-linearities	•	•		•
LFLGCORR	N/A	Flag pixels for local and global non-linearities	•	•		•

Table 2.3: Reference File Keywords

Keyword	File Suffix	Format	Explanation	Spectra	Images	CCD	MAMA
BPIXTAB	_bpx	Table	Bad pixel	•	•	•	•
ATODTAB	_a2d	Table	A-to-D correction	•	•	•	
CCDTAB	_ccd	Table	CCD parameters	•	•	•	
BIASFILE	_bia	Image	Bias (structure)	•	•	•	
CRREJTAB	_crr	Table	Cosmic ray rejection parameters	•	•	•	
SHADFILE	_ssc	Image	Shutter shading correction	•	•	•	
DARKFILE	_drk	Image	Dark	•	•	•	•
LFLTFILE	_lfl	Image	Low-order flat	•	•	•	•
PFLTFILE	_pfl	Image	Pixel-to-pixel flat	•	•	•	•
DFLTFILE	_dfl	Image	Delta-flat	•	•	•	•
SDSTFILE	_ssd	Image	Small-scale distortion correction	•	•		•
MLINTAB	_lin	Table	Flux linearity	•	•		•
PHOTTAB	_pht	Table	Photometric conversion	•	•	•	•
APERTAB	_apt	Table	Aperture throughput	•	•	•	•
LAMPTAB	_lmp	Table	Template CAL lamp spectra	•		•	•
APDESTAB	_apd	Table	Aperture descriptions	•		•	•
IDCTAB	_idc	Table	Image distortion correction		•	•	•
SDCTAB	_sdc	Table	2-D spectrum distortion correction	•		•	•
INANGTAB	_iac	Table	Incident angle correction	•		•	•
MOFFTAB	_moc	Table	MAMA offset correction	•			•
DISPTAB	_dsp	Table	Dispersion coefficients	•		•	•
SPTRCTAB	_1dt	Table	1-D spectrum trace	•		•	•
XTRACTAB	_1dx	Table	1-D extraction parameter	•		•	•
WAVECAL	_wav	Image	Wavelength calibration	•		•	•
PCTAB	_pct	Table	Photometry correction	•		•	•

Keyword	File Suffix	Format	Explanation	Spectra	Images	CCD	MAMA
WCPTAB	_wcp	Table	Wavecal parameters	•		•	•
CDSTAB	_cds	Table	Cross-disperser scattering	•			•
ECHSCTAB	_ech	Table	Echelle scattering	•			•
EXSTAB	_exs	Table	Echelle cross-dispersion Scattering	•			•
RIPTAB	_rip	Table	Echelle ripple	•			•
HALOTAB	_hal	Table	Detector halo	•			•
TELTAB	_tel	Table	Telescope point spread function	•			•
SRWTAB	_srw	Table	Scattering reference wavelengths	•			•

2.5 Error and Data Quality Arrays

The STIS pipeline propagates both statistical errors and data quality flags throughout the calibration process. These are then combined from both the science data and the reference file data to produce triplets of science, error, and data quality in the calibrated data and extracted spectra.

Note that both the error and data quality image extensions may be represented with a null array (i.e., NAXIS=0 following STScI conventions) if all the values are identically zero (see table 2.1 and [STIS ISR 95-06](#)).

2.5.1 The Error Array

The error array contains an estimate of the statistical error at each pixel. In the raw file, the error array is empty. The first step of **calstis** is to calculate the error array for the input data. This raw data error is simply given as:

$$\sigma_{DN} = \frac{\sqrt{\sigma_c^2 + R \cdot g}}{g}$$

where

- R is the observed data number (counts) minus the electronic bias of the pixel; note that the electronic bias is zero for the MAMA.
- g is the gain factor; that the gain is unity for MAMA observations.

- σ_c is the read noise in electrons for CCD observations (it is set to zero for MAMA observations).

The bias, gain factor, and read noise are read from the CCD parameters reference file for CCD data (a `_ccd` file specified by keyword CCDTAB). As the data are calibrated through the **calstis** pipeline, the statistical errors are propagated through, reflecting both the science and reference file errors (see *STIS ISR 98-06*, section 6, for details).

2.5.2 Data Quality Flagging

Data quality flags are assigned to each pixel in the data quality extension. Each flag has a true (set) or false (unset) state. Flagged conditions are set as specific bits in a 16-bit integer word. For a single pixel, this allows for up to 15 data quality conditions to be flagged simultaneously, using the bitwise logical OR operation. Note that the data quality flags cannot be interpreted simply as integers but must be converted to base 2 and interpreted as flags. Table 2.1 gives the specific conditions that are flagged, depending on the states of different bits (i.e., being on or off).

The raw data quality files will be filled only when there are missing (data lost) or dubious (software error) data. If no such errors exist, initialization will produce an empty data quality file whose header has NAXIS=0.

These flags are set and used during the course of calibration, and may likewise be interpreted and used by downstream analysis applications.

Table 2.1: STIS Data Quality Flags

FLAG Value	Bit Setting ¹	Quality Condition Indicated
1	0000 0000 0000 0001	Error in the Reed-Solomon decoding (an algorithm for error correction in digital communications).
2	0000 0000 0000 0010	Lost data replaced by filled values.
4	0000 0000 0000 0100	Bad detector pixel (e.g., bad column or row, mixed science and bias for overscan, or beyond aperture).
8	0000 0000 0000 1000	Data masked by occulting bar.
16	0000 0000 0001 0000	Pixel having dark rate > 5 sigma times the median dark level.
32	0000 0000 0010 0000	Large blemish, depth > 40% of the normalized p-flat.
64	0000 0000 0100 0000	Reserved.
128	0000 0000 1000 0000	Reserved.

FLAG Value	Bit Setting ¹	Quality Condition Indicated
256	0000 0001 0000 0000	Saturated pixel, count rate at 90% of max possible—local non-linearity turns over and is multivalued; pixels within 10% of turnover and all pixels within 4 pixels of that pixel are flagged
512	0000 0010 0000 0000	Bad pixel in reference file.
1024	0000 0100 0000 0000	Small blemish, depth between 40% and 70% of the normalized flat. Applies only to MAMA p-flats at present.
2048	0000 1000 0000 0000	>30% of background pixels rejected by sigma-clip, or flagged, during 1-D spectral extraction.
4096	0001 0000 0000 0000	Extracted flux affected by bad input data.
8192	0010 0000 0000 0000	Data rejected in input pixel during image combination for cosmic ray rejection.
16384	0100 0000 0000 0000	Reserved.

1. The most significant bit is on the left in this representation.

2.6 STIS Coordinate Systems

References to multiple coordinate systems appear in the headers of STIS data, tied to the spacecraft frame, the user frame, and the aperture frame. We briefly explain here the relationships among these coordinate systems, and show them in figure 2.1. These coordinate systems are discussed in detail in STIS TIR 99-02, available upon request from the Help Desk.

The three coordinate systems of interest are the:

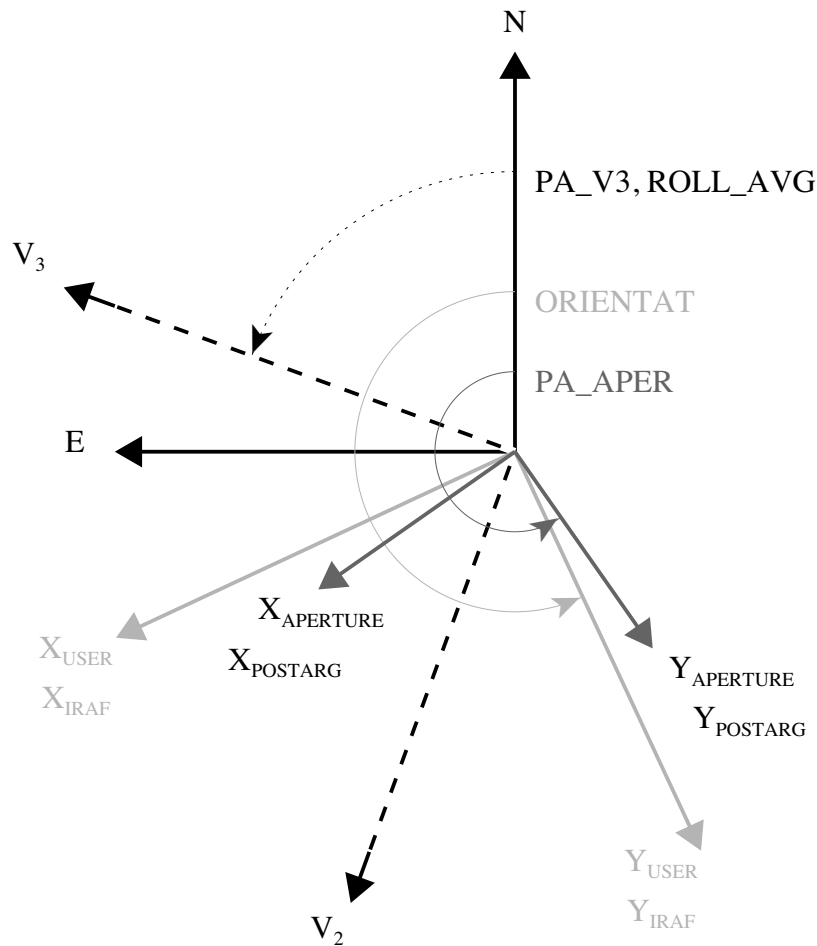
- Vehicle Frame (V_1, V_2, V_3): The right-handed coordinate system for the telescope, with V_1 pointing in the direction the telescope is looking.
- User Frame ($X_{\text{user}}, Y_{\text{user}}$): This is the user frame, aligned with the detector. The coordinate system in IRAF is aligned with the user frame.
- Aperture Frame ($X_{\text{aperture}}, Y_{\text{aperture}}$): This frame is aligned with the slit. POS-TARG movements are aligned with the aperture frame.

The angles associated with these frames that appear in the headers of STIS data files are:

- PA_V3: North to V_3 through East, measured at the center of the HST focal plane (in the `_spt` header).

- **ROLL_AVG**: North to V_3 through East, measured at the position of the STIS field in the HST focal plane (in the `_jit` header, computed).
- **PA_APER**: North to Y_{aperture} through East, measured at the aperture reference (in science header). $PA_{V3} - PA_{APER}$ is either 225 or -135 degrees, with small variations (a few tenths of a degree).
- **ORIENTAT**: North to Y_{user} through East, measured at the aperture reference (in science header). It can differ from PA_{APER} by up to ~ 1.5 degrees (e.g., for the long slits with offset occulting bars). Note that this is not the same angle as the **ORIENT** specified in Phase II, which gives the position angle of the U_3 axis, where $U_3 = -V_3$. For further information on slit orientation while planning STIS observations, see figure 3.2 and "Fixing Orientation on the Sky" in Chapter 11 of the *STIS Instrument Handbook*,

Figure 2.1: STIS Coordinate Systems



2.7 STIS Paper Products

The calibration pipeline currently produces *paper products* that summarize the data obtained. General Observers (GO) can receive these, by request, a few weeks after their data are taken (for up to 6 months after the observations). Archival observers can recreate these paper products by retrieving all of the science and jitter data for a particular observation (see chapter 1 of the HST Introduction) and using the STSDAS **pp_dads** task (at the IRAF prompt type: **pp_dads *.fits**).

STIS paper products are designed to summarize a set of exposures for a single visit. A given page in the STIS paper product falls into one of two categories: a visit-level page or an exposure-level page. Below, we list the individual pages contained in the STIS paper products; figure 2.1 through figure 2.4 provide samples of the paper products pages. Users are encouraged to read *ISR STIS 97-11*, which provides more details about the STIS paper products.

Visit-level Pages

- **Cover Page:** A cover page containing the proposal ID, the visit number, the PI's name, and the proposal title.
- **Explanatory Notes** (figure 2.1): A set of notes explaining the paper products and the information they contain.
- **Target List** (figure 2.2): A table listing the targets for a given set of observations.
- **Observation list:** A table recapping the proposal information for each exposure for the set of observations being summarized, including three processing and data quality flags, corresponding to OK, not OK, and unknown status.
- **Optional Parameters:** A table listing the proposal-level optional parameters for the set of observations being summarized.
- **Statistics:** A table of simple statistics for the set of observations being summarized to allow for a quick comparison among observations.

Exposure-level Pages

- **Exposure Plots:** This is a graphical representation of the data contained in each exposure. Plots are specific to a particular instrument configuration and observing mode. In some cases, more than one plot is produced. The types of exposure plots are: ACQ image plot (figure 2.3), ACQ/PEAK image plot, image plot (figure 2.4), rectified two-dimensional spectral image plot, one-dimensional extracted spectrum plot, and time-series plot.

- **Data Quality Summary:** Consists of a comprehensive summary of the spacecraft performance, pipeline processing status, and calibration data quality for each exposure.
- **Calibration Reference File Summary:** Is a summary of the calibration processing switches and reference files used to process each exposure.

Figure 2.1: Explanatory Notes

STIS

Description of Visit Summaries

Target List

The Target List contains the target name, the coordinates for the target as calculated by the ground system based on the target information taken from the proposal, and the text description of the target given in the proposal. Note that the coordinates listed represent the predicted position of the target in the sky and do not give the pointing of HST at the time of the observation.

Observation List with Data Quality Flags

The Observation List contains information that uniquely identifies individual exposures as specified in the observing proposal. Additionally, the status of the spacecraft and ground-system performance during the execution of the observation are summarized by the Procedural Quality Flags:

OBS	Status of the performance of HST.
PROC	Status of the pipeline processing of the observations.
CAL	Status of the reference data used in calibration.

Symbols used to indicate the status of the Procedural Quality are:

○	OK.
●	Not OK-Refer to the Data Quality Summary for details.
Blank	Status unknown.

Observation List-Optional Parameters

The Observation List contains additional instrument configuration information. Entries in the table reflect the values of the Optional Parameters specified in the observing proposal.

Observation Statistics

This Observation Statistics table contains a simple set of statistics of the raw (or flat-fielded) data for the observations.

Description of Exposure Summaries

Plots for Each Exposure

Plots are created for each exposure. Gray-scale or line plots are produced as appropriate for the instrument configuration and observing mode for each exposure. Exposure information taken from the headers of the data files is also provided.

Data Quality Summary for Each Exposure

The Data Quality Summary contains details of problems flagged by the Data Quality flags. Exposure information taken from the headers of the data files is also provided.

Calibration status summary for each exposure

The calibration summary gives detailed information about the calibration of the observations. Individual calibration steps are listed with completion status. Reference files used are listed by name and information about the pedigree of the calibration data is provided.

Need Help?

Send e-mail to your contact scientist or help@stsci.edu

Space Telescope Science Institute: Wed 15:07:38 25-Jun-97, Page 2

Figure 2.2: Target List

Visit: 03

Proposal: 7605

STIS

Target List

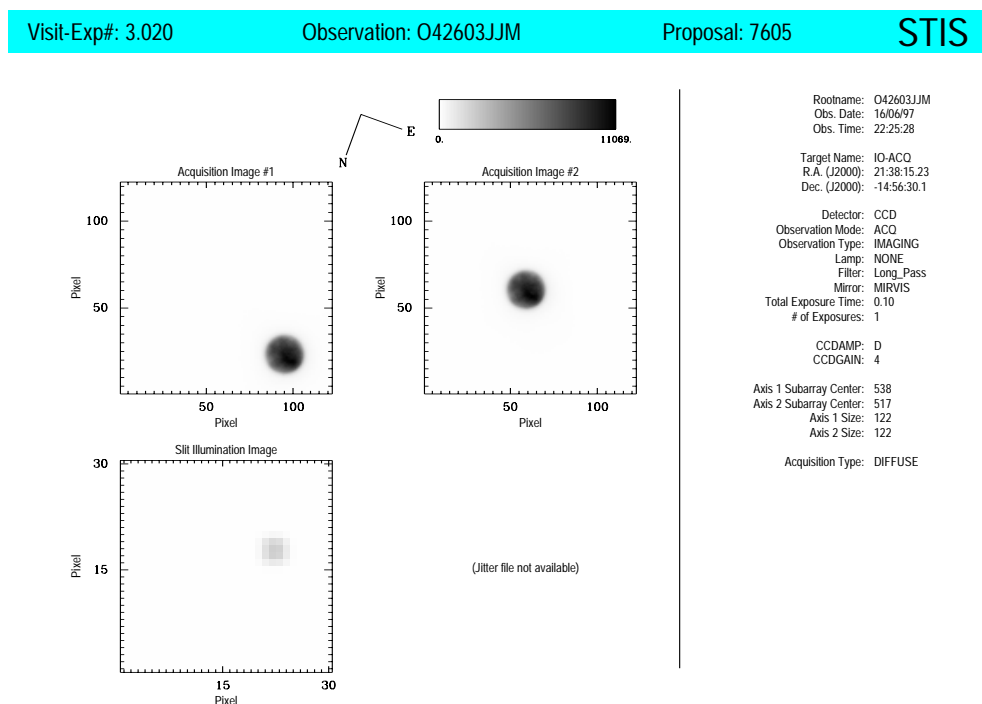
Target Name	R.A. (J2000)	Dec. (J2000)	Description
IO-ACQ	21:38:15.24	-14:56:30.0	(N/A)
NONE	0:00:00.00	0:00:00.0	(N/A)

Observation List

Visit-Exp#	Rootname	Target Name	Detector	Operating Mode	Aperture	Optical Element	Centwave (Å)	Total Exp Time (s)	# of Exp	File	Quality Flags
											Obs Proc Cal
3.010	O42603JIM	IO-ACQ	CCD	ACCUM	F28X50LP	MIRVIS	(N/A)	0.1	1	flt	○ ○
3.020	O42603JJM	IO-ACQ	CCD	ACQ	F28X50LP	MIRVIS	(N/A)	0.1	1	raw	○ N/A
3.030	O42603JKM	IO-ACQ	CCD	ACCUM	F28X50LP	MIRVIS	(N/A)	0.1	1	flt	○ ○
3.040	O42603020	NONE	CCD	ACCUM	52X0.2	G750L	7751	30.0	2	flt	○ ○
3.050	O42603030	IO-ACQ	CCD	ACCUM	52X0.2	G750L	7751	24.0	2	crj	○ ○
3.070	O42603040	IO-ACQ	CCD	ACCUM	52X0.2	G750L	7751	24.0	2	crj	○ ○
3.080	O42603050	IO-ACQ	CCD	ACCUM	52X0.2	G750L	7751	24.0	2	crj	○ ○
3.090	O42603060	IO-ACQ	CCD	ACCUM	52X0.2	G750L	7751	24.0	2	crj	○ ○
3.100	O42603070	NONE	CCD	ACCUM	52X0.2	G750L	7751	30.0	2	flt	○ ○
3.160	O42603080	NONE	CCD	ACCUM	52X0.2	G750M	8561	150.0	2	flt	○ ○
3.170	O42603090	IO-ACQ	CCD	ACCUM	52X0.2	G750M	8561	36.0	2	crj	○ ○
3.190	O426030A0	IO-ACQ	CCD	ACCUM	52X0.2	G750M	8561	36.0	2	crj	○ ○
3.200	O426030B0	IO-ACQ	CCD	ACCUM	52X0.2	G750M	8561	36.0	2	crj	○ ○
3.210	O426030C0	IO-ACQ	CCD	ACCUM	52X0.2	G750M	8561	36.0	2	crj	○ ○
3.220	O426030D0	NONE	CCD	ACCUM	52X0.2	G750M	8561	150.0	2	flt	● ○ ○

Space Telescope Science Institute, Wed 15:07:38 25-Jun-97, Page 3

Figure 2.3: ACQ Plot



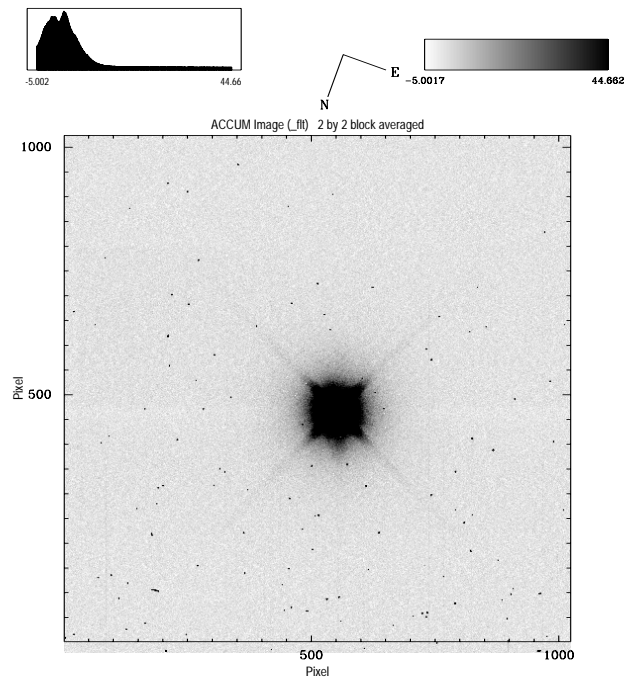
Space Telescope Science Institute, Wed 15:07:38 25-Jun-97, Page 11

Figure 2.4: Image Plot

Visit-Exp#: 3.010

Observation: O42603JIM

Proposal: 7605

STIS

Rootname: O42603JIM
 Obs. Date: 16/06/97
 Obs. Time: 22:24:01

Target Name: IO-ACO
 R.A. (J2000): 21:38:15.24
 Dec. (J2000): -14:56:30.0

Detector: CCD
 Observation Mode: ACCUM
 Observation Type: IMAGING
 Lamp: NONE
 Filter: Long_Pass
 Mirror: MIRVIS
 Total Exposure Time: 0.10
 # of Exposures: 1

CCDAMP: D
 CCDGAIN: 1

Pixels: 1048576
 # Good Pixels: 740906
 Good Minimum: -205.17
 Good Maximum: 30674.50
 Good Mean: 10.44
 Min S/N: -35.04
 Max S/N: 134.32
 Mean S/N: 0.34

STIS Calibration

In this chapter. . .

3.1 Pipeline Processing Overview / 3-1
3.2 Structure of calstis / 3-3
3.3 Data Flow Through calstis / 3-10
3.4 Descriptions of Calibration Steps / 3-16
3.5 Recalibration of STIS Data / 3-38
3.6 Updates to calstis / 3-44

This chapter describes how the STIS pipeline at STScI calibrates incoming STIS data. We begin with a high-level overview of the STIS calibration process, and subsequent sections describe the pipeline calibration steps and methodology in successively greater detail. We then discuss several reasons why you might want to recalibrate your data, and how to use the pipeline tasks for recalibration.

3.1 Pipeline Processing Overview

Science data obtained with the STIS are received from the Space Telescope Data Distribution Facility and sent to the STScI pipeline, which unpacks the data, extracts keywords from the telemetry stream, reformats the data, and repackages them into raw, uncalibrated, but scientifically interpretable data files. When a user requests data from the Hubble Data Archive, the raw files are then calibrated by the On The Fly Reprocessing (OTFR) system. What is described in this chapter is **calstis**, the program that performs the calibration of the science data and is available to the community as part of the STSDAS package.

Conceptually, **calstis** is several pipelines in one, reflecting the complexity and diversity of STIS observing modes. Your STIS data will have been calibrated to different levels, depending on their nature:

- ACQs, ACQ/PEAKs, and all available-mode data are not calibrated by **calstis**; you will get only the raw data from observations taken in these modes.
- All other science data are processed through basic two-dimensional image reduction (**basic2d**), which includes such things as bias subtraction, dark subtraction, flatfielding, and linearity correction. In the case of CCD CR-SPLIT or REPEATOBS data, your data will also be passed through cosmic ray rejection, although this is done by a separate task, called **ocrreject**.
- Data taken in TIMETAG mode are output both as a raw uncalibrated event stream (to a FITS binary table), and as an ACCUM mode image, which then passes through standard calibration.
- For MAMA data, the input raw data format is 2048 x 2048 (so called “highres” pixels), while the calibrated data are binned by the pipeline to 1024 x 1024 native format pixels (see section 3.4.16).
- All spectroscopic data (exclusive of slitless spectra or long slit echelle data) are then passed through spectroscopic reduction, to produce flux and wavelength calibrated science data. In the case of long slit data, a two-dimensional rectified spectral image is produced. For echelle data, a two-dimensional echelle cross-order scattering correction is also applied. For all supported spectroscopic modes, a one-dimensional background subtracted aperture extracted spectrum is produced.

See chapter 2 for the naming conventions of the various input, intermediate, and output calibrated files.

As with the calibration pipelines for the other HST instruments, the specific operations that are performed during calibrations are controlled by *calibration switches*, which are stored in the image headers as KEYWORD=VALUE pairs. Any given step in the calibration process may require the application of zero, one, or more *calibration reference files*, the names of which are also found in the image header. The names of the keywords containing the switches and reference file names were introduced in the previous chapter; section 3.3 will outline the role these keywords play in the data reduction, and are given in section 3.4. The path your data files take through the pipeline is determined by the calibration switches set in the header of the raw data, which in turn depends directly on the type of data you have.

A few other general comments are in order. It is important to note that some of the STIS calibration reference data are obtained contemporaneously with the science observations. These data may be used

to refine the calibration process (as with the automatic wavecal), or may require you to replace a default calibration reference file with a contemporaneously obtained one, as in the case of a CCD near infrared (NIR) fringe flat. The details of how these contemporaneous calibration files are used in **calstis** can be found in section 3.4. The STIS (and NICMOS) pipelines are also unusual in that they are re-entrant. That is, a user running **calstis** off-line may choose to reprocess STIS data partially, performing one or more of the intermediate steps without re-exercising the complete **calstis** pipeline, for instance to perform cosmic-ray rejection or one dimensional spectral extraction. Refer to section 3.5 for the mechanics (and restrictions) of this kind of processing. Finally, as with other HST pipelines, **calstis** propagates statistical errors and tracks data quality flags throughout the calibration process.

3.2 Structure of calstis

calstis consists of a series of individual modules that:

- Orchestrate the flow through the pipeline.
- Perform the initial tasks of basic two-dimensional image reduction (e.g., overscan subtraction, bias subtraction).
- Reject cosmic rays from CCD data.
- Perform the remaining tasks of basic two-dimensional image reduction (e.g., dark subtraction, flatfielding).
- Process the contemporaneously obtained wavecal data to obtain the zeropoint shifts in the spectral and spatial directions.
- Perform spectroscopic wavelength and flux calibration.
- Sum any CR-SPLIT and REPEATOBS exposures.

Table 3.1 describes in more detail the individual modules in **calstis** and what they do. The IRAF task that can be used to run a particular segment of the pipeline independently is also provided (see section 3.5.2).

Below, we present a series of flow charts that provide a more complete overview of the processing of data through the **calstis** pipeline, starting with the fundamental step of two-dimensional image reduction.

Figure 3.1 shows the route taken by CCD and MAMA data.

Table 3.1: Calstis Module Description Summary

IRAF Task	Description of Processing Step	Module ¹
Full Pipeline		
calstis	“Wrapper” program calls each of the calstis tasks as needed , according to the switches set in the <i>primary header</i> of the input file. The calstis constituent tasks can instead be executed independently when recalibrating.	<i>calstis0</i>
Basic 2-D image Reduction		
basic2d	Basic 2-D image reduction. This step includes overscan subtraction, bias subtraction, dark subtraction, flatfielding, initializing the data quality array from the bad-pixel table, assigning values to the error array, and computing some simple statistics.	<i>calstis1</i>
Cosmic Ray Rejection		
ocreject	Detect and remove cosmic rays in CCD data. This module identifies cosmic rays (by optionally flagging them in the input file) for multiple images taken at the same pointing. The input images are then co-added, resulting in an image with cosmic rays removed.	<i>calstis2</i>
Contemporaneous Wavecal Processing		
wavecal	Determine MSM offset from wavecal. This step is used in conjunction with calstis7, calstis11, and calstis12. Its purpose is to find the offset of the spectrum from the expected location, owing to nonrepeatability of the mode select mechanism. The shift is written into the SCI extension header of the input wavecal image.	<i>calstis4</i>
	Subtract science image from wavecal. For CCD wavecal observations taken with the HITM system prior to 1998-Nov-9, the detector is exposed to both the wavecal and the science target (after this date, the external shutter would be closed). This task reads both the wavecal and science files and subtracts the science data from the wavecal. Following this step, calstis4 can be used to determine the spectral shift.	<i>calstis11</i>
	Write spectral shift value to science header. A series of science images (i.e., CRSPLIT or REPEATOBS) and wavecals may have been taken, with the wavecals interspersed in time among the science images. For each image in the science file, this task linearly interpolates the wavecals to the time of the science image, and then writes the keyword values for the spectral shift to the science header.	<i>calstis12</i>
Spectroscopic Calibration and Extraction		
x1d	1-D spectral extraction. This task is most appropriate for observations of a point source. A spectrum is extracted along a narrow band, summing over the cross-dispersion direction and subtracting background values to produce a 1-D array of fluxes for each spectral order. An array of wavelengths is generated, with each output spectrum written to a separate row of a FITS binary table, together with the arrays of the gross, net, and background count rates. One output table is generated for each image in a series of REPEATOBS data. For echelle observations, x1d is called iteratively, to correct for scattered light.	<i>calstis6</i>
x2d	2-D rectification. This task performs geometric correction for imaging or long-slit spectroscopic data. It extracts a 2-D spectrum, linear in both wavelength and spatial directions.	<i>calstis7</i>
Sum Images		
	Sum repeatobs data. This task adds together (pixel-by-pixel) multiple MAMA images, taken and combined into one input FITS file. This would not normally be used for CCD data because they would already have been combined for cosmic ray rejection.	<i>calstis8</i>

1. Referenced in the trailer file.



Note that although “Reject Cosmic Rays” is included in the “Basic 2-D” tasks of figure 3.1, the basic2d task itself does not include cosmic ray rejection. Cosmic ray rejection is performed by ocrreject. To perform basic 2-D reduction on CCD data, basic2d is called for the calibration steps leading up to cosmic ray rejection, then ocrreject is called, and then basic2d is called for the remaining steps in the basic 2-D reduction process. These tasks are discussed in more detail later in this chapter.

The calibration beyond the basic 2-D image processing depends upon whether the data are obtained in imaging or spectroscopic mode. For imaging modes (figure 3.2), the primary operations are geometric distortion correction and photometric calibration, and a summation of multiple MAMA exposures if $\text{NRPTXP} > 1$. The output is a geometrically rectified image with header keywords that specify the photometric calibration. When geometric correction is not applied, the output will be flatfielded data with suffix `_crj`, `_flt`, or `_sfl`.

For spectroscopic exposures, **calstis** will process the associated wavelength calibration (wavecal; figure 3.3) exposure to determine the zero point offset of the wavelength and spatial scales in the science image, thereby correcting for the lack of repeatability of the mode select mechanism (MSM) or for thermal drift. The accompanying wavecal observations are stored in the `rootname_wav.fits` file.

Two-dimensional spectral processing (figure 3.4) produces a flux-calibrated, rectified spectroscopic image with distance along the slit running linearly along the y axis and dispersion running linearly along the x axis.

One-dimensional spectral extraction produces a one-dimensional spectrum of flux versus wavelength (`rootname_x1d.fits` or `rootname_sx1.fits`), uninterpolated in wavelength space, but integrated across an extraction aperture in the spatial direction.

Figure 3.1: Basic 2-D Image Reduction (First Step in Subsequent Flowcharts)

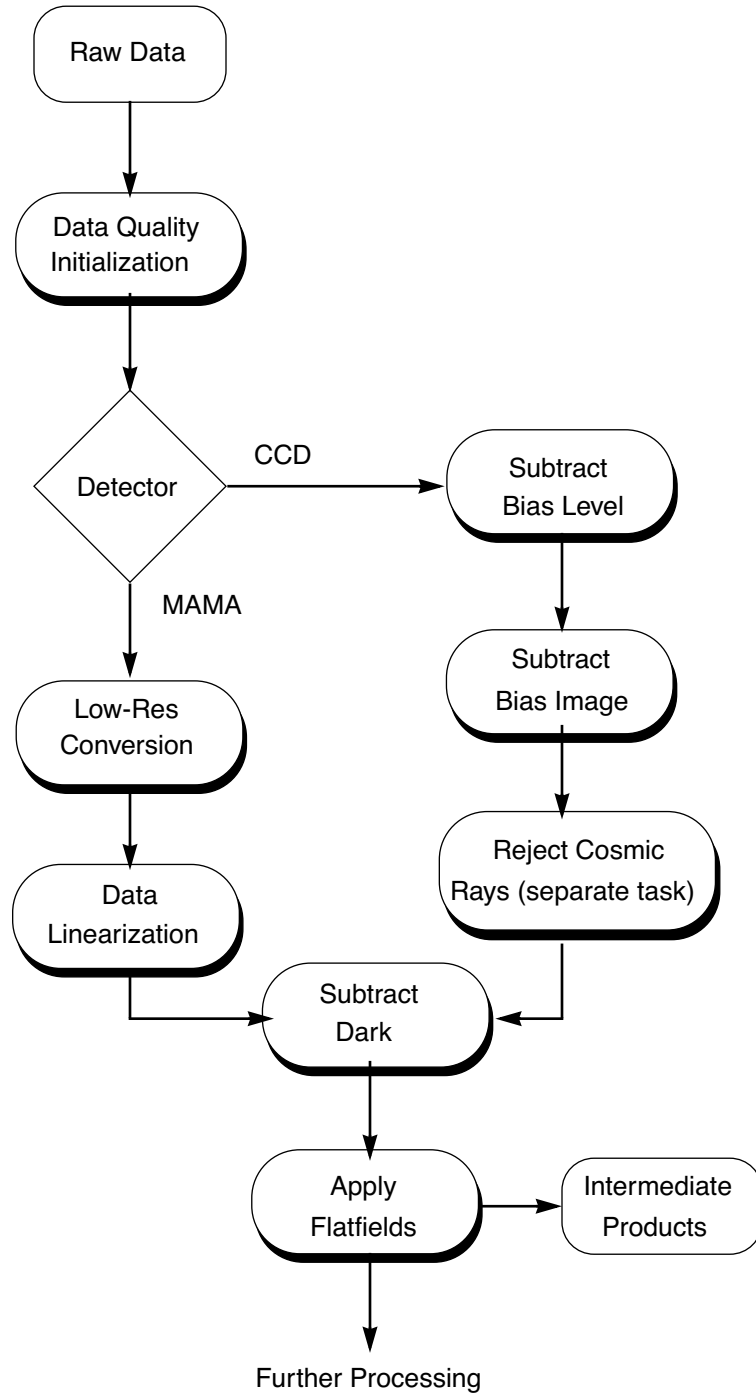
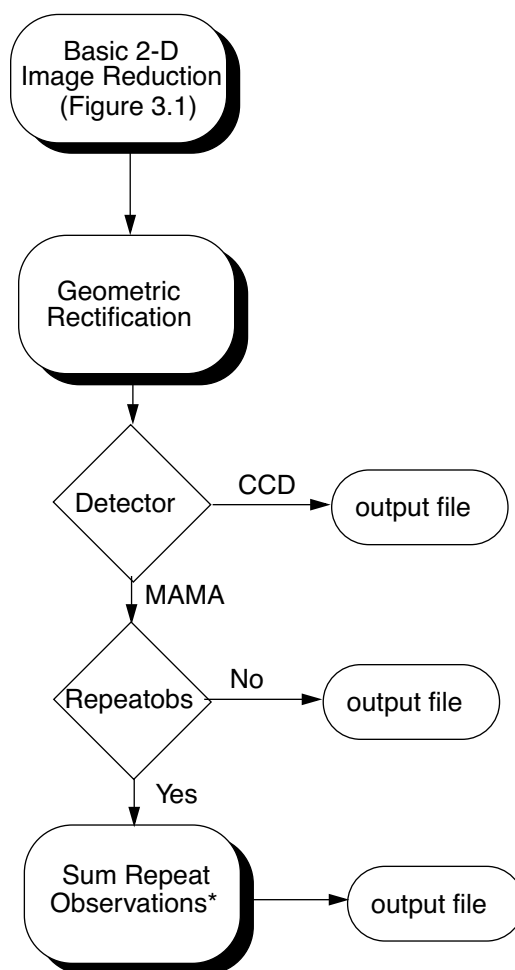


Figure 3.2: Schematic of calstis for Secondary Image Processing



*For CCD observations, exposures are summed only if the number of exposures does not exceed 30.

Figure 3.3: Schematic of calstis for Contemporaneous Wavecal

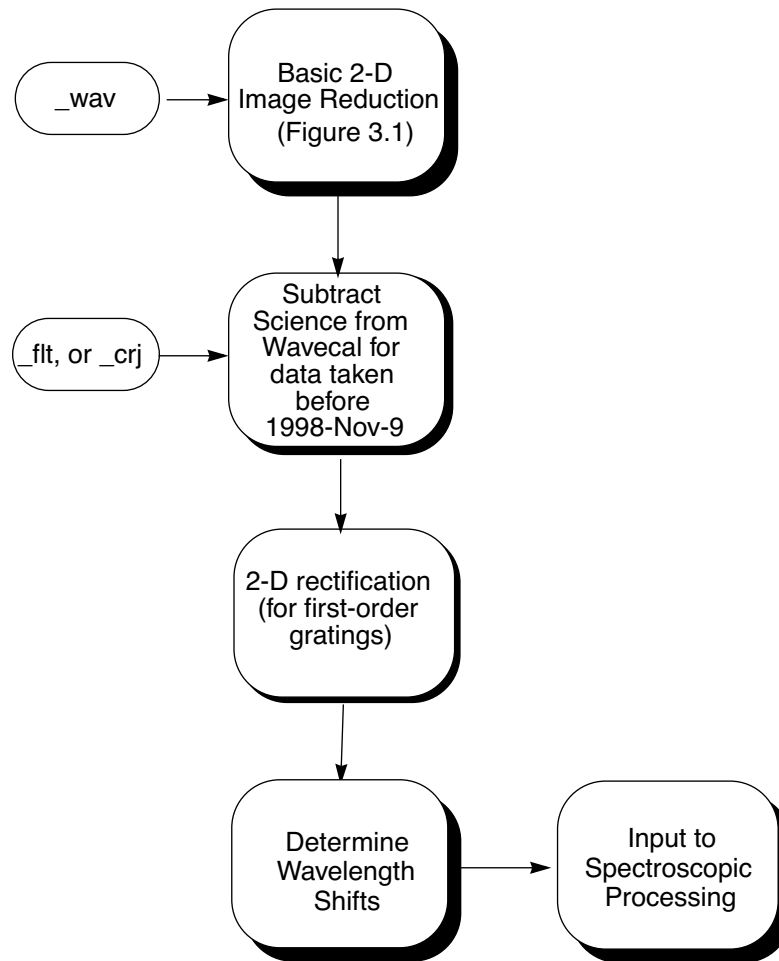
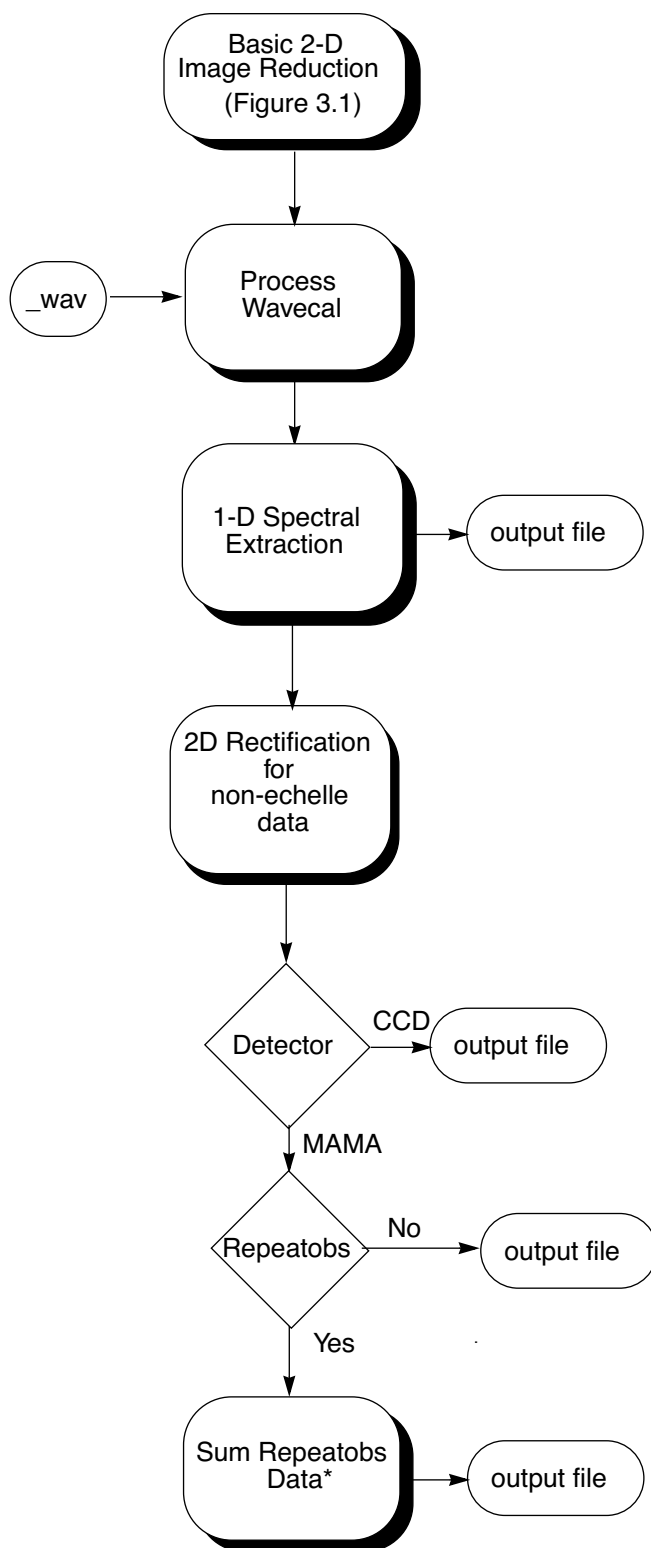


Figure 3.4: Schematic of calstis for Spectroscopic Data



*For CCD observations, exposures are summed only if the number of exposures does not exceed 30.

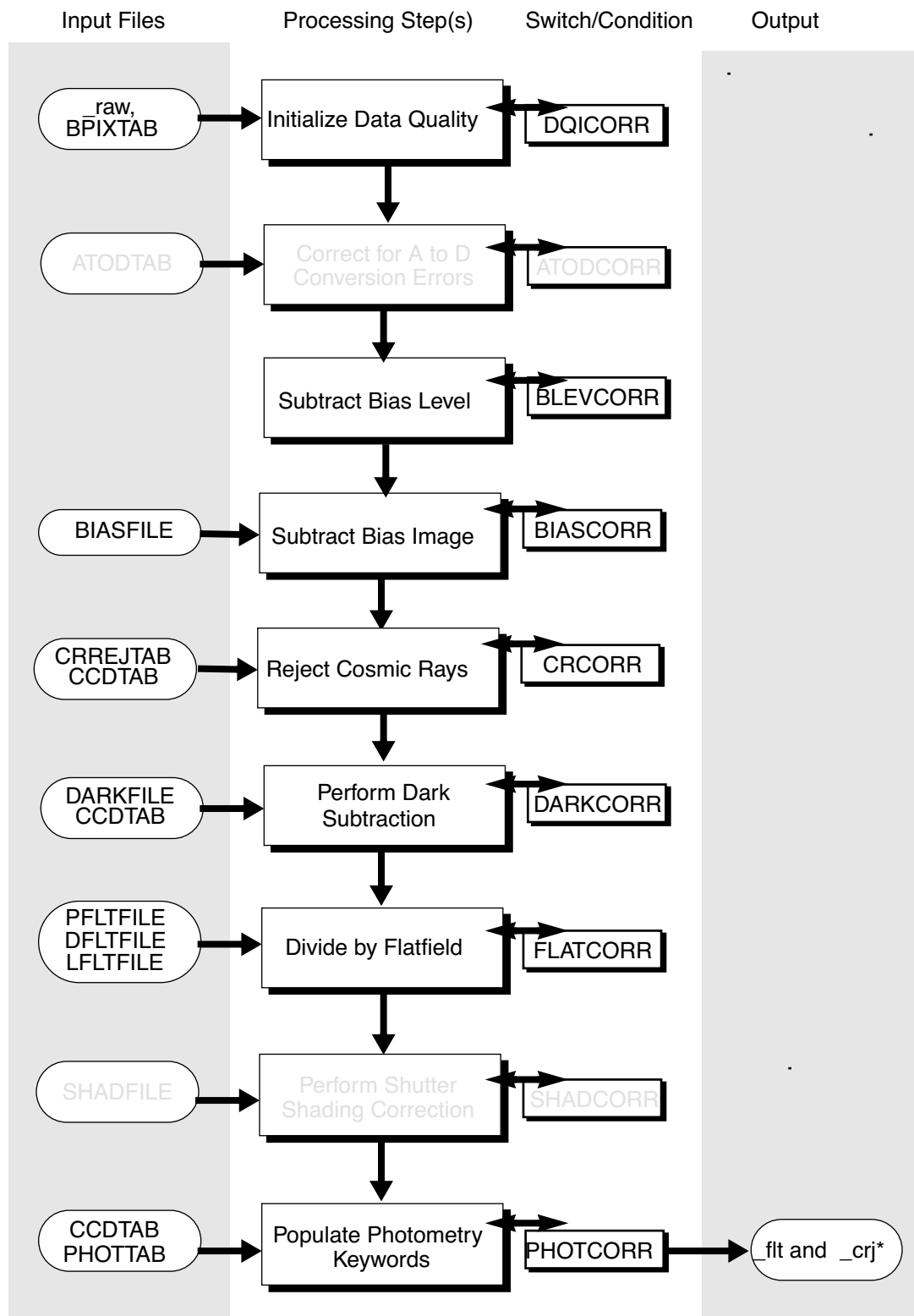
3.3 Data Flow Through calstis

This section details the data flow through the **calstis** pipeline for each calibrated operating mode, showing the switches, the reference file inputs, the science file inputs, and the output products. These details are shown as flow charts in figure 3.1 through figure 3.7. The next section describes the tasks corresponding to the various calibration switches.



Note that some tasks are not implemented at this time. These appear in grey text in the flow charts, and are noted in the descriptions given in the subsequent section.

Figure 3.1: 2-D CCD Data Reduction Common to Imaging and Spectroscopy



*The _flt files contain individual images where the cosmic rays are flagged but not removed. The _crj file is created only if CR-SPLIT>1; it is a combined, flat-fielded image where the cosmic rays have been removed.

Figure 3.2: 2-D MAMA Data Reduction Common to Imaging and Spectroscopy

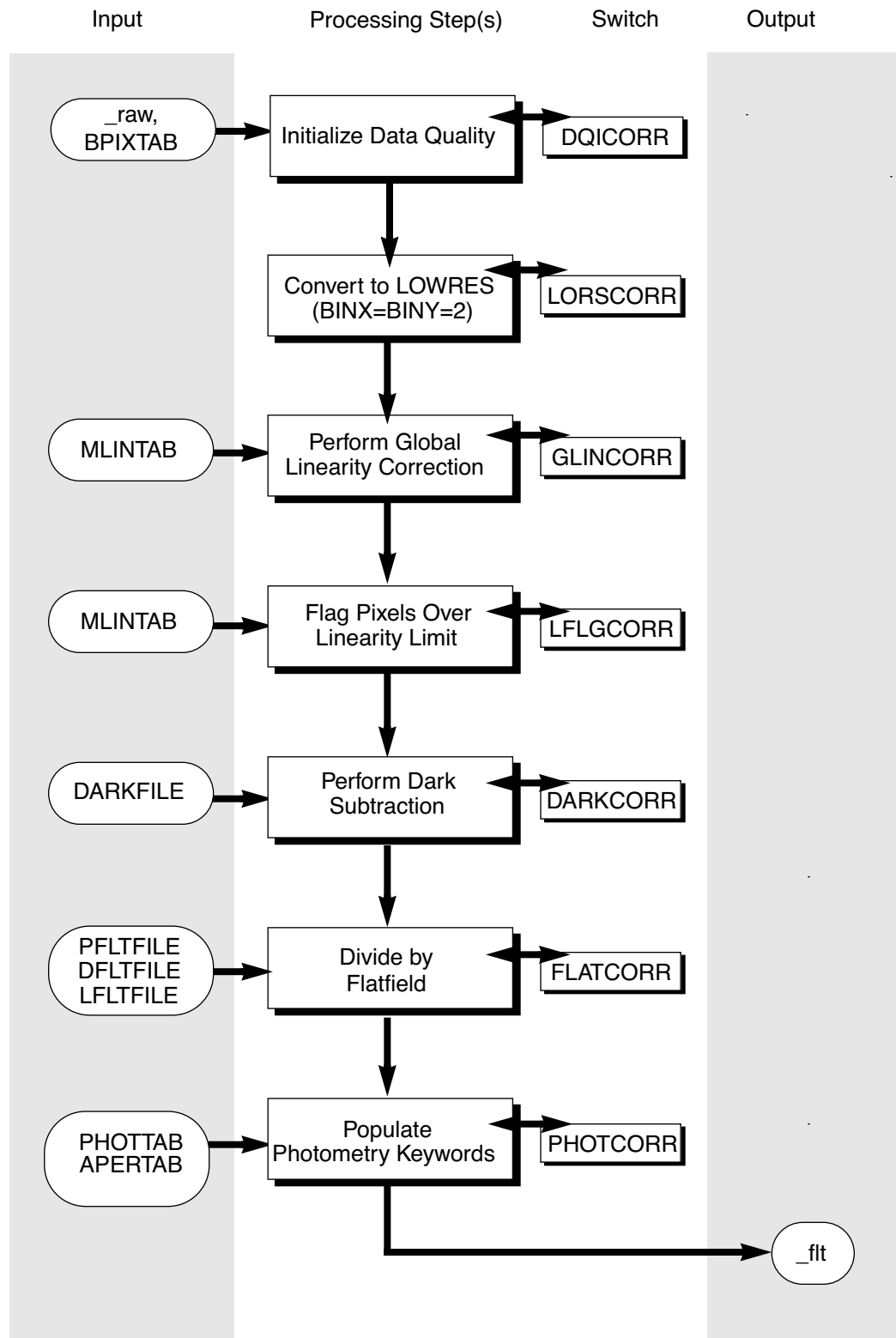


Figure 3.3: Flow of Imaging Data through Calstis-7, 2D Rectification

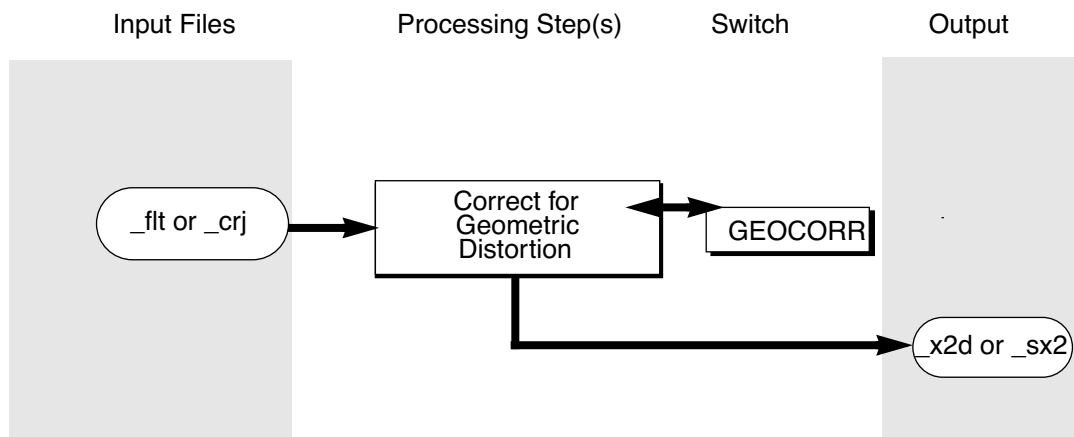


Figure 3.4: Flow of Spectroscopic Data through Calstis-4, wavecal Processing

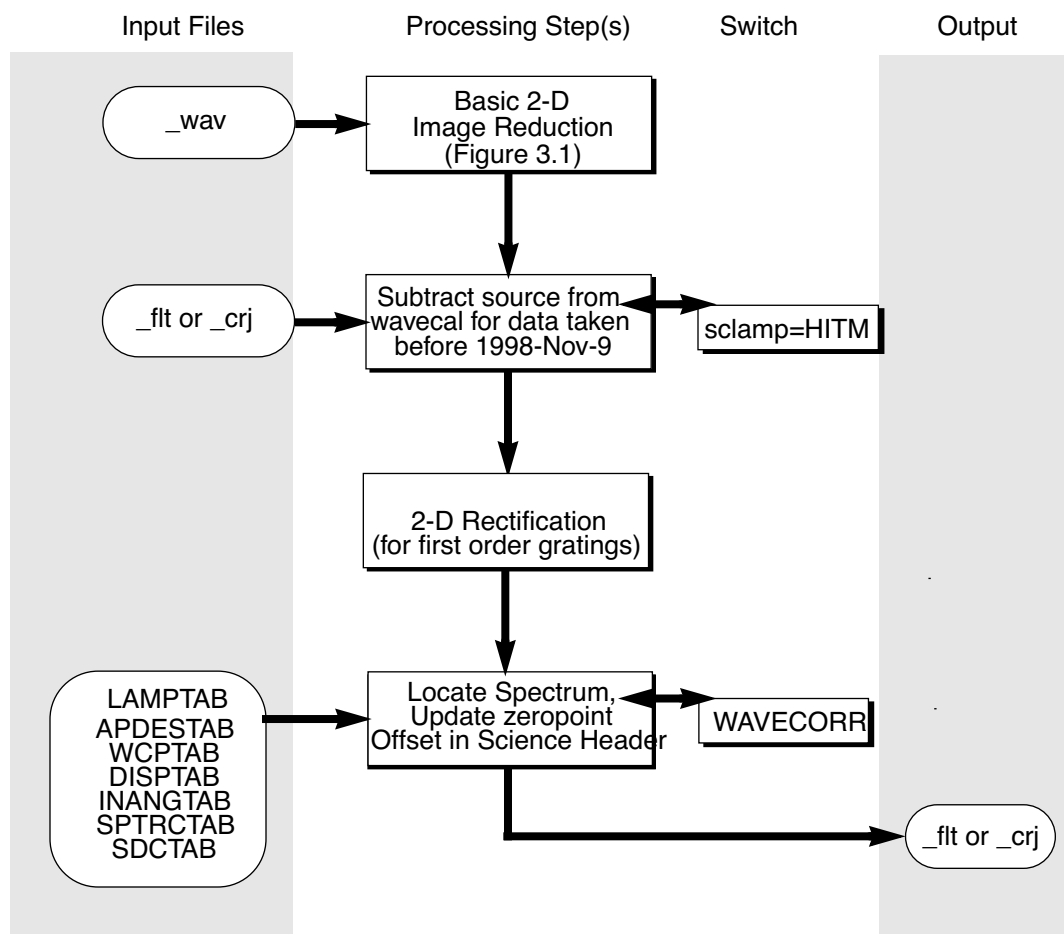


Figure 3.5: Flow of Spectroscopic Data through calstis-6, 1-D Extraction

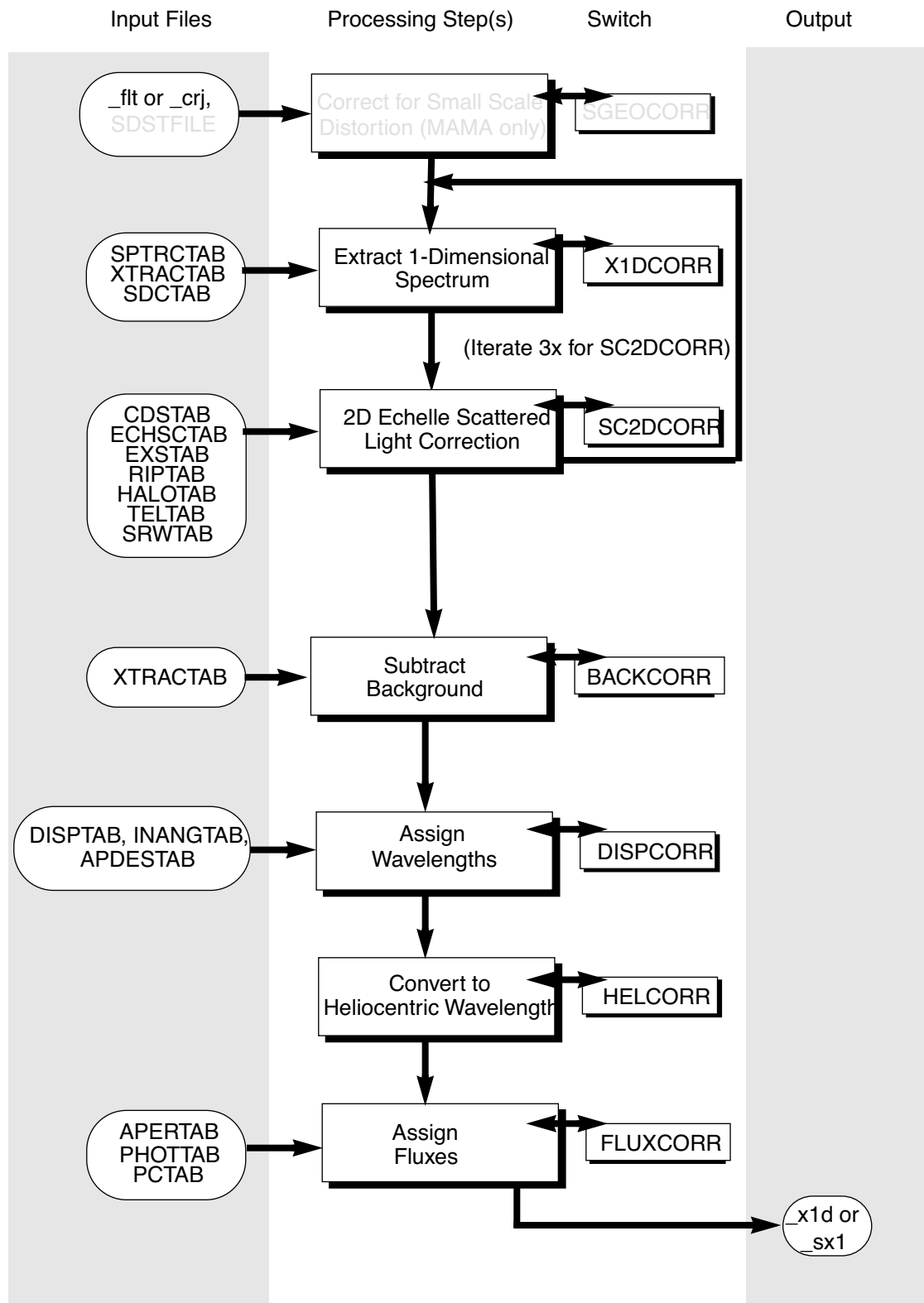


Figure 3.6: Flow of Spectroscopic Data through calstis-7, 2-D Rectification

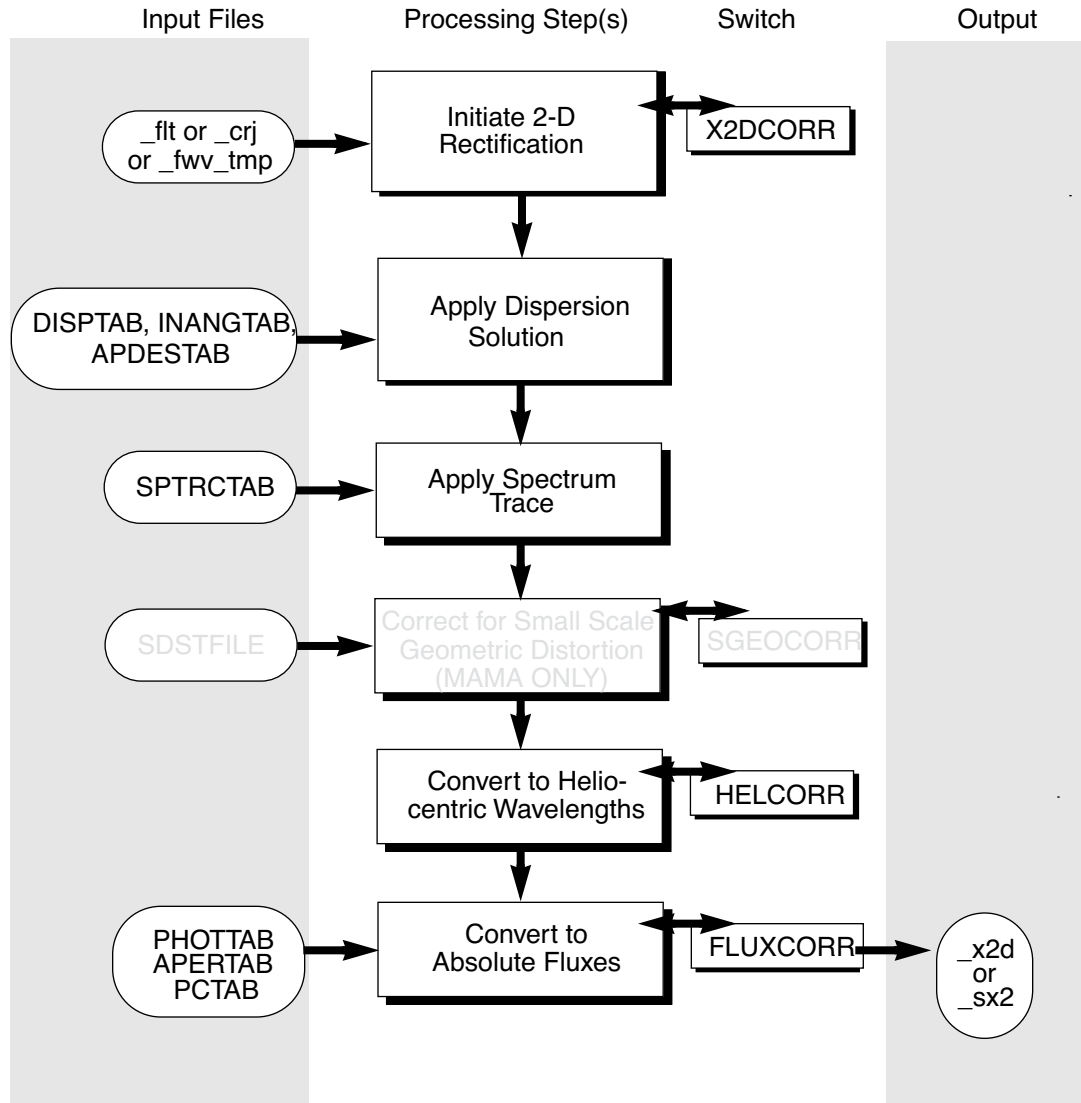
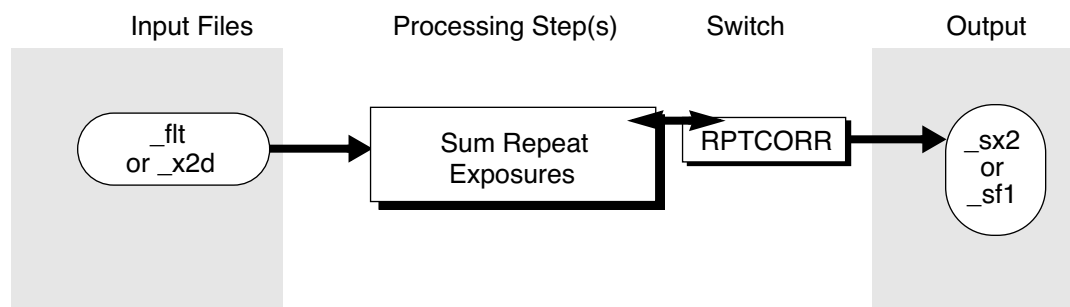


Figure 3.7: Flow through calstis-8, Summing repeatobs Exposures



3.4 Descriptions of Calibration Steps

In this section we provide a more detailed description of the algorithms applied by **calstis**. As always, a given step will be performed on your data if the corresponding calibration switch in the input data was set to **PERFORM** (see chapter 2). The algorithmic descriptions below are described according to the major component of the **calstis** pipeline in which they are used, namely:

- Two-dimensional image reduction, including basic 2-D reduction, cosmic-ray rejection and image co-addition.
- Processing of the contemporaneously obtained wavecal.
- Two-dimensional and one-dimensional spectral extraction, with flux and wavelength calibration.

Within each component, the individual steps are listed alphabetically, because the order in which they are performed can change for different types of data (e.g., CCD or MAMA, spectroscopic or imaging, **CR-SPLIT** or not).

More detailed descriptions can be found in a series of *Instrument Science Reports* (ISRs) that discuss the pipeline. Be aware, however, that while these reports describe the original design of the pipeline and the associated algorithms in detail, they do not always contain information concerning later modifications. Over time these reports will be updated to include a more complete description of the pipeline. In the meantime we refer the reader to the STIS WWW page, where a history of the important changes to the **calstis** pipeline code is maintained:

<http://www.stsci.edu/instruments/stis/>

3.4.1 ATODCORR: analog to digital conversion correction

- Reference file: ATODTAB

This step is part of 2-D image reduction and applies only to CCD data. *Ground test results show that this correction is not currently needed, so the ATODCORR switch will always be set to OMIT.* An analog to digital correction would be applied if the CCD electronic circuitry that performs the analog to digital conversion were biased toward the assignment of certain DN (data number) values.

3.4.2BACKCORR: 1D spectral background subtraction

- Reference file: XTRACTAB

This step is a part of spectral extraction and applies to one-dimensional extraction only. If the calibration switch BACKCORR is PERFORM, the background is calculated and subtracted from the extracted spectrum. The background is extracted from regions above and below the spectrum on the detector, and a function is fit to the variation of the background along the spatial axis (AXIS2). The fitting function is restricted to a zeroth or first order polynomial. The polynomial order, BACKORD, is read from the XTRACTAB reference table. The background extraction is done one column at a time, with the average background values (in cts/sec/pixel) calculated (with 3σ clipping) from each background bin, accounting for fractional pixel contributions. However, this is likely to lead to too much noise for a good extraction (improved algorithms are under investigation). In the case of BACKORD=0, a simple average of the two background bins is computed. For BACKORD=1, the background value at the center of each pixel that contributes to the extracted spectrum is derived from the linear fit to the background. The background in the spectrum extraction box is integrated and subtracted from the sum of the spectrum box. The total background at each pixel in the output spectrum is written to the output data table.

In general, the background or sky is not aligned with the detector pixels. To accommodate this misalignment, the definition of the background extraction apertures includes not only a length and offset (center-to-center) but also a linear tilt to assist in properly subtracting the background. This tilt is taken into account when calculating the average background in the background extraction boxes.

3.4.3BIASCORR: small-scale bias subtraction

- Reference file: BIASFILE

This step is part of basic 2-D image reduction and is performed only for CCD data, removing any two-dimensional additive stationary pattern in the electronic zeropoint of each CCD readout. To remove this pattern, a bias reference image is subtracted. The bias reference file is a full-format *superbias* image created from many bias frames to assure low noise. Separate bias files are used for different values of binning and for different values of gain. The bias image has an associated data quality image extension; bad pixels in the bias image are flagged in the science data quality image.

3.4.4 BLEVCORR: large-scale bias & overscan subtraction

- Reference file: none

This step is part of basic 2-D image reduction and is performed only for CCD data. This step subtracts the electronic bias level for each line of the CCD image and trims the overscan regions off of the input image, leaving only the exposed portions of the image. Thus, the BLEVCORR corrects the large-scale bias (a plane), while BIASCORR corrects the small-scale (pixel-by-pixel) bias.

Because the electronic bias level can vary with time and temperature, its value is determined from the overscan region in the particular exposure being processed. A raw STIS CCD taken in full frame unbinned mode will have 20 rows of virtual parallel overscan in the AXIS2 (image y) direction, which is created by over-clocking the readout of each line past its physical extent, and 19 leading and trailing columns of serial physical overscan in the AXIS1 (image x) direction, which arise from unilluminated pixels on the CCD. Thus the size of the uncalibrated and unbinned full frame CCD image is 1062 (serial) by 1044 (parallel) pixels, with 1024 x 1024 exposed science pixels.

Both the serial (physical) and virtual (parallel) overscans are used for the overscan bias level determination. A line-by-line subtraction is performed in the following way. An initial value of the electronic bias level, or overscan, is determined for each line of the image, using the serial and parallel overscans, and a function, currently a straight line, is fit to these values as a function of image line. This determines the slope of the BLEV across the image at each line. The actual overscan value subtracted from an image line is the value of the linear fit at that image line. The initial value for each line is found by taking the median of a predetermined subset of the trailing serial overscan pixels. Currently, that region includes most of the trailing overscan region, however the first pixel and last three pixels are skipped, as they have been shown to be subject to problems; pixels flagged as bad in the input data quality flag are also skipped. The region used changes as a function of the binning or subarray (see table 3.1). The mean value of all overscan levels is computed and the mean is written to the output SCI extension header as MEANBLEV.

In addition to subtracting the electronic bias level, the BLEVCORR step also trims the image of overscan. The sizes of the overscan regions depend on binning and whether the image is full-frame or a subimage. The locations of the overscan regions depend on which amplifier was used for readout. The number of pixels to trim off each side of the image (before accounting for readout amplifier) is given in table 3.2. The values of NAXIS1, NAXIS2, BINAXIS1, and BINAXIS2 are obtained from image header keywords. Because the binning factor does not divide evenly into 19 and 1062, when on-chip pixel binning is used the raw image produced will contain both pure overscan pixels, overscan plus science pixels, and

science pixels. The **calstis** pipeline will only calibrate pixel binnings of 1, 2, and 4 in either AXIS1 or AXIS2.

The keywords **CRPIX_i** (giving the pixel coordinates of the reference pixel) and **LTV_i** (giving the pixel coordinates of the subsection start) are updated in the output. These depend on the offset due to removal of the overscan.

Table 3.1: Raw Image Pixels Used to Determine Line by Line Bias Level

Columns in Raw Image	Binning
2 through 16	Unbinned
2 through 8	All other supported binnings

Table 3.2: Pixels Trimmed During CCD Bias Level Correction for Amp D

Side	Full Image	Subarray Images	Binned Images
Right	19	18	$(19 + 1) / \text{BINAXIS1}$
Left	19	18	$\text{NAXIS1} - (1024 / \text{BINAXIS1} - 1) - \text{Right}$
Top	0	0	0
Bottom	20	0	$\text{NAXIS2} - 1024 / \text{BINAXIS2}$

3.4.5CRCORR: cosmic-ray correction

- Reference files: CRREJTAB & CCDTAB

The CRCORR step is applicable only to CCD data; the correction is applied partway through the basic 2D reduction process, but is performed by the **ocrreject** task, not **basic2d** itself. For STIS, the recommended procedure for observations longer than several minutes is to obtain two or more identical exposures so that an anti-coincidence technique can be used in the data calibration pipeline. For this reason, the CCD exposures are split to multiple associated exposures, specified by the NUMBER-OF-ITERATIONS or CR-SPLIT parameters, with roughly the same exposure times. The CRCORR step sums the individual CR-SPLIT exposures in an associated dataset, producing a single cosmic ray rejected file (**_crj.fits**). The **ocrreject** task in the **calstis** pipeline is similar to the WFPC2 **crrej** task, except that when forming the output image, **ocrreject** uses the input data quality flags to discard data. This is done by reading the controlling input parameter (**crmask**) from the CRREJTAB reference file, which indicates whether or not to flag an input file.

The CRCORR contains the following steps:

- Forms a stack of images to be combined (the CR-SPLIT or REPEATOBS exposures in the input file).
- Forms an initial guess image (minimum or median).
- Forms a summed CR-rejected image, using the guess image to reject high and low values in the stack, based on sigma and the radius parameter that signifies whether to reject pixels neighboring cosmic-ray impacts.
- Iterates, using different (usually decreasing) rejection thresholds to produce a new guess image at each iteration.
- Produces a final cosmic ray rejected image (`_crj`), including science, data quality, and error extensions, which is the sum of the input images. Then updates various header keywords.
- Flags the data quality arrays of the individual (non-CR-rejected) input files to indicate where an outlier has been found (pixels that were rejected because of cosmic ray hits can be identified by looking for data quality bit = 14 in the `_flt.fits` file).

Note that the `_crj` image is the sum of the input CRSPLIT images. For each pixel in the `_crj` image, the sum has been scaled to reflect any cosmic-ray rejections in the individual images contributing to that sum; thus, the counts in every `_crj` pixel correctly reflect the total exposure time (TEXPTIME) of the CRSPLIT images. E.g., if CRSPLIT=3, and one of the 3 input pixels contributing to a given pixel in the `_crj` file is rejected during CRCORR, that pixel in the `_crj` file will be the sum of the 2 good input pixels, scaled by 3/2.

If `sky=mode`, a sky value is calculated as the mode in all pixels for each input image and subtracted from each pixel value prior to placement in the stack. If `sky=none`, no background is subtracted.

The method for forming the initial guess image is controlled through the parameter `initguess`, which takes the median or minimum pixel value to indicate which of these in the stack, at each pixel, is to be used. Only input pixels whose data quality flags are not set to the `badinpdq` parameter are used to form the guess. The bad pixel masking takes place prior to forming the initial guess image, which prevents bad data in a single input image from contaminating the output product.

The cosmic-ray-rejected image is created by setting the value at each pixel to the sum of the values of all good pixels in the stack whose values are within plus or minus `crsigmas*noise` of the initial guess image. Deviant (out of range) stack pixels are flagged as cosmic ray impacted by setting their stack data quality flags to $2^{13}=8192$ in the input file.

The value of *noise* (in DN) is computed as:

$$noise = \sqrt{\left(\frac{READNSE}{ATODGAIN}\right)^2 + \left(\frac{DN}{ATODGAIN}\right)^2 + (SCALENSE \times 0.01 \times DN)^2}$$

Where:

- DN = the data number of the stack pixel value in counts.
- READNSE is the read noise in electrons, read from the primary header READNSE.
- ATODGAIN is the calibrated conversion from electrons to DN, read from the primary header ATODGAIN.
- `scalense` is an input parameter, read from the CRREJTAB calibration reference file.

The `crsigmas` parameter is a string, e.g., `crsigmas = "4,3"`, read from the CRREJTAB calibration reference file. The number of entries in the string dictates the number of iterations to be performed (in this example two) and the values in the string indicate the value of `crsigmas` for each iteration. In this example, stack values that deviate from the guess image value by more than $4 \times noise$ in the first iteration are considered to be outliers and are excluded from the average on the first iteration when an improved guess image is formed. A second iteration is then performed in which `crsigmas` is set to 3 and good stack values dispartate by more than $\pm 3 \times noise$ from the guess image are excluded when determining the average. In each iteration, if `crradius` is not INDEF, then pixels neighboring rejected pixels will be subjected to a more stringent rejection test. Most CR impacts affect more than one pixel. For this reason, the mechanism is designed to permit neighboring pixels to be examined as separate cases with more stringent rejection thresholds to exclude cosmic rays (see *ISR STIS 98-11*).

The `scalense` parameter is a string containing a multiplicative factor in the noise relation, linearly proportional to the signal level. This allows for the treatment of extra noise due to other factors, such as image motion (e.g., a bright star in one pixel moving to the next could accidentally get flagged as a CR). If `scalense = "2.0"`, then the term $0.02 \times value$ is added in quadrature to the *noise*. This term accounts for multiplicative effects that would be expected if this rejection were applied to flat-fielded data. The `scalense` parameter allows well-exposed regions to be used (such as the centers of stars, where jitter from the telescope may slightly change the pointing from image to image), rather than incorrectly rejecting such regions as cosmic rays.

The combination of the individual CRSPLIT or REPEATOB exposures into a single cosmic-ray rejected frame is performed early in the

calstis flow. The cosmic ray rejection is performed after each exposure has had its data quality file initialized (DQICORR), the overscan bias level subtracted (BLEVCORR), and the bias frame subtracted (BIASCORR), but prior to subtraction of a dark frame (DARKCORR) and flatfielding (FLATCORR). The CR-rejected image is then passed through the remainder of the two-dimensional image reduction to produce a flatfielded CR-rejected image (`*_crj.fits`). This CR-rejected flatfielded image is then passed through the subsequent processing steps in **calstis**. If EXPSCORR is set to PERFORM (see below), then the individual flatfielded but not cosmic ray rejected exposures are also produced.

The strategy for optimizing the CR rejection parameters for the STIS pipeline is explained in detail in *ISR STIS 98-22*.

3.4.6DARKCORR: dark signal subtraction

- Reference files: DARKFILE & CCDTAB

The DARKCORR step is part of basic 2-D image reduction and removes the dark signal (count rate created in the detector in the absence of photons from the sky) from the uncalibrated science image. If the science image is a subarray or was binned, the relevant section of the dark reference image must be extracted and binned to match the science image. If Doppler correction was applied on-board for the science data (i.e., if DOPPON = T, or, in the case of TIMETAG data, if DOPPON = F and DOPPMAG > 0), the Doppler smearing function is computed and convolved with the dark image (if DOPPCORR=PERFORM) to account for the contributions of various detector pixels to a particular image pixel. This is done before binning the dark image and applies only to MAMA data taken with the first order medium resolution gratings or in the echelle gratings. The science data quality file is updated for bad pixels in the dark reference file.

The mean of the dark values subtracted is written to the SCI extension header with the keyword MEANDARK. For CCD data, the dark image is multiplied by the exposure time and divided by the `atodgain` (from the CCD parameters table) before subtracting. We are currently implementing a temperature-dependent dark subtraction for CCD data taken during Side-2 operations (see section 1.3, and also *ISR STIS 2001-03*). This subtraction will use the CCD housing temperature to estimate the dark current because of the absence on Side 2 of a temperature sensor for the detector. For MAMA data, the dark image will be multiplied by the exposure time before subtracting. For the NUV MAMA, the dark image is also scaled by a temperature-dependent factor prior to subtraction (see Chapter 7 of the *STIS Instrument Handbook*), although this algorithm is currently being improved; for the FUV MAMA, a dark “glow” in the upper left-hand quadrant of the image is also correlated with temperature, but it is

not currently corrected by the dark subtraction. For either MAMA, the dark will also be convolved with the Doppler smoothing function if DOPPCORR is PERFORM.

Hot pixels, caused by radiation damage, also occur in the STIS CCD. Annealing of hot pixels is performed by raising the normal operating temperature of the CCD (-83 C) to its ambient value of 5 C. Analysis of the on-orbit data has shown that this process is successful in removing 80% of the transient pixels hotter than 0.1 e⁻/sec/pix. While post-pipeline calibration using appropriate STIS reference superdarks allows one to subtract most hot pixels correctly, the best way to eliminate all hot pixels is by dithering (making pixel scale positional offsets between individual exposures).

3.4.7DISPCORR: apply dispersion solution

- Reference files: DISPTAB, INANGTAB, & APDESTAB

This step is a part of spectral extraction. Wavelengths are assigned using dispersion coefficients from the reference table DISPTAB when the calibration switch DISPCORR is PERFORM; if DISPCORR is OMIT, no wavelengths are assigned. The DISPTAB table contains dispersion solutions for a defined reference aperture. Offsets introduced by using apertures other than a reference aperture are removed using coefficients in the INANGTAB reference table. Also, offsets due to MSM shift determined from wavecal exposures are corrected by applying a linear offset after all other corrections have been made. DISPCORR is always done for _x2d files, independent of the CAL switch setting in the header. More details on the DISPCORR task are given in *ISR STIS 98-13*.

For MAMA data, offsets of the projection of the spectrum onto the detector in both the spectral and spatial directions are deliberately introduced by offsetting the Mode Select Mechanism (grating wheel) tilts. This is done approximately monthly to assure a more uniform charge extraction from the microchannel plate over time. For MAMA observations, these induced offsets are measured during wavecal processing, and they are subtracted along with the random MSM offset.

The DISPTAB table of dispersion contains coefficients for fits to the following dispersion solution:

$$s = A_0 + A_1 m \lambda + A_2 (m \lambda)^2 + A_3 m + A_4 \lambda + A_5 m^2 \lambda + A_6 m \lambda^2 + A_7 (m \lambda)^7$$

where

- λ is the wavelength in Angstroms.
- s is the detector AXIS1 position.
- m is the spectral order.

- A_i are the dispersion coefficients.

The prism dispersion relation is different, giving the wavelength as a function of pixel position in the following manner:

$$\lambda = A_1 + \frac{A_2}{s - A_0} + \frac{A_3}{(s - A_0)^2} + \frac{A_4}{(s - A_0)^3} + \frac{A_5}{(s - A_0)^4}$$

For each pixel in the AXIS1 direction, a wavelength is calculated. First, any modification to the dispersion coefficients due to spectrum offsets must be made (see table 3.1). For each integer value of s in the AXIS1 direction, a wavelength is solved for iteratively using the Newton-Raphson method.

Table 3.1: Modifications to the Dispersion Coefficients Caused by Offsets

Correction	Ref Table	Algorithm	Definitions
Incidence Angle	INANGTAB	$A_i = A_i + \sum c_1(i) \times s$ $A_0 = A_0 + c_2(1) \times s + c_2(2) \times s^2$	A_i dispersion coefficients c_i incidence angle coefficients S aperture offset in the axis 1 direction calculated as difference of aperture centers (arcsec)

3.4.8 DOPPCORR: correct reference files for Doppler shift

- Reference files: none

This step is part of basic 2-D image reduction and is performed only for spectroscopic data taken with the MAMA detectors (the correction is not significant for CCD spectroscopy, which is at lower resolution). When MAMA data are taken in ACCUM mode in the first order medium (M) gratings or the echelle modes, the MAMA flight software corrects the location of each photon for the Doppler shift induced by the spacecraft motion, prior to updating the counter in the ACCUM mode image being produced. In this case, the flatfielded and dark reference files should be convolved with the Doppler smearing function, because the counts on a single image pixel were actually detected at different (Doppler-shifted) detector pixel locations. Therefore, during basic two-dimensional image reduction of the MAMA data, the darks and flats must be processed with the same Doppler smoothing as the science data prior to application of the reference image. Because this is not an independent routine, but a modifier for the steps DQICORR, DARKCORR and FLATCORR, the DOPPCORR flag is never set to COMPLETE.

The first step is to compute an array containing the Doppler smearing function. The expression below gives the computed Doppler shift, where the time t begins with the value of the header keyword EXPSTART and is incremented in one-second intervals up to EXPSTART + EXPTIME

inclusive. At each of these times, the Doppler shift in unbinned pixels is computed as:

$$shift = -DOPPMAG \times \sin(2\pi(t - DOPPZERO)/ORBITPER)$$

where DOPPMAG is the Doppler shift amplitude in high-res pixels, DOPPZERO is the time when the Doppler shift was zero and was increasing (i.e., near when the HST was closest to the target) and ORBITPER is the orbital period of the HST in seconds. DOPPMAG, DOPPZERO and ORBITPER are all SCI extension header keywords. If a photon hits detector location (X,Y), then pixel (X+shift,Y) in the image would have been incremented. Therefore, a positive shift means that the reference files (dark and flat) should be shifted to the right before being applied. The value of *shift*, estimated from the above equation, is rounded to the nearest integer.

In TIMETAG mode with the medium or high resolution echelles, this correction is not applied on-board, but is done by Generic Conversion when making an ACCUM image from the TIMETAG table. The Doppler-corrected coordinates are then written in a new column in the TIMETAG table (`_tag.fits`). When the TIMETAG data are integrated in time (in Generic Conversion or by using `inttag`) to produce an uncalibrated accumulated science image (`_raw.fits`), it is the Doppler-corrected positions that are used.

3.4.9DQICORR: Initialize data quality file

- Reference file: BPIXTAB

The DQICORR step is part of basic 2-D image reduction. This routine takes the initial data quality file output for the science data and performs a bit-wise OR with the values in the bad pixel reference file table (BPIXTAB) to initialize the science data quality file for propagation through subsequent steps in `calstis`. If DOPPCORR=PEFORM, `calstis` will combine data quality information from neighboring pixels to accommodate Doppler smearing prior to performing the OR operation with the (unsmeared) science input data quality image. The DQICORR step also appropriately combines data quality flags in neighboring pixels if the images are binned.

For CCD data, this step also includes a check on saturation, comparing the science data values with the saturation level read from the CCD parameters table (CCDTAB). Also, it flags the regions of the CCD beyond the edge of the aperture to prevent problems with sky level computation and cosmic ray rejection. When a large fraction of the image is not illuminated, the computation of the sky level can be seriously affected. The aperture size is read from the APER_FOV keyword.

3.4.10EXPSCORR: retain individual CR-SPLIT exposures

- Reference file: none

The EXPSCORR step is a part of basic 2-D image reduction, for CCD data only. If the EXPSCORR calibration switch in the header is set to PERFORM, the pipeline will also process the SCI extensions in the `_raw.fits` files as individual exposures through **calstis**, outputting an intermediate product, `_flt.fits`. This file contains the individual flatfielded CRSPLIT exposures in successive imsets of a single file. This file will not be passed through the subsequent calibration steps (e.g., spectroscopic reduction), but will be retained as an intermediate data product, to allow users to see exactly which pixels were excluded from the summed cosmic-ray rejected image.

3.4.11FLATCORR: flat-field correction

- Reference files:PFLTFILE, DFLTFILE, & LFLTFILE

The FLATCORR step is part of basic 2-D image reduction and corrects for pixel-to-pixel and large-scale sensitivity gradients across the detector by dividing the data by a flatfield image. The flatfield image used to correct the data is created from three flatfield reference files:

- **PFLTFILE** - This flat is a configuration (grating, central wavelength and detector) dependent pixel-to-pixel flatfield image, from which any large-scale sensitivity variations have been removed (i.e., it will have a local mean value of unity across its entirety). Such configuration dependent flats are expected to be produced infrequently, perhaps once per year.
- **DFLTFILE** - This file is a *delta flat* that gives the changes in the small scale flatfield response relative to the pixel to pixel flat (**PFLT-FILE**). *Delta flats are not currently used.*
- **LFLTFILE** - This flat is a subsampled image containing the large-scale sensitivity variation across the detector. It is usually grating- and central wavelength-dependent (for spectroscopic data) and aperture (filter) dependent for imaging data.

To flatfield science data, **calstis** creates a single flatfield image from these three files¹ as described below and then divides the science image by the flat so created. The pixels of the science data quality file are updated to reflect bad pixels in the input reference files and the errors in the science data are updated to reflect the application of the flat. Blank values of **PFLTFILE**, **DFLTFILE**, or **LFLTFILE** in the science data indicate that type of flat is not to be used.

To create the single combined flatfield file, **calstis** first expands the large-scale sensitivity flat (**LFLTFILE**) to full format, using bilinear interpolation. The pixel-to-pixel flat, delta flat, and expanded low-order flat are then multiplied together. For MAMA data, the product of the flatfield images will be convolved with the Doppler smoothing function if **DOPPCORR = PERFORM**. If a subarray or binning was used, after taking the product of all the flatfields that were specified, a subset is taken and binned if necessary to match the uncalibrated image, and the uncalibrated data are then divided by the binned subset.

The error and data quality arrays in the calibrated file will be updated to reflect the error and data quality from the flatfield (i.e., the errors are rescaled by the flatfield, and the errors associated with the flatfield are added in quadrature). If Doppler convolution is applied, a correction is also applied for the loss of counts at the image edges due to flight software's "effective sub-array" (see *ISR STIS 98-05* for details).

3.4.12FLUXCORR: conversion to cgs flux

- Reference files: **APERTAB**, **PHOTTAB**, & **PCTAB**

This step is part of spectral extraction. If **FLUXCORR** is **PERFORM**, the raw counts are corrected to F_λ (erg cm⁻² sec⁻¹ Å⁻¹). Execution of this calibration step requires that wavelengths have been assigned previously by **DISPCORR**. Corrections for vignetting and echelle blaze are handled within the **PHOTTAB** reference files. The conversion to absolute flux for a point source is calculated as:

$$F_\lambda = \frac{hc \cdot g \cdot H}{A_{HST} R_\lambda T_\lambda \lambda \Delta \lambda} C_\lambda$$

where:

- F_λ is the calibrated flux at a particular wavelength. This quantity is also multiplied by the **ATODGAIN** if the data were obtained with the CCD.
- h is Planck's constant.

1. The rationale for maintaining three types of flatfield reference files rather than a single integrated reference file is described in detail in *ISR STIS 95-09* "Calibration Plans for Flat Fielding STIS Data."

- c is the speed of light.
- g is the detector gain, which is unity for MAMA observations. For the CCD, this is the conversion from counts to electrons, the value for which is given in the header keyword ATODGAIN.
- H is a correction factor accounting for the finite extraction box height (EXTRSIZE) used to extract the spectra; it is the ratio of throughput for an infinite extraction box height divided by the throughput for the extraction box height used to extract the spectrum, where the throughputs are taken from the PCTAB (*_pct.fits) reference table.
- A_{HST} is the area of the unobstructed HST primary mirror (45238.93416 cm²).
- R_{λ} is the throughput of the STIS instrument configuration read from the PHOTTAB.
- λ is the wavelength in Angstroms.
- $\Delta\lambda$ is the dispersion (Å/pixel) at a particular wavelength.
- C_{λ} is the net count rate (cts/sec) at a particular wavelength.
- T_{λ} is the aperture throughput at a particular wavelength.



The flux correction is applied somewhat differently for 1-D and 2-D extractions. See section 5.4 for details.

The flux calibration applied for one-dimensional extraction (X1DCORR) applies to a point source only; there is no simple conversion at present for diffuse sources. The data in the two-dimensional rectified, flux-calibrated images are appropriate for diffuse, continuum sources. The keyword DIFF2PT is written to the _x2d (or _sx2) image header to allow the extraction of point source flux from the 2-D rectified data. DIFF2PT is *not* found in the output written during one-dimensional extraction. See section 5.4 for details.

3.4.13GEOCORR: geometric distortion correction

- Reference files: IDCTAB

Geometric correction is part of secondary 2-D image reduction and is applicable to all ACCUM mode imaging data. The method used is similar to 2-D rectification of spectroscopic data (see section 3.4.24). For each pixel in the output rectified image, the corresponding point is found in the

input distorted image. Bi-linear interpolation in the input image is used to get the data value to assign to the output. The output pixel value is the weighted sum of the values of the four nearest pixels in the input image. The weights depend on distances of that particular pixel from each of the four pixels, with the sum of the weights normalized to unity (see [ISR STIS 98-13](#)). Note that such distances are determined from the centers of the pixels, as per the IRAF convention. One rectified image is produced for each spectral order. Mapping from an output pixel back into the input images is specified by a two-dimensional power-series polynomial and is stored in the format generated by the IRAF **gsurfit** package. The distortion coefficients are fully described in [ISR ACS 2000-11](#). The errors are interpolated using the same weights as for the science data, except that the errors are combined in quadrature.

3.4.14 GLINCORR and LFLGCORR: nonlinearity corrections

- Reference file: MLINTAB

These steps are part of basic 2-D image reduction and are performed only for the MAMA detectors. The MAMAs are photon counting detectors. At high photon (pulse) rates, the MAMA response becomes nonlinear due to four effects:

- Pore paralysis in the micro channel plates arises when charge cannot flow rapidly enough to replenish channels whose electrons have been depleted due to high local photon rates. This depletion produces a *local* non-linearity. The local count rate is roughly linear up to ~200 cts/sec/pixel and then turns directly over, showing an inverted V shape. Thus, it is not possible to reliably correct for or flag pixels that have exceeded the local linearity limit in the pipeline (because the relation is bi-valued).
- The electronic processing circuitry has a dead-time of roughly 350 nano-seconds between pulses; thus at global count rates (across the detector) of 300,000 counts (pulses) per second, the electronic circuitry counts roughly 90% of the pulses.
- The MIE electronics and flight software can process at most 300,000 pulses per second (i.e., it is matched to the expected global count rate performance of the electronic circuitry). At count rates higher than this, the MIE will still count only 300,000 pulses per second—this represents a hard cutoff beyond which no information is available to allow correction to the true count rate. In practice, at count rates approaching 270,000 counts/sec the flight software begins losing counts due to the structure of its data buffers.

- For subarrays, the hard cutoff limit of the MIE electronics and software will differ from that for full frame processing, but will still be dependent on the total global rate in addition to the rate within the subarray. The **calstis** pipeline currently applies the full frame correction to subarray data.

The local and global count rates that lead to nonlinearity are both above the screening limits for MAMA observations, so under normal circumstances, these nonlinearities are unimportant. However, if a brighter-than-allowed target is inadvertently observed without shuttering the detector, these linearity corrections are important.

The global count rate (across the entire detector) is determined as part of the bright object protection sequence and is passed down with the exposure as a header keyword, **GLOBRATE**, in the science header. **calstis** also computes **GLOBRATE** and updates the value in the header. If either **GLINCORR** or **LFLGCORR** is **PERFORM**, the global count rate will be checked; a correction for global non-linearity is applied if **GLINCORR** is **PERFORM**, using the parameters **GLOBAL_LIMIT**, **LOCAL_LIMIT**, **TAU**, and **EXPAND** read from the **MLINTAB** reference table.

If the value of the SCI extension header keyword **GLOBRATE** is greater than **GLOBAL_LIMIT**, the keyword **GLOBLIM** in the SCI extension header will be set to **EXCEEDED**, with no change made to the data. Otherwise, **GLOBLIM** will be set to **NOT-EXCEEDED**, and a correction factor will be computed and multiplied by each pixel in the science image and error array. The correction factor is computed by iteratively solving $GLOBRATE = X * \exp(-TAU * X)$ for X , where X is the true count rate. This algorithm has not yet been updated to account for the linearity effects from the flight software data buffer management.

If **LFLGCORR** is **PERFORM**, each pixel in the science image is also compared with the product of **LOCAL_LIMIT** and the exposure time **EXPTIME**. That count rate limit is then adjusted for binning by dividing by the pixel area in high-res pixels. If the science data value is larger than that product, that pixel and others within a radius of **EXPAND** high-res pixels are flagged as nonlinear. Because our understanding of the MAMA processing electronics is currently incomplete, accurate fluxes (global linearity) at count rates exceeding 270,000 cts/sec cannot be expected from the **calstis** pipeline.

3.4.15HELCCORR: correction to heliocentric reference frame

- Reference file: none

This step is part of spectral extraction. The correction of wavelengths to a heliocentric reference frame is controlled by calibration switches **HELCCORR** and **DISPCORR** — if both switches are set to **PERFORM** then the correction is made. The functional form of the correction (shown

below) requires the calculation of the heliocentric velocity (v) of the Earth in the line of sight to the target.

$$\lambda_{helio} = \lambda_{obs} \left(1 - \frac{v}{c} \right)$$

where:

- λ_{helio} is the heliocentric wavelength.
- λ_{obs} is the wavelength observed, prior to heliocentric correction.
- v is the component of the velocity of the Earth away from the target.
- c is the speed of light.

The derivatives of low-precision formulae for the Sun's coordinates described in the *Astronomical Almanac* are used to calculate the velocity vector of the Earth in the equatorial coordinate system of the epoch J2000. The algorithm does not include Earth-Moon motion, Sun-barycenter motion, nor light-time correction from the Earth to the Sun. This value for the Earth's velocity should be accurate to ~ 0.025 km/sec during the lifetime of STIS. (Note that the uncertainty of 0.025 km/s is much less than the ~ 2.6 km/s resolution obtained with the STIS high dispersion echelle gratings.) The value of heliocentric velocity, v , is written to the HISTORY record in the primary header of the output spectrum file.

3.4.16 LORSCORR: conversion to native format MAMA pixels

- Reference file: none

This step is part of basic 2-D image reduction and is performed for MAMA data only. MAMA data are, by default, taken in high resolution mode (2048 x 2048 pixels), in which the individual microchannel plate pixels are subsampled by the anode wires. This mode produces an image with improved sampling but with appreciably worse flatfielding properties (see Chapter 11 of the *STIS Instrument Handbook* for more details). If LORSCORR is set to PERFORM, **calstis** simply adds the counts in pairs of adjacent pixels to produce images in the native format (or so-called *reference format*) of the MAMA detectors, with 1024 x 1024 pixels.

The binning of the uncalibrated image is determined from the LTM1_1 and LTM2_2 keywords in the SCI extension header of the raw data file. LTMi_i = 1 implies the reference pixel size, and LTMi_i = 2 means the pixels are subsampled into high-res format. In this step, if either or both axes are high-res, they will be binned down to low-res. The binning differs from binning reference files to match an uncalibrated image, in that the pixel values in this step are summed rather than averaged.

3.4.17PHOTCORR: populate photometry header keywords

- Reference files:PHOTTAB & APERTAB

This step is part of basic 2-D image reduction and is applicable only for OBSTYPE=IMAGING data. For image mode, the total system throughput is calculated from the reference files. A **synphot** routine is then called to compute the inverse sensitivity, reference magnitude, pivot wavelength, and RMS bandwidth (see the help for **bandpar** for more details); these four quantities are written to the primary header keywords PHOTFLAM, PHOTZPT, PHOTPLAM, and PHOTBW, respectively.

3.4.18RPTCORR: sum repeated MAMA observations

- Reference file: none

This step is part of secondary 2-D image reduction and is applicable only for MAMA data. If the number of repeat exposures is greater than one, then **calstis** will sum the flatfielded data in the case of image mode data (producing a `_sfl.fits` file) or the two-dimensionally extracted data (producing a `_sx2.fits` file) in the case of long-slit data. RPTCORR just applies a straight pixel-to-pixel addition of the science values, bit-wise ORs the data quality files, and determines the error as the square root of the sum of the squares of the errors in the individual exposures.

3.4.19SC2DCORR: echelle scattered light correction

- Reference files: CDSTAB, ECHSCTAB, EXSTAB, RIPTAB, HAL-OTAB, TELTAB, SRWTAB

This step is part of spectral extraction. If SC2DCORR is set to PERFORM, **calstis** iteratively constructs a two-dimensional model of scattered light present in the raw image. After 3 iterations, the scattered light model is subtracted from the raw data and spectral extraction proceeds using the **x1d** “unweighted” algorithm described in section 3.4.23. The “sc2d” algorithm has been used since 2000-Dec-21 to process all HST archive requests for STIS echelle data, regardless of when the observations were performed, unless use of an unsupported mode prevents spectral extraction. Two-dimensional scattered light subtraction may be suppressed either by setting SC2DCORR to OMIT in the `_raw` file and reprocessing the entire association with **calstis**, or more directly by setting the `algorithm` parameter in **x1d** to `unweighted` and reprocessing the `_flt` file with the **x1d** task.

Construction of the two-dimensional scattered light model involves several steps. In general terms, counts in an approximate one-dimensional

extracted spectrum are mapped back onto a hypothetical two-dimensional scene extending beyond the edge of the physical detector. Echelle scatter is simulated by redistributing counts along diagonal lines of constant wavelength, using echelle line spread functions. Simulated images are created with and without scattered light by using either the full echelle line spread function or only that of the core. Post-echelle smearing along columns is simulated by independently convolving columns with a smoothing kernel. Scattering due to the aperture truncated telescope PSF, isotropic detector halo, and scattering by the cross-disperser are treated by two-dimensional convolution. Taking the difference between simulated images with and without scattered light yields a two-dimensional scattered light model. Iteration is used to achieve self-consistency between the scattered light model used to extract a one-dimensional spectrum and the model derived from that extracted spectrum.

3.4.20SGEOCORR: small-scale distortion correction

- Reference file: SDSTFILE

This step would be part of spectral extraction and apply only to MAMA data, *but it is not presently implemented*. If SGEOCORR were PERFORM, a correction would be applied for the small scale geometric distortions in the MAMA detectors. The corresponding reference file, SDSTFILE, would contain the distortion offsets for each pixel in the MAMA image.

3.4.21SHADCORR: CCD shading correction

- Reference file: SHADFILE

This step is part of basic 2-D image reduction and applies only to CCD data, *but it is not currently performed*. It is designed to correct for shading by the CCD shutter in very short integration time exposures. The STIS CCD shutter is specified to produce exposure non-uniformity less than or equal to 5 milliseconds for any integration time: the shortest possible STIS CCD exposure time is 100 milliseconds. Ground testing has shown that this step is not currently required.

3.4.22WAVECORR: wavecal correction

- Reference files: LAMPTAB, APDESTAB, WCPTAB, DISPTAB, INANGTAB, SPTRCTAB, SDCTAB

This step is a part of wavecal processing and applies only to spectroscopic data. The purpose of wavecal processing is to determine the shift of the image on the detector along each axis owing to uncertainties in

positioning by the Mode Select Mechanism (MSM) and to thermal motions. It requires one or more contemporaneous wavecal (line lamp) observations, taken without moving the MSM from the setting used for the science data.

Basic two-dimensional image reduction is first applied to the wavecal. For CCD data taken with the HITM system prior to 1998-Nov-9, the external shutter was left open during wavecal exposures, so the detector was exposed to radiation from both the science target and the line lamp. In this case, the next step is to subtract the flatfielded science image from the wavecal, after scaling by the ratio of exposure times and by the ratio of gains.

For first-order data, two-dimensional rectification is then applied to the flatfielded (and possibly science subtracted) wavecal. The result is a temporary file that will normally be deleted after processing. For prism or echelle data, the 2-D rectification step is not done.

Because wavecal data are not CR-SPLIT, cosmic rays must be identified and eliminated by looking for outliers within columns, i.e., in the cross-dispersion direction. Because the data have been rectified, values within a single column are all at the same wavelength and hence, are fairly consistent in their brightness for a line lamp exposure. Cosmic rays will stand out above this brightness level. The image can then be collapsed along columns to get a long-slit integrated spectrum or along rows to get an outline of the slit (in the cross-dispersion direction).

In removing cosmic rays from a wavecal, a different cosmic ray rejection function has been added to *calstis4*. This new function is called if the SCI extension SDQFLAGS (serious data quality flags) includes 8192. The logic here is that if SDQFLAGS does not include 8192 (DATA REJECT), the bit used to flag cosmic rays, then the cosmic rays are not considered “serious”. Therefore, when a pixel is identified by *calstis4* as being affected by a cosmic ray hit, it is flagged by including 8192 in the DQ array for that pixel. This procedure is explained in detail in [ISR STIS 98-12](#).

The shifts in the dispersion and cross-dispersion directions are determined differently for first-order grating, prism, and echelle data. First-order grating data are 2-D rectified before being passed to *calstis4*. This is done so the image can be collapsed along columns or rows to make 1-D arrays that are then cross correlated with a template spectrum or a template of the slit illumination to get the shift in the dispersion or cross-dispersion direction, respectively. For a long slit, the features that are used to locate the image in the cross-dispersion direction are the occulting bars, but the cross correlation is not done directly between the slit illumination and a template. The slit illumination pattern is first normalized and inverted, so it is nominally zero where there is light, and it is unity within the occulting bars; this is then cross correlated with a template of the slit that is normalized the same way.

Prism data are handled by *calstis4* in nearly the same way as first-order grating data. The differences are that the data are not rectified first: when collapsing along columns to make a 1-D spectrum, only the middle 224 pixels are used to reduce the effect of any tilt in the spectral lines, and when collapsing along rows the spectral trace (SPTRCTAB) is applied to account for spatial distortion. The reason for working with the flatfielded but not rectified data is that the dispersion for the prism is highly nonlinear, and the shift is in pixel space, not wavelength space.

For echelle data, *calstis4* computes a 2-D cross correlation between the flatfielded wavecal image and a template image. The offset of the peak in the cross correlation gives the shift in both axes. This works because the aperture is small in both dispersion and cross-dispersion directions.

The shifts are initially measured in units of pixels of the wavecal image, but they are then scaled (depending on the binning of the wavecal) to the reference pixel size (unbinned CCD or low-res MAMA). They are subsequently written to the extension header of the 2-D rectified wavecal in the keywords SHIFTA1 (the shift in pixels along AXIS1, or dispersion direction) and SHIFTA2 (the shift in pixels along AXIS2, or spatial direction). The SHIFTA1 and SHIFTA2 keyword values are also copied from the 2-D rectified wavecal file to the flatfielded science extension header. This is the final step performed on the science data prior to 2-D rectification or 1-D extraction of the science data in the pipeline.

Either or both the wavecal file and science file can contain multiple exposures, and the image can drift across the detector over time due to such things as thermal effects, so it is necessary to select the most appropriate wavecal exposure for each science exposure. If there are multiple wavecal exposures, the shifts are linearly interpolated at the time of a given science exposure to get the shifts that will be used for that science exposure. There is also an option to use the wavecal that is nearest in time to the given science exposure.

3.4.23X1DCORR: locate and extract 1-D spectrum

- Reference files: SPTRCTAB, XTRACTAB, SDCTAB

This step is part of spectral extraction. If X1DCORR is PERFORM, *calstis* will locate a one-dimensional spectrum to extract and flux calibrate the spectrum, as explained below, and in more detail in [ISR STIS 99-03](#).

Locate the Spectrum

The nominal location of the spectrum is specified in the spectrum trace table, SPTRCTAB, and is given by (A1CENTER, A2CENTER) from this table. These coordinates are not constrained to be integers. The nominal position along the slit must be modified to include the previously updated position information found in the header. The nominal A2CENTER

position of the spectrum (i.e., the position of the target along AXIS2, or the slit direction) is calculated as follows:

$$A2CENTER = A2CENTER + SHIFTA2$$

where A2CENTER is the detector line number at which (or with respect to which) reference data were measured, and SHIFTA2 is the uncertainty in MSM position as measured by *calstis4* from wavecal observations. SPTRCTAB also contains the description of the distorted shape of the spectrum. The shape is stored as an array consisting of pixel offsets (in the AXIS2 direction) relative to the nominal center of the spectrum (A2CENTER). This spectrum trace is used to find, and eventually to extract, the 1-D spectrum.

The location of the spectrum is improved by searching in the vicinity of the nominal location of the spectrum by performing a cross-correlation between the distortion vector and the input spectrum image. The search extends for $\pm n$ pixels around the nominal center, where n is read from the MAXSEARCH column in the XTRACTAB table. At each AXIS2 position in the search range (which differs from the nominal center by an integer number of pixels), a sum of the counts along the spectrum shape is formed. This sum is created by adding the value of one pixel's worth of data at each of the AXIS1 pixel positions. The pixel extracted in the AXIS2 direction is centered on the spectrum position (A2CENTER + pixel offset) and may include fractional contributions from two pixels. Quadratic refinement is used to locate the spectrum to a fraction of a pixel.

The final A2CENTER becomes:

$$A2CENTER = A2CENTER + CRSCROFF$$

where CRSCROFF is the offset found during the cross correlation. If the cross correlation fails, the value of CRSCROFF is set to zero, a warning message is written to the output, and the A2CENTER calculated prior to the cross correlation attempt is used as the location of the spectrum. CRSCROFF is written to the header of the output spectrum table. The errors in X1DCORR are combined in quadrature, as described in [ISR STIS 98-13](#).

Extract the 1-D Spectrum

The extraction of the spectrum is defined by a triplet of extraction *boxes* found in the reference table, XTRACTAB; one extraction box is used for the spectrum, and the other two are used for the background. The background regions may be tilted with respect to the image axes, but the spectral region is not tilted, to avoid interpolation. For each pixel in the dispersion direction, **calstis** sums the values in the spectrum extraction box. The extraction box is one pixel wide and has a height determined from the EXTRSIZE column in XTRACTAB, centered on the spectrum. The extraction height can be overridden by the `extrsize` parameter in **x1d**. Each endpoint of the extraction box may include a fractional part of a pixel. In the case of a fractional pixel, **calstis** will scale the counts in the given pixel by the fraction of the pixel extracted. Thus, each pixel in the output

spectrum consists of the sum of some number (or fraction) of pixels in the input image.

The extraction of the spectrum allows for unweighted or optimal extraction. The extraction algorithm is selected based on the value of the reference table parameter XTRACALG. This flag has possible values of UNWEIGHTED and OPTIMAL. The value of XTRACALG is written to the header of the output spectrum data file. At present, **calstis** performs an unweighted extraction of the 1-D spectra; the optimal extraction algorithm has not yet been implemented. At the end of the 1-D extraction step, a spectrum of gross cts/sec is produced.

3.4.24X2DCORR: produce rectified 2-D spectral image

- Reference files: DISPTAB, INANGTAB, APDESTAB, SPTRCTAB

This step is part of spectral extraction and applies to two-dimensional extraction. If X2DCORR is PERFORM, a two dimensional rectified spectral image will be produced for spectroscopic data. The two-dimensional rectified output image (`_x2d.fits` or `_sx2.fits`) will have a linear wavelength scale and uniform sampling in the spatial direction. The dispersion direction is the first image axis (AXIS1). The size of the rectified image is made somewhat larger (the increase can be substantial for subarrays) than the input in order to allow for variations in heliocentric correction and offsets of the spectrum on the detector. The binning of the output image will be approximately the same as the input. For each pixel in the output rectified image, the corresponding point is found in the input distorted image, and bi-linear interpolation is used on the four nearest pixels to determine the value to assign to the output. No correction for flux conservation is applied, as this is included in the flatfield.

Mapping from an output pixel back into the input image makes use of the dispersion relation and one-dimensional trace table. The dispersion relation gives the pixel number as a function of wavelength and spectral order. The one-dimensional trace is the displacement in the cross-dispersion direction at each pixel in the dispersion direction. Both of these can vary along the slit, so the dispersion coefficients and the one-dimensional trace are linearly interpolated for each image line. Corrections are applied to account for image offset, binning, and subarray. The spectrum can be displaced from its nominal location on the detector for several reasons, including Mode Select Mechanism (MSM) uncertainty, deliberate offsets for distribution of charge extraction for MAMA data, and the aperture location relative to a reference aperture. These offsets are corrected by modifying the coefficients of the dispersion relations and by adjusting the location of the one-dimensional trace. See also DISPCORR and FLUXCORR for algorithmic details. The process of dispersion

solution, spatial rectification, and wavelength calibration is similar for one-dimensional and two-dimensional spectral-extracted data. However, the flux calibration and error estimates are somewhat different for one- and two-dimensional extractions (see *ISR STIS 98-13*).

3.5 Recalibration of STIS Data

Sometimes the default pipeline calibration, performed shortly after the data were obtained from the telescope, is not the best possible calibration for your science program. There are a number of reasons why it may be desirable to recalibrate your data. The most likely reasons include:

- More appropriate reference files have become available since the data were taken. CCD darks and biases are examples of reference files that are updated frequently, but they require some time to be installed in the pipeline.
- Contemporaneous CCD flatfields were obtained with the science data for G750L or G750M NIR observations to remove fringing.
- Some steps need to be repeated with different input parameters. For example, you may wish to re-perform the cosmic ray rejection or the 1-D spectral extraction after adjusting the input parameters. The best target and background extraction regions for extracting 1-D spectra can depend on the science goals of the program.

Because the data archive now offers On-The-Fly-Reprocessing (OTFR), the simplest way to recalibrate your data with the most appropriate references files is to simply rerequest the data from the archive. However, to tailor the calibration to your individual preferences, it may be beneficial to run **calstis** yourself, on your local machine. The IRAF version of **calstis** is available in the **hst_cal.stis** package in **stdas**.

The STIS calibration pipeline was designed to accommodate the need for full or partial recalibration. As mentioned at the beginning of this chapter, **calstis** is re-entrant, so that certain calibration steps can be performed outside of the pipeline, and others can be executed multiple times, depending upon the science goals.

Generally, the calibration switches in the header control the operations that **calstis** performs on the data. There are three basic ways to select which operations are performed during calibration:

- Edit the calibration switches and run the **calstis** task.
- Use one or more of the pipeline subset tasks described below, managing the calibration through task parameters.

- Run the **calstis** sub-tasks at the host level (i.e., outside of IRAF) using the command-line switches and flags to control the processing.

This section describes the first two methods. In the end, the calibration switches in the headers of the calibrated data files will reflect the operations performed on the calibrated data and the reference files used.

3.5.1 Mechanics of Full Recalibration

You have chosen to fully recalibrate your STIS data. There is a certain amount of set-up required for **calstis** to run properly. The operations mentioned in the checklist below will be described in detail in the following subsections:

- Set up a directory with the required reference files.
- Determine which reference files are needed and retrieve them from the Archive.
- Set the environment variable **oref** to point to your reference file directory. *Note:* you must do this before starting an IRAF session!
- In an IRAF session, update the input data file headers using **chcalpar**.
- Set the calibration switches to perform the needed steps.
- Update the reference file names.
- Run **calstis** or a subset of its constituent tasks.

Retrieve Reference Files

To recalibrate your data, you will need to retrieve the reference files used by the different calibration steps to be performed. The names of the reference files to be used during calibration must be specified in the primary header of the input files, under the section “CALIBRATION REFERENCE FILES.” Note that the data headers will be populated already with the names of the reference files used during pipeline calibration at STScI.

Chapter 1 of the HST Introduction describes how to obtain the best available reference files from the HST Data Archive using StarView. Now that we are in the era of On-The-Fly-Reprocessing (OTFR), the reference files available from Starview are always the most recent reference files.

If better calibration reference files have become available since the original pipeline calibration, they can be retrieved from the HST Data Archive as explained in chapter 1 of the HST Introduction. These files might contain updated information about the instrument signatures, such as an improved background bias level.

The STIS reference files are all in FITS format, and can be in either IMAGE or BINTABLE extensions. The names of these files along with

their corresponding primary header keywords, extensions, and format (image or table), are listed in chapter 2. The (somewhat obscure) rootname of a reference file is based on the time that the file was delivered to the Calibration Data Base System (CDBS).

Set up the Directory Structure for Running Calstis

Before running **calstis**, you will need to define an environment variable to indicate the location of the directory containing the needed calibration reference files. The names of the calibration files are preceded with the logical path name “oref\$” in the STIS science headers. Ordinarily you would define this directory in an IRAF session to be, for example, “/data/vega3/stis/cal_ref/” using the **set** command:

```
cl> set oref "/data/vega3/stis/cal_ref/" # Won't work!
```

Note the trailing slash (/). However, **calstis** and all of its modules are actually foreign tasks and as such do not access IRAF environment variables. Therefore, *before invoking the cl*, you will need to define an environment variable from the host command line (see below) that is appropriate to your host machine. For Unix systems, the appropriate command for the example above is:

```
% setenv oref /data/vega3/stis/cal_ref/
```

Then start IRAF. Note that an alternative to using the oref\$ variable is specifying the full pathnames to the reference files.



When running calstis or any of its modules, you must define environment variables (such as oref\$) before starting the cl. It is not possible to define them within IRAF using the set command, nor is it possible to define them with an escape to the host level, such as:

```
!setenv oref /data/vega3/stis/cal_ref/
```

Edit the Calibration Header Keywords

To edit file headers in preparation for recalibration, use the STSDAS task **chcalpar**. The **chcalpar** task takes a single input parameter: the name(s) of the raw data files to be edited. When you start **chcalpar**, the task automatically determines that the data are from STIS, determines the detector used and whether the observing mode was SPECTROSCOPIC or IMAGING, and opens one of four STIS-specific parameter sets (*pset*) that will load the current values of all the calibration-related keywords. To edit the calibration keyword values:

1. Start the **chcalpar** task, specifying a list of images in which you want to change calibration keyword values. If you specify more than one image (using wildcards, for example) the task will read the initial keyword values from the first image in the list. For example, you could change keywords for all STIS raw science images in the current directory (with initial values from the first image), using the command:

```
ct> chcalpar o*_raw.fits
```

2. After starting **chcalpar**, you will be placed in **eparam**—the IRAF parameter editor; from there you will be able to edit the set of calibration keywords. Change the values of any calibration switches, reference files or tables to the values you wish to use for recalibrating your data.
3. Exit the editor when you are done making changes by typing “:q” two times. The task will ask if you wish to accept the current settings. If you type “y”, the settings will be saved and you will return to the IRAF c1 prompt. If you type “n”, you will be placed back in the parameter editor to redefine the settings. If you type “a”, the task will abort and any changes will be discarded.

The parameter editor screen for STIS MAMA spectroscopy is illustrated in figure 3.1. The characters “oref\$” preceding the names of the reference files specify a logical directory for the location of the reference files. The method for setting a corresponding environment variable is given in the next subsection.

Figure 3.1: Editing Calibration Keywords with chcalpar

```

xterm
IRAF
Image Reduction and Analysis Facility
PACKAGE = ctools
TASK = ckwstis4

<dqicorr=      perform) initialize data quality?
<lorscor=      perform) convert to low-res?
<glinco=       perform) correct global nonlinearity?
<lflgcor=      perform) flag nonlinearity?
<darkcor=      perform) dark correction?
<flatcor=      perform) flat field correction?
<wavecor=      perform) use wavecal?
<dispcor=      perform) use dispersion solution?
<helcor=       perform) heliocentric correction?
<fluxcor=      perform) convert to absolute flux?
<x2dcor=       perform) rectify 2-D spectral image?
<x1dcor=       perform) 1-D spectral extraction?
<backcor=      perform) subtract background?
<rptcor=       omit) add individual repeat obs?
<bpixtab= oref$h1v11477o_bpx.fits) bad pixel table
<mlintab= oref$h1v15598o_lin.fits) MAMA linearity correction table
<darkfil= oref$h1v1208fo_drk.fits) dark reference file
<pfltfil= oref$h2i1352co_pfl.fits) flat field reference file
<dfltfil=      ) delta flat reference file
<lfltfil= oref$h2i1352bo_lfl.fits) low order flat reference file
<apertab= oref$h1v1141oo_apt.fits) aperture throughput table
<apdesta= oref$h1v1126no_apd.fits) aperture description table
<sptrecta= oref$h4s1350fo_1dt.fits) 1-D spectrum trace table
<disptab= oref$h1v1530to_dsp.fits) dispersion coefficients table
<inangta= oref$h1v1541eo_iac.fits) incidence angle correction table
<lamptab=      ) template lamp spectrum
<sdctab = oref$h5e1312fo_sdc.fits) 2-D spectral extraction parameters
<phottab= oref$h2315583o_pht.fits) photometry calibration table
<xtracta= oref$h4s1350ho_1dx.fits) 1-D spectral extraction table
<instrum=      stis) Instrument represented by this pset
<detecto=      mama) Detector represented by this pset
<obstype=      spectroscopic) Obstype represented by this pset
<Version=      10Jul1997) Date of Installation
<mode =        al)

ESC-? for HELP

```

It is also possible to use **hedit** to update the input file keywords. The example below illustrates how to turn on the bias correction switch and update the name of the bias image reference file for all STIS raw images in the current directory that begin with the characters “o3y.”

```

cl> hedit o3y*_raw.fits[0] biascorr PERFORM up+
cl> hedit o3y*_raw.fits[0] biasfile "oref$new_bias.fits" up+

```



If you are changing keywords that reside in the FITS primary header unit with hedit, be sure to explicitly specify the primary header by appending “[0]” to the FITS file name.

For each task (except **calstis**), one cannot simply specify which calibration steps to perform in the primary header keywords of the input files. The execution of each step *must* be specified in the input parameters of each stand-alone task; however, if the input parameter of a task is set to **perform** and the corresponding header keyword is already set to **COMPLETE**, that correction usually will not be performed (the situation depends upon the task). The only exception is **calstis**, where the switches in the primary header control the calibration steps to be performed.

The reference file names for all the stand-alone tasks and **calstis** have to be specified in their corresponding header keywords. See table 2.3 for the list of reference files, and figure 3.1 through figure 3.7 for their relation to each executable.

3.5.2 Rerunning Subsets of the Calibration Pipeline

Selected portions of the pipeline can be executed with special tasks in the STSDAS **stis** package. The tasks that can be simply used in this fashion are listed in table 3.1 below. See also table 3.1 for the association between **basic2d**, **ocrreject**, **wavecal**, **x1d**, and **x2d** and the components of the **calstis** pipeline. When you run these tasks individually, the calibration parameters usually read from the reference file *must* be entered either as command line arguments or via **epar**; however, most corrections will not be applied if the corresponding header keyword is already set to **COMPLETE** (see the appendix of *ISR STIS 98-26* and section 6 of *ISR STIS 98-10* for details).

The **inttag** task for TIMETAG data will accumulate selected events from the raw event table, writing the results as one or more image sets (imsets) in a single, output FITS file. You can optionally specify an explicit starting time, time interval, and number of intervals over which to integrate, and the collection of imsets will be written to the output file, simulating a REPEATOBS ACCUM observation. Breaking the data into multiple, short exposures can be useful not only for variables but also to improve the flatfielding when the Doppler shift is significant. Once the images have been created, it is straightforward to process them with **calstis** and to analyze the output image or spectra, as appropriate.

The screen messages that appear when running any **calstis** module are equivalent to the trailer file contents delivered with the data.

Table 3.1: calstis Pipeline Calibration Tasks

Task	Description
basic2d	Perform basic 2-D calibration.
inttag	Integrate TIMETAG event list to form an image.
ocrreject	Combine images, rejecting cosmic rays.
sc2dref	Add supporting SC2DCORR keywords to header.
wavecal	Process wavecal images.
x1d	Extract 1-D spectrum.
x2d	Rectify spectral images.

3.6 Updates to calstis

The **calstis** modules evolve and improve with time, as we understand and characterize more fully the on-orbit performance of STIS. It is possible, even likely, that improvements in the **calstis** software will improve the calibration of your data. To determine the version of the software used to calibrate your data, note the value of the **CAL_VER** keyword in the primary header. The following example uses **hselect** to print the rootname, the optical element, and the version of **calstis** for all **_flt** files in the current directory:

```
cl> hselect o*_flt.fits[0] "rootname,opt_elem,cal_ver" yes
```

Watch the STScI Analysis Newsletters (STANs) or consult the STIS WWW pages for any announcement of enhancements to **calstis**. A history of the enhancements made to **calstis** is maintained at http://www.stsci.edu/instruments/stis/calibration/pipe_soft_hist. If you are uncertain whether a given enhancement to **calstis** merits recalibrating your data, contact the Help Desk. Often, it is instructive to recalibrate (or simply rerequest your data via On-The-Fly-Reprocessing) and to determine empirically whether the revised calibration files or software affect the scientific interpretation of your data. If you need to upgrade your version of the **stis** package, contact your IRAF system administrator.

STIS Error Sources

In this chapter. . .

4.1 Summary of Accuracies / 4-1

4.2 Flats / 4-3

4.3 Factors Limiting Flux and Wavelength Accuracy / 4-4

In this chapter we discuss the sources of uncertainty in STIS data, and provide pointers to resources beyond this manual for the latest descriptions of our evolving understanding of these issues.

4.1 Summary of Accuracies

In table 4.1 through table 4.5, the accuracies are listed for each basic observation mode of the STIS: CCD spectroscopy, MAMA spectroscopy, CCD imaging, MAMA imaging, and target acquisition. The pixels in these tables are low-resolution pixels. All accuracies quoted are 2σ limits, and reflect our current understanding of the STIS.

We maintain on the STIS web pages (under “Calibration”) both a table of our ultimate expected accuracies and a table of the accuracies delivered by pipeline data at any one time. The latter is frequently updated as we continue to work through the on-orbit calibration of the STIS, providing updated calibration reference files to the pipeline and the archive for its many modes and capabilities.

We remind you that calibration data are immediately non-proprietary and should you have need for extreme accuracy or urgent results, you may wish to consider direct analysis of the calibration data for your particular observing mode (see also chapter 17 of the *STIS Instrument Handbook* for a description of our on-orbit calibration program).

Table 4.1: CCD Spectroscopic Accuracies

Attribute	Accuracy ¹	Limiting Factors
Relative wavelengths—within an exposure	0.1–0.3 pixel	<ul style="list-style-type: none"> Stability of optical distortion Accuracy of dispersion solutions
Absolute wavelengths—across exposures	0.2–0.5 pixel	<ul style="list-style-type: none"> Thermal stability Derivation of wavecal zero point Accuracy of dispersion solutions
Absolute photometry ^{2,3,4}		Instrument stability, evolution of charge transfer efficiency, and photometric calibration
L modes	5%	
M modes	5%	
Relative photometry ^{2,3} (within an exposure)		Instrument stability, evolution of charge transfer efficiency, and photometric calibration
L modes	2%	
M modes	2%	

1. All accuracies refer to prime wavelength settings and directly calibrated special secondary settings. Intermediate settings have roughly a factor of two less accuracy.
2. Assumes star is well-centered in slit.
3. Assumes use of a 2" wide photometric slit.
4. Photometric accuracies referenced are for continuum sources; equivalent width and line profile measures are subject to other uncertainties (such as spectral purity and background subtraction).

Table 4.2: MAMA Spectroscopic Accuracies

Attribute	Accuracy ¹	Limiting Factors
Relative wavelengths—within an exposure	0.25–0.5 pixel	<ul style="list-style-type: none"> Stability of small-scale geometric distortion Optical distortion Accuracy of dispersion solutions
Absolute wavelengths ¹	0.5–1.0 pixel	<ul style="list-style-type: none"> Thermal stability Derivation of wavecal zero point Accuracy of dispersion solutions
Absolute photometry ^{1,2,3}		Instrument stability and photometric calibration
L modes	4%	
M modes	5%	
Echelle modes ⁴	8%	
Relative photometry (within an exposure) ^{2,3}		Instrument stability and flat fields
L modes	2%	
M modes	2%	
Echelle modes ⁴	5%	Ripple correction accuracy, scattered light subtraction

1. All accuracies refer to prime wavelength settings and directly calibrated special secondary settings. Intermediate settings have roughly a factor of two less accuracy.
2. Assumes star is well-centered in slit.
3. Assumes use of a wide photometric slit.
4. For 0.2x0.2 arcsecond slit. These are typical accuracies which can be 2 to 3 times better or worse as a function of wavelength (see [STIS ISR 98-18](#) for details).

Table 4.3: CCD Imaging Accuracies

Attribute	Accuracy	Limiting Factors
Relative astrometry within an image	0.1 pixel	Stability of optical distortion
Absolute photometry	5%	Instrument stability
Relative photometry within an image	5%	External illumination pattern

Table 4.4: MAMA Imaging Accuracies

Attribute	Accuracy	Limiting Factors
Relative astrometry within an image	0.25 pixel	Small scale distortion stability
Absolute photometry	5%	Instrument stability and calibration
Relative photometry within an image	5%	Flat fields and external illumination

Table 4.5: Target Acquisition Accuracies

Attribute	Accuracy	Limiting Factors
Guide star acquisition	1–2 arcseconds	Catalog uncertainties
Following target acquisition exposure		
Point sources	0.01 arcsecond	Centering accuracy plus plate scale accuracy to convert pixels to arcseconds
Diffuse sources	0.01–0.1 arcsecond	See Chapter 8 of the <i>STIS Instrument Handbook</i>
Following pickup acquisition exposure	5% of the slit width	Number of steps in scan and PSF

4.2 Flats

We intend to update all library flats to a signal-to-noise of at least 100:1 per pixel (a “pixel” for the MAMA refers to 1024 x 1024 native format pixels). The CCD flats have temporal variation of < 1% per year. The MAMA flats have shown some evidence for variation at the 1-2% per resolution element level over roughly year timescales. Due to the limited calibration-lamp lifetimes, we expect to take MAMA flats once per year per detector. As our knowledge grows, we will provide updates on the web pages.

4.3 Factors Limiting Flux and Wavelength Accuracy

4.3.1 Flux Accuracy

The accuracy to which you can trust the absolute flux calibration of your STIS spectroscopic data at slit center is limited by several factors including:

- The accuracy of the absolute sensitivity calibration of the grating and central wavelength setting. The on-orbit absolute sensitivity calibration is determined by observing a standard star, with known absolute flux calibration, well centered in both wavelength and cross dispersion in a large slit. The STIS spectrum of this star is then extracted over a standard aperture extraction box and the sensitivity required to return the known flux from the star is determined as a function of wavelength. The standard aperture extraction box is large enough to be relatively insensitive to spacecraft jitter and breathing but small enough that the signal-to-noise of a typical stellar spectrum will not be degraded. STIS calibration accuracies are defined for the standard aperture extraction box; the standard boxes are mode dependent and are given in the XTRACTAB reference file.
- The accuracy of the calibration of the aperture throughput for the aperture you are using for your science relative to the aperture that was used for the absolute sensitivity calibration.
- The accuracy to which your source is centered in your slit.
- The size of the extraction aperture you use to measure your flux and the accuracy to which the cross dispersion profile is known in the mode in which you are observing. Because the corrections for the aperture extraction can be large (e.g., 30% in the near-infrared and the far-UV) this effect can be important.
- Bias and background subtraction can add considerable additional uncertainty for faint sources or spectra with significant variations in flux, particularly for the echelle modes.
- The importance of scattering, which can play an important role, particularly in the G230LB and G230MB gratings.

Additional uncertainties arise for flux measurements not at slit and field center. These uncertainties are relevant when, for example, you wish to determine relative fluxes in an extended source along the long slit or when you have used POS TARGs to move a target along the long slit. They include:

- The variation in slit throughput along the slit. The slits have 5 micron (corresponding to ~ 0.02 arcsecond) variations along their widths, which for a 0.1 arcsecond wide slit on a diffuse source, would produce a 20% variation in flux. For a point source the variation would be more in the 5% range for that same slit. There are also dust specks with appreciably higher opacity along some places in some slits.
- The accuracy to which the vignetting along the cross-dispersion direction is known for your grating and central wavelength
- The accuracy to which the low-order flatfield along the dispersion direction is known off of field center for your grating and central wavelength.

4.3.2 Wavelength and Spatial Accuracies

The accuracy with which the wavelength scale is known in your calibrated STIS spectrum will depend on several factors:

- The accuracy of the dispersion solutions, which governs the accuracy to which relative wavelengths are known in a given spectrum
- The accuracy of the wavelength zeropoint, which governs the accuracy to which relative wavelengths are known across spectra.
- The accuracy to which your source is centered in the science slit (a pixel of miscentering corresponds to a pixel in absolute wavelength space).

The dispersion solutions used to calibrate STIS on orbit data were derived on the ground during thermal vacuum testing. On-orbit tests confirm the applicability and accuracy of the ground dispersion solutions for on-orbit data, producing relative wavelength accuracies of 0.2 pixels across the spectrum for first-order gratings at the prime settings and appreciably better in some instances. For the echelle modes, at the prime settings, the accuracies are roughly 0.2 pixels for wavelengths in the same order and approximately 0.5 pixels for the entire format. The intermediate wavelength settings have roughly twice these errors.

Due to the lack of repeatability of the Mode Select Mechanism (STIS's grating wheel), the projection of your spectrum onto the detector in both wavelength and space will vary slightly (1 to 10 pixels) from observation to observation (if the grating wheel is moved between observations). In addition, thermally induced motions can also affect the centering of your spectrum. The calstis pipeline removes the zeropoint offsets using the contemporaneous wavecals (see section 3.4.22). The wavelength zeropoint in your calibrated data (the `rootname_sx2.fits`, `_x2d.fits`, `_x1d.fits`, and `_sx1.fits` files) is corrected for these offsets and should have a wavelength zeropoint accuracy of ~ 0.1 – 0.2 pixels (better

when the WAVECAL is taken through small slits, worse for those taken through wider slits). This accuracy should be achieved, so long as contemporaneous WAVECALs were taken along with the science data, distributed at roughly one hour intervals among the science exposures, and assuming the target was centered in the slit to this accuracy or better.

The accuracy of the zeropoint pipeline calibration in the spatial direction is slightly less, roughly 0.2-0.5 pixels. This is because the finding algorithm, which must locate the edges of the aperture for short slits and the edges of the fiducial bars on the slits for the long slits, is less robust. Observers need to be aware of possible offsets between spectra in the spatial direction, particularly when deriving line ratios for long-slit observations of extended targets taken with different gratings.

STIS Data Analysis

In this chapter. . .

5.1 Data Reduction Applications / 5-1
5.2 STIS-Specific Reduction and Analysis Tasks / 5-3
5.3 Working with Imaging Data / 5-4
5.4 Working with Rectified Two Dimensional Spectral Images / 5-5
5.5 Combining Dithered Spectroscopic Data / 5-8
5.6 TIME-TAG Data / 5-9
5.7 Target Acquisition Data / 5-12

In this chapter we discuss the data reduction applications available to work with STIS data, and describe specific analyses you may want to apply to your spectral, timetag, and acquisition data.

5.1 Data Reduction Applications

Most of the software tools for operating on STIS FITS files are contained in two STSDAS packages:

- **toolbox.imgtools.mstools**: Contains image manipulation tasks created especially for STIS and NICMOS image set (*imset*) data and which exploit the error and data quality arrays in their operations (e.g., **msarith**, **mscombine**, **msstatistics**, **mssplit**, and **msjoin**). These tasks are described in section 3.3.3 of the HST Introduction.
- **hst_calib.stis**: Contains the STIS specific tasks including **basic2d**, **calstis**, **inttag**, **ocrreject**, **odelaytime**, **wavecal**, **x1d**, and **x2d**.

In addition to the above packages, most basic image manipulation software (e.g., **display**, **daophot**, **imexamine**, **contour**) and spectral

analysis software (e.g., **splot**, **tables**, **specfit**, **igi**) available in IRAF/STSDAS can be used on STIS data, either directly through the IRAF FITS interface or by converting data to another IRAF format. Chapter 3 of the HST Introduction includes information about how to display STIS images and extracted spectra, as well as how and when to convert data formats, and a description of spectral analysis tasks. Table 5.1 lists some of the more useful IRAF/STSDAS applications for working with STIS data.

Table 5.1: Spectral Analysis Tasks

Task	Input Formats	Purpose
echplot	3-D tables	Plots multiple STIS echelle spectral orders.
nfit1d	2-D & 3-D tables, images	General 1-D feature fitting; part of the STSDAS fitting package.
igi	2-D & 3-D tables, images	General presentation graphics; supports world coordinates.
sgraph	2-D & 3-D tables, images	General 1-D plotting; supports world coordinates.
specfit	1-D images, ASCII tables	General 1-D spectrum modeling package.
splot	<i>multispec</i> images	General 1-D spectral analysis.

Some of the tasks are intended for browsing data or producing publication-quality plots: the **igi** and **sgraph** tasks are described in chapter 3 of the HST Introduction. The **echplot** task is useful for browsing STIS extracted spectra. You can plot single spectral orders, overplot multiple orders on a single plot, or plot up to 4 orders in separate panels on the same page. For example, we overplot the orders contained in rows 2 through 4 and 6 on one page:

```
cl> echplot "stis_xld.fits[1][r:row=(2:4,6)]" output.igi \
>>> plot_style=m
```

Note that the `plot_style` parameter governs whether the spectral orders are plotted one per page, overplotted, or plotted one per panel, for parameter values of “s”, “m”, or “p” respectively. The brightness unit is calibrated FLUX by default, though other quantities (e.g., NET counts) can be specified using the `flux_col` parameter.

5.2 STIS-Specific Reduction and Analysis Tasks

In chapter 3 we discussed the components of the STIS pipeline that can be run as individual tasks. Observers may find that they wish to perform parts of the pipeline reduction again on their data using these tasks. Typical examples will be to re-perform cosmic ray rejection (perhaps altering the input parameters or using data from separate datasets), to perform one-dimensional spectral extraction on long-slit data, or to modify the input parameters (e.g., aperture extraction box, or background region) when doing one-dimensional spectral extraction for echelle data. For completeness we list the tasks again below. To run these tasks you need to retrieve the calibration reference files they require from the Archive and set the `oref` parameter appropriately (see section 3.5.1). An example of using `x1d` with a user-specified extraction box (13 pixels high, instead of the default 11) and a user-specified center (line=500) is given below:

```
cl> x1d stis_flt.fits output=stis.fits a2center=500
extrsize=13
```

Table 5.1 lists some of the tasks for working with STIS data. Consult the on-line help for more information about all of these tasks.

Table 5.1: STIS-Specific Tasks

Task	Description
basic2d	Perform basic 2-D calibration.
calstis	Process STIS data through the calibration pipeline.
daydark	Update a current dark image with hot pixels from daily dark.
defringe	Correct a 750M or 750L spectrum for fringing.
echplot	Generate and plot IGI scripts for extracted spectral orders.
echscript	Generate an IGI script for selected extracted spectral orders.
infostis	Print information about STIS FITS files.
inttag	Integrate TIMETAG event list to form an image.
mkfringeflat	Matching fringes in STIS fringe flat to those in science data.
normspflat	Normalize STIS CCD fringe flat.
ocrreject	Combine images, rejecting cosmic rays.
odelaytime	Corrects for light delay time for STIS time-tag data.
ovac2air	Convert wavelengths from vacuum to air.
prepspec	Prepare a spectrum for subsequent de-fringing procedures.

Task	Description
sc2dref	Add header keywords necessary for SC2DCORR.
sdqflags	Print sum of data quality flags.
tastis	Analyze STIS target acquisition images.
wavecal	Process wavecal images, determine spectral and spatial shifts.
x1d	Extract 1-D spectrum., perform optimal extraction.
x2d	Rectify spectral images.

5.3 Working with Imaging Data

5.3.1 Sensitivity Units and Conversions

Your calibrated image (`_flt` or `_crj` file) has units of counts. The conversion to flux ($\text{erg cm}^{-2} \text{sec}^{-1} \text{\AA}^{-1}$) for a flat spectrum is:

$$f_{\lambda} = \frac{\text{counts} \times \text{PHOTFLAM}}{\text{EXPTIME}}$$

where

- PHOTFLAM is the sensitivity for the observing mode, and
- EXPTIME is the exposure time in seconds.

Both of these parameters are given in the science header.

These fluxes can be converted to magnitudes in the STMAG system by the relation $\text{STMAG} = -2.5 \log_{10} f_{\lambda} - 21.1$.

5.3.2 Spatial Information

At the time of this writing, the only IRAF software available to users for converting between pixel coordinates and Right Ascension and Declination are the **xy2rd** and **rd2xy** tasks, available in the **stsdas.toolbox.imgtools** package. The programs work on images corrected for geometric distortion. We are currently developing new programs that work on STIS images that are uncorrected for geometric distortion (analogous to the **metric** and **invmetric** tasks used in the analysis of WFPC2 data).

5.4 Working with Rectified Two Dimensional Spectral Images

5.4.1 Sensitivity Units and Conversions

The **calstis** pipeline software produces a rectified two-dimensional spectral image when X2DCORR is set to PERFORM (see section 3.4.24). At each pixel i , this image contains the surface brightness per Angstrom, $B_{\lambda i}$, in $\text{erg cm}^{-2} \text{sec}^{-1} \text{\AA}^{-1} \text{arcsec}^{-2}$. The image has a linear wavelength scale and uniform sampling in the spatial direction. Here we review how that image is calculated by the pipeline, and how to convert the data in this image to other quantities.

The flux detected by a pixel in a two-dimensional spectral image is:

$$F_i = \frac{C_{\lambda} \times g \times h \times c}{T_{\lambda}^{\text{sys}} \times A_{\text{HST}} \times \lambda}$$

in $\text{erg cm}^{-2} \text{sec}^{-1}$, where:

- C_{λ} is the wavelength-dependent count rate, which is the ratio of the total counts to the exposure time. The exposure time is given in the EXPTIME header keyword.
- g is the detector gain, which is unity for MAMA observations. For the CCD, this is the conversion from counts to electrons, the value for which is given in the header keyword ATODGAIN.
- h is Planck's constant.
- c is the speed of light.
- T_{λ}^{sys} is the wavelength-dependent integrated system throughput, given in the PHOTAB reference file table. STIS uses throughput curves for individual optical elements.
- A_{HST} is the area of the unobstructed HST mirror.
- λ is the wavelength in Angstroms.

The flux per Angstrom is:

$$F_{\lambda i} = \frac{F_i}{d}$$

in $\text{erg cm}^{-2} \text{sec}^{-1} \text{\AA}^{-1}$, where:

- d is the dispersion in $\text{\AA}/\text{pixel}$, derived from the CD1_1 header keyword.

The surface brightness detected by that pixel is:

$$B_i = \frac{F_i}{W \times m_s}$$

in $\text{erg cm}^{-2} \text{ sec}^{-1} \text{ arcsec}^{-2}$, where:

- m_s is the plate scale in arcsec/pixel in the cross-dispersion direction (i.e., spatial direction). This corresponds to the CD2_2 header key-word value multiplied by 3600 arcsec/degree.
- W is the slit width in arcsec.

The surface brightness per Angstrom, $B_{\lambda i}$, given in the flux-calibrated, rectified, two-dimensional spectra image (`_sx2` or `_x2d` file) is thus:

$$B_{\lambda i} = \frac{F_i}{d \times W \times m_s}$$

in $\text{erg cm}^{-2} \text{ sec} \text{ \AA}^{-1} \text{ arcsec}^{-2}$. So, starting from the rectified two-dimensional spectral image (`_sx2` or `_x2d` file), F_i , $F_{\lambda i}$, and B_i can be computed from $B_{\lambda i}$ as:

$$F_i = B_{\lambda i} \times d \times W \times m_s$$

$$F_{\lambda i} = B_{\lambda i} \times W \times m_s$$

$$B_i = B_{\lambda i} \times d$$

Similarly, fluxes can be summed over regions in the `_sx2` and `_x2d` files and used to compute the flux or surface brightness per Angstrom for the continuum, or flux or surface brightness of a spectral feature (after subtracting off the continuum). For a rectangular region in the spectral image spanning N_λ pixels in the dispersion direction (N_λ is proportional to

Angstroms) and N_s pixels in the spatial direction ($N_s m_s$ arcsec along the slit):

$$F = \sum_i F_i = N_\lambda \times N_s \times \langle F_i \rangle = N_\lambda \times N_s \times \langle B_{\lambda i} \rangle \times d \times W \times m_s$$

$$F_\lambda = \frac{\sum_i F_i}{N_\lambda \times d} = N_s \times \langle B_{\lambda i} \rangle \times W \times m_s$$

$$B = \frac{\sum_i F_i}{W \times N_s \times m_s} = N_\lambda \times \langle B_{\lambda i} \rangle \times d$$

$$B_\lambda = \frac{\sum_i F_i}{N_\lambda \times d \times W \times N_s \times m_s} = \langle B_{\lambda i} \rangle$$

where F is in $\text{erg cm}^{-2} \text{sec}^{-1}$, F_λ is in $\text{erg cm}^{-2} \text{sec}^{-1} \text{\AA}^{-1}$, B is in $\text{erg cm}^{-2} \text{sec}^{-1} \text{arcsec}^{-2}$, and B_λ is in $\text{erg cm}^{-2} \text{sec}^{-1} \text{\AA}^{-1} \text{arcsec}^{-2}$.

No slit loss corrections have been made above. This is correct in the limiting case of spatially uniform surface brightness. For a point source, the flux can be corrected for slit loss by using the data header keyword `DIFF2PT`. This keyword is calculated as:

$$DIFF2PT = \frac{W \times m_s \times H_\lambda(h)}{T^{ap}}$$

where W and m_s are as above, T^{ap} is the wavelength-averaged point-source aperture throughput for the science aperture (which is determined from the reference table specified by the keyword `APERTAB`), and $H_\lambda(h)$ is the wavelength-dependent correction for the extraction slit of height h to a slit of infinite height, which is obtained from the reference table specified by the keyword `PCTAB`. That is, to derive the flux from a point source, integrate the `_x2d` or `_sx2` file over the default extraction slit height (from the `PCTAB`) and multiply the result by `DIFF2PT`. The default extraction slit height for first-order modes is at present 11 pixels for the MAMAs and 7 pixels for the CCD. If the desired extraction slit height differs from the default, the `PCTAB` has a set of wavelength-dependent corrections for selected alternative apertures that must also be applied. See [ISR STIS 98-01](#)

for further details. Of course, point source observers are better advised to use the **x1d** task to extract a one-dimensional spectrum from their long-slit first-order data, which will then apply the wavelength-dependent aperture throughput, the defined point source extraction aperture and calibration, as well as perform background subtraction.

In general we note that the cross dispersion profiles can be quite extended (particularly in the far-UV and in the near-infrared). Fluxes derived for extended sources from the `_x2d` files, as above, assume that the sources are extended on scales that contain the bulk of the cross dispersion flux from a point source. Encircled energies for the first-order modes can be found in *ISR STIS 97-13*.

See also Chapter 6 of the *STIS Instrument Handbook* for a more detailed discussion of units and conversions for different source types.

5.4.2 Wavelength and Spatial Information

Two-dimensionally extracted spectra have been wavelength calibrated and rectified to a linear wavelength scale. Tasks such as **splot** can work directly on the `_sx2.fits` and `_x2d.fits` files and can read the wavelength header information that is stored in the standard FITS CD-matrix keywords. Alternatively you can use these keywords directly to determine the wavelength or distance along the slit at any pixel as:

$$\lambda(x) = CRVAL1 + (x - CRPIX1) \times CD1_1$$

$$s(y) = (y - CRPIX2) \times CD2_2$$

where $\lambda(x)$ is the wavelength at any given x pixel, and $s(y)$ is the distance along the slit from slit center for any given y pixel, in units of degrees.

5.5 Combining Dithered Spectroscopic Data

New tasks are under development to handle dithered spectroscopic data. These tasks will be added to **stsdas.hst_calib.stis**. At present, one can use a combination of **imshift** in IRAF and **imcalc** in **stsdas.toolbox.imgtools** to achieve a shift of an integer number of pixels along Y, the spatial dimension in a spectroscopic image. Dithering by a few pixels along Y is recommended to allow removal of hot pixels when CCD images are combined. One first performs bias, flat, and dark correction of the raw images with **basic2d**. Some images must then be shifted up or down to compensate for dithering away from the target position. Because **imshift** works only on a single extension of an image, the SCI, ERR, and DQ images must be copied and shifted separately. **imcalc** can then be used to put these 3 images back into a standard STIS fits file, as follows: Use **imreplace** in IRAF to set all data values to 0 in a copy of the original

basic2d output, `copy.fits`. Then, using **imcalc**, add the shifted SCI image to `copy.fits[1]` with output to `copy.fits[1]`; add the shifted ERR image to `copy.fits[2]` with output to `copy.fits[2]`; add the shifted DQ image to `copy.fits[3]` with output to `copy.fits[3]`. When all necessary shifting is complete, **ocrreject** can be applied to the set of images to combine them. Hot pixels will be rejected along with cosmic rays.

5.6 TIME-TAG Data

STIS MAMA detectors can be used in ACCUM or TIME-TAG modes, as described in chapter 11 of the *STIS Instrument Handbook*. In TIME-TAG mode, the position and detection time of every photon is recorded in an event list. Detection times are recorded with 125 microsecond precision, although events from bright sources may be buffered for as long as 128 milliseconds prior to assignment of a detection time.

For TIME-TAG datasets, the HST archive returns all normal data products and also an event list in a file with a `_tag` suffix. The `_tag` file is a FITS file with binary table extensions. Files retrieved from the HST archive prior to 2001-Sep-13 have one FITS extension for each STIS buffer dump. The number of buffer dumps equals the largest integer less than or equal to the exposure time divided by the requested BUFFER-TIME, except that there is always at least one dump. Files retrieved from the HST archive from 2001-Sep-13 onwards have all buffer dumps combined into the first FITS extension. Regardless of when a `_tag` file was retrieved, the last extension is a list of good time intervals, indicating time intervals when events could have been detected. TIME-TAG data retrieved from the HST archive prior to 2001-Sep-13 may contain a variety of errors, including gaps in event lists, errors in event times, or flux calibration errors in extracted spectra. These errors occurred infrequently, but TIME-TAG data retrieved prior to 2001-Sep-13 should nonetheless be discarded in favor of newly retrieved data.

An event list in a `_tag` file is a FITS binary table extension named EVENTS, containing four columns named TIME, AXIS1, AXIS2, and DETAXIS1:

The TIME column contains the time when each event was recorded, relative to the start time (MJD) of the exposure given in the TEXPSTRT keyword of the primary FITS header. Each relative time is stored as an integer number of 125 microsecond hardware clock ticks, but application of the column scale factor converts relative times to seconds. Application of the column scale factor is automatic in the IRAF **tables** package.

The AXIS1 column contains the pixel coordinate along the spectral axis where each event was recorded plus a correction term to remove Doppler shifts introduced by the orbital motion of HST. The correction term depends on optical element and the projected orbital velocity of HST,

which varies over the course of an observation. In ACCUM mode, this Doppler compensation is applied during an observation, but in TIME-TAG mode raw positions are downlinked and Doppler compensation is applied during ground processing.

The AXIS2 column contains the pixel coordinate along the spatial or cross-dispersion axis. No Doppler compensation is applied.

The DETAXIS1 column contains the pixel coordinate along the spectral axis where each event was recorded with no correction for Doppler shifts introduced by the orbital motion of HST. In general, AXIS1 coordinates are more relevant than DETAXIS1 coordinates.

After all EVENTS extensions in a `_tag` file, there will be one final binary table extension named GTI, containing columns named START and STOP. There will be associated start and stop times for every uninterrupted observing interval during a planned exposure. For most datasets, there will be only one START and one STOP time encompassing all buffer dumps in an exposure. Multiple good time intervals are possible, however - for example, if guide star lock is lost. Times in START and STOP are expressed in seconds since the start time (MJD) of the exposure given in the TEXPSTRT keyword of the primary FITS header. In IRAF, good time intervals can be examined using the **tprint** task in the **tables** package: `tprint rootname_tag.fits[GTI]`, where `rootname` must be replaced by the rootname of the `_tag` file being examined.

Useful IRAF tasks for analyzing and manipulating data taken in the TIME-TAG observing mode are listed in table 5.1.

Table 5.1: Useful IRAF tasks for reducing TIME-TAG Data

Task	Purpose
tpar, tprint	Read the value for the EVENTS table header keywords
hstephem	Convert the ephemeris file into an STSDAS table
odelaytime	Apply the light delay correction
inttag	Convert TIME-TAG events into an ACCUM image
calstis	Calibrate the ACCUM image and extract a 1-D spectrum
sgraph	Display the 1-D spectrum

5.6.1 Heliocentric and Barycentric Time Correction

TIME-TAG observations with STIS have a temporal resolution of 125 microseconds. The effects on the observed times of both the orbital motion of the Earth and the HST may need to be corrected relative to the solar system barycenter. The **odelaytime** task creates a file of observation time events corrected for light delay from three sources: 1) general relativistic

effects (up to 2 milliseconds), 2) displacement of the telescope from the center of the Earth (up to 20 milliseconds), and 3) displacement of the Earth from the solar-system barycenter (up to 500 sec). The inputs to **odelaytime** are the TIME-TAG data and the ephemeris files. For details regarding the **odelaytime** task and parameters, the reader is referred to *ISR STIS 2000-02*.

5.6.2 Converting TIME-TAG Data to an Image

The **inttag** task in IRAF was developed to convert the TIME-TAG table (EVENTS tables) into subexposures for a given time interval (provided by the user). The result is similar to the ACCUM mode data. The default start time here is the first value in the GTI table. The time increment value can be checked so that each sub-exposure has reasonable signal-to-noise. Details regarding the **inttag** task and its parameters are summarized in *ISR STIS 2000-02*.

5.6.3 Echelle TIME-TAG Data

STIS echelle TIME-TAG data require longer processing and calibration time because they consist of many orders, especially if multiple subexposures are used. After converting the TIME-TAG data into an image and calibrating it, the IRAF tasks **txtable** can be used to extract wavelength dependent data for a single order from the `_x1d` file.

```
cl> txtable "rootname_x1d.fits[3][r:sporder=116]\
[c:wavelength,flux]" table.tab
```

where [3] refers to the third subexposure, the desired order is 116, and the wavelength and flux columns are written to the output table `table.tab`. A useful task to display echelle spectra is **echplot**.

5.6.4 Timing Analysis Tasks

Apart from the **inttag** task which can be used to create subexposures for a fixed time interval, other tasks are also available for timing analysis of the STIS TIME-TAG data. For example, the timing analysis tasks in IRAF that are part of the **xray** data analysis software may be accessed and used in reducing the TIME-TAG data. However, to use STIS TIME-TAG data with the **xray** tasks requires that the FITS file be converted into QPOE format. In order for this conversion to be done successfully, the TIME column values must be in double precision format and the keywords such as "TIME=XS-SORT" must be in the science header. A script to convert STIS TIME-TAG FITS data into QPOE format is given in the appendix of *ISR STIS 2000-02*.

5.6.5 Merging TIME-TAG Event Tables

Many TIME-TAG data sets will consist of several EVENTS tables in a single FITS file, depending on the number of buffer dumps. In such cases it is convenient to merge them into one EVENTS table. To check the number of event binary tables one could use the IRAF task **catfits**. To merge the EVENTS tables in IRAF, the **tmerge** task should be used. More details about these tasks are provided in *ISR STIS 2000-02*.

5.6.6 TIME-TAG Data Calibration

After processing with **inttag**, the `_raw` data output may be calibrated using **calstis**. Here, the same rules apply as with STIS data taken in other modes. However, two calibration steps that require particular attention in case of TIME-TAG data are the Doppler correction and the spatial resolution correction. These are explained in detail in *ISR STIS 2000-02*.

5.7 Target Acquisition Data

5.7.1 Target Acquisition Basics

There are two types of STIS target acquisition: ACQ and ACQ/PEAK; for more details on target acquisition, see *ISR STIS 97-03*. For ACQ observations, there are three parts to the target acquisition data that you receive. The first is an image of the target in the target acquisition sub-array (100 x 100 pixels for POINT sources, user-defined for DIFFUSE sources) based on the initial pointing (see figure 5.1a, `image_raw.fits[sci,1]`). The software then determines the position of the target with a flux-weighted pointing algorithm and calculates the slew needed to place the target at a reference point in the target acquisition sub-array; for DIFFUSE sources, an option to perform a geometric centroiding is available. An image of the target at this corrected position is then obtained (see figure 5.1b, `image_raw.fits[sci,2]`)—this is the coarse centering. To perform the fine centering (i.e., to place the object precisely in a slit), a 32 x 32 pixel image of the reference 0.2X0.2 aperture is obtained (see figure 5.1c `image_raw.fits[sci,3]`), and the location of the aperture determined. A fine slew is then performed to center the target in the reference aperture, which should be accurate to 0.2 pixels (0.01 arcsec). A final slew to center the target in the science aperture is performed at the start of the following science observation.

Table 5.1: Measured Brightness for STIS ACQ Image

Type	Enclosed	Gaussian	Direct
Star	1.59	1.57	1.49
Sky	15.22	142.84	4.81

If a narrow slit is used for the science, an ACQ/PEAK acquisition may have been performed. The slit is scanned across the object with a pattern determined by the aperture selected. The telescope is then slewed to center the target in the aperture, and a confirmation image (a 32 x 32 pixel grid) is obtained; the accuracy of the ACQ/PEAK is 5% of the slit width. There is only one image in the ACQ/PEAK data file, showing the confirmation image through the slit, and associated with the extension [`sci,1`]. The last extension in the file (`image_raw.fits[4]`) contains the values in the individual steps of the ACQ/PEAK (use **listpix** to view these values).

When examining the confirmation image, note that the slit will be illuminated by the sky even if no target is present (see figure 5.1; `image_raw.fits[sci,1]`). To confirm the presence of a compact target, use the **imexamine** task and make certain that the FWHM is small. The measured values for the images in figure 5.1 are given in table 5.1.

Figure 5.1: Three Stages of an ACQ Observation

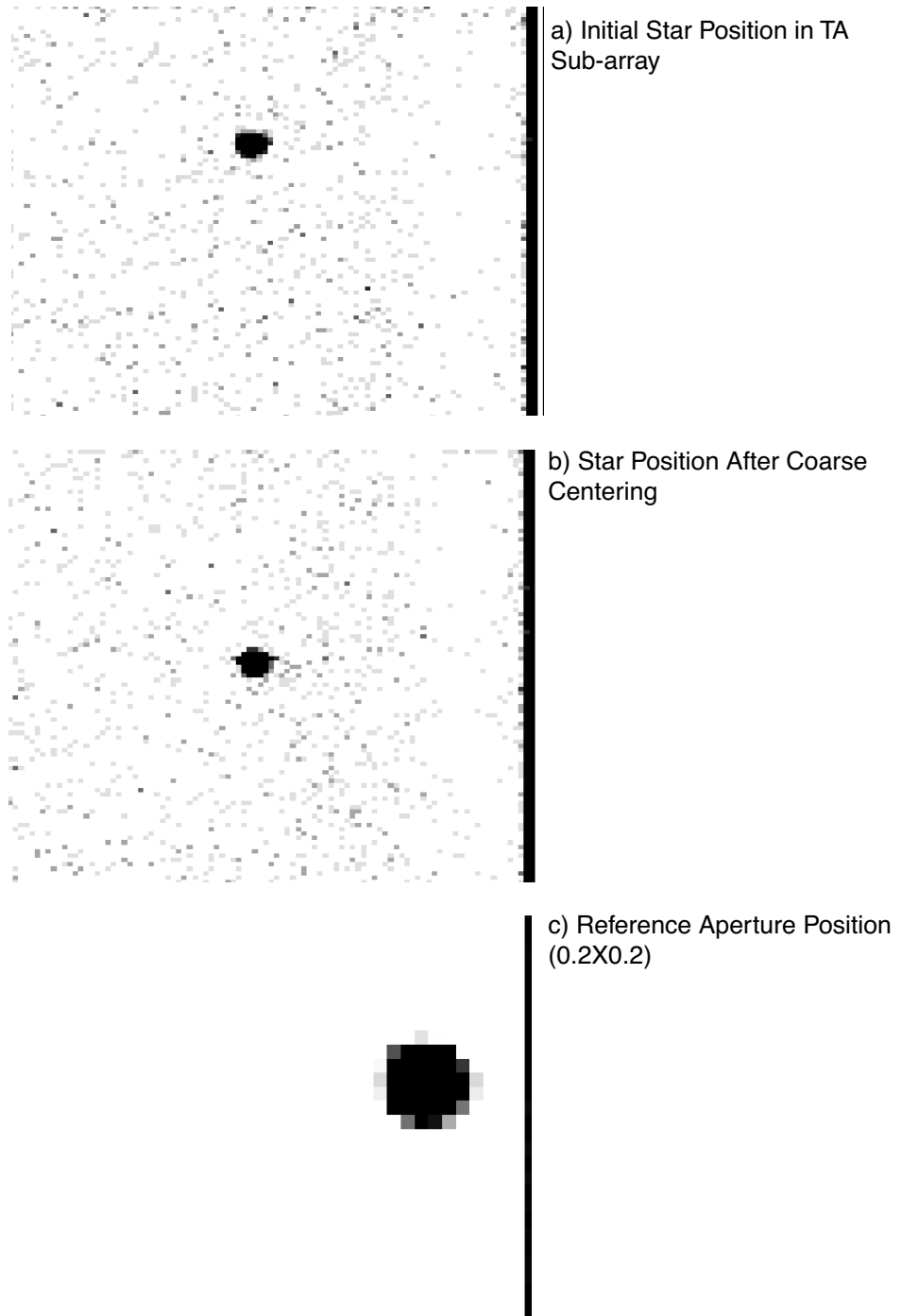
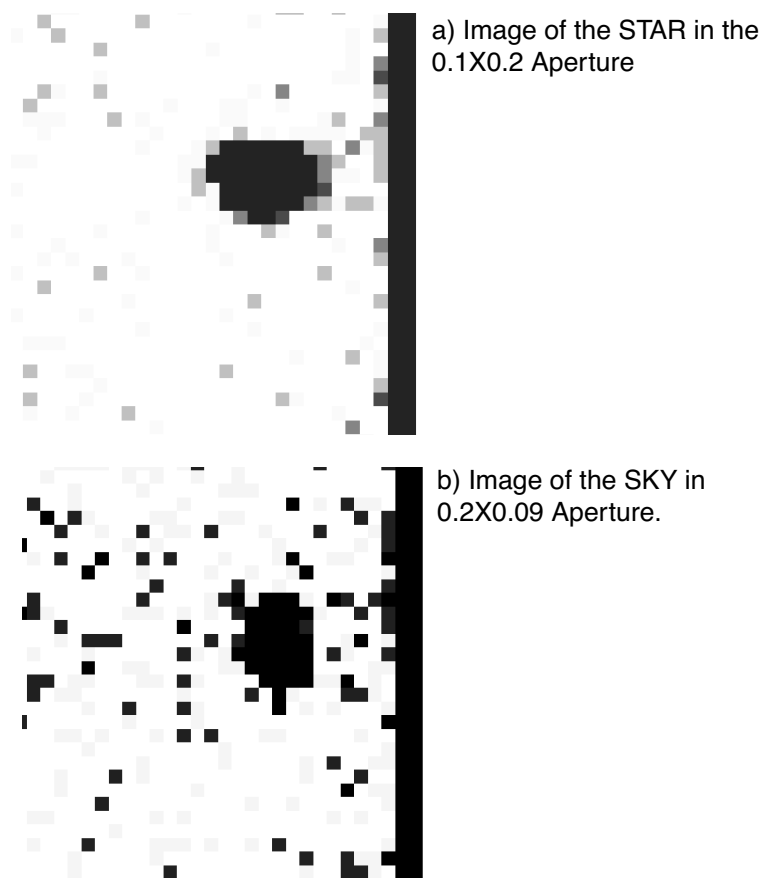


Table 5.2: ACQ/PEAK Confirmation Images



5.7.2 ACQ Data

An examination of the target acquisition data (either from the raw data or the paper products) will allow you to detect gross errors in the centering of your target. Note that target acquisition data are always uncalibrated. A comparison of the initial `[sci, 1]` and post-coarse slew `[sci, 2]` images should show the object moving close to the center of the acquisition sub-array.

You can also verify that the fluxes in both images, which are found in the science header under the keyword `MAXCHCNT`, are consistent by performing the following steps in `IRAF`:

```
c1> imheader image_raw.fits[1] long+ | grep MAXCHCNT
c1> imheader image_raw.fits[4] long+ | grep MAXCHCNT
```

The first value will be the target flux in the maximum checkbox (3x3 for POINT sources, user-defined for DIFFUSE sources) in the initial image,

while the second is the maximum checkbox in the post-coarse slew image. If the fluxes are not consistent, or if the object did not move closer to the center of the array, there is likely a problem with your acquisition.

5.7.3 ACQ/Peak Data

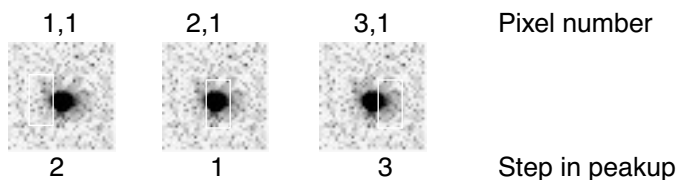
To verify that the ACQ/PEAK worked, examine the flux values at each stage of the peakup (listed in the paper products or in the data file). The fluxes can be found by looking at the fourth extension (`image_raw.fits[4]`) of the peakup data via the IRAF command:

```
cl> listarea image_raw.fits[4]
```

Sample	1	2	3
Line			
1	4707.	260769.	0.

For a 3-step linear peakup, the pixel [1,1] is the leftmost scan position, [2,1] is the middle position, and [3,1] is the rightmost position. See figure 5.1.

Figure 5.1: Flux Values at Each Stage in Peakup



For a 9-step spiral pattern, pixels [1,1] through [3,1] are the lower row of the pattern, [1,2] through [3,2] are the middle row, and [1,3] through [3,3] are the upper row. Note that one of the steps will always have a value of zero.

To determine the flux in the confirmation image, do the following in IRAF:

```
cl> imstat image_raw.fits[1] fields="image,npix,mean"
```

This will give you an output like the following:

```
cl> imstat o4de01jdq_raw.fits[1]
```

#	IMAGE	NPIX	MEAN
1	o4de01jdq_raw.fits[1]	32704	8.241

The total counts in the image is the product of the number of pixels (NPIX) and the mean value (MEAN), or 269,514 in this example.

Note that you will need to perform one correction to the mean value prior to your comparison. The flux values in the peakup scan have been adjusted to subtract the minimum flux value in the peakup data (which is why one value in the peakup is always zero). This value needs to be subtracted from the counts in the confirmation to do a proper comparison. The value can be found in pixel 712 of the `_spt` image (i.e., support, planning and telemetry file). To display the value on the screen, type the following in IRAF:

```
cl> listpix image_spt.fits[1] | grep 712
```

In the example, the value was 6008, which means the corrected number of counts in the confirmation image is 263,506.

Comparison of the maximum flux value during the peakup (260,769) with the flux in the post-ACQ/PEAK confirmation image (263,506) should show that the flux in the confirmation image was greater than or equal to the maximum flux in the peakup grid. If this is not the case, then there is likely a problem with your peakup acquisition.

5.7.4 The `tastis` Task

The `stsdas.hst_calib.stis` package contains a task, `tastis`, that will print general information about each input target acquisition image, and will analyze both types of STIS target acquisitions: ACQs and ACQ/PEAKs.

For an ACQ, target positions, in global and local (subarray) coordinates, and the total flux of the target in the maximum checkbox during both acquisition phases (coarse and fine) are displayed; the location of the reference aperture (used during the fine locate phase) is also displayed. For an ACQ/PEAK, the flux values at each step during the peakup, the total flux in the post-slew confirmation image, and the pedestal value subtracted from each dwell point are displayed.

The task will calculate the slews performed during the acquisition, and provide warnings of possible problems that could indicate a failure of the acquisition:

ACQ:

- If the fine slew (to center the target in the reference aperture) is greater than 4 pixels, it may indicate that the target found in the coarse locate phase was different than that found in the fine locate phase.
- If the ratio of the flux in the maximum checkbox in the coarse and fine stages differs by more than 10%, it may indicate different targets were found in each phase.

ACQ/PEAK:

- If the flux in the confirmation image (minus the pedestal value) is less than 90% of the maximum flux in the pickup, it may indicate that the source is not well centered.
- If the flux test failed because no point in the pickup scan has a flux that is at least 30% higher than any other point, it may indicate no target was found.
- If the first or last (left and right) flux value in the scan is not 0, it may indicate contamination from a nearby object.

If any acquisition in the input list did not succeed according to the above criteria, then a message to that effect will be printed.



PART III:

Appendixes

■ Part III:Appendixes



APPENDIX A:

IRAF Primer

In this appendix . . .

A.1 Initiating IRAF / A-2
A.2 IRAF Basics / A-4
A.3 Getting IRAF and STSDAS / A-15

The Image Reduction and Analysis Facility (IRAF), developed by the National Optical Astronomy Observatories (NOAO), forms the basis of the Space Telescope Science Data Analysis System (STSDAS). IRAF contains numerous packages of programs, called *tasks*, that perform a wide range of functions from reading data tapes to producing plots and images. Most astronomers will already be familiar with IRAF, but we provide this tutorial for HST observers who are beginners with IRAF. It includes information on:

- How to set up IRAF the first time you use the software.
- How to start and stop an IRAF session.
- Basic concepts, such as loading packages, setting parameters, etc.
- How to use the on-line help facility.

Additional information on IRAF, in particular *A Beginner's Guide to Using IRAF* is available through the NOAO IRAF Home Page at: <http://iraf.noao.edu>

A.1 Initiating IRAF

This section explains:

- How to set up your IRAF working environment.
- How to start and logout of the IRAF program.



We assume that your site has IRAF and STSDAS installed. If not, you must obtain and install the software. See appendix section A.3 for details.

A.1.1 Setting Up IRAF

Before running IRAF for the first time you need to follow these three steps:

1. Create your IRAF root directory.
2. Move to that directory and set the necessary environment variables or system logicals and symbols.
3. Run **mkiraf** to create a `login.cl` file and a `uparm` subdirectory.

Users generally name their IRAF home directory `iraf` (also referred to as your IRAF *root* directory) and set it up in their account's root directory (i.e., the default directory that you are in when you log in to the system). The IRAF home directory doesn't need to be in your account's root directory, nor does it need to be called `iraf`, but you should *not* put it on a scratch disk that is periodically erased.

If you call your root IRAF directory "`iraf`", you can set up IRAF as follows:

Under Unix:

```
% mkdir iraf
% cd iraf
% setenv iraf /usr/stsci/iraf/
% source $iraf/unix/hlib/irafuser.csh
% mkiraf
```

Can be placed in .login file →

← The directory name is site-dependent—check with your system staff

Under VMS:

```
$ CREATE/DIR [.IRAF]
$ SET DEFAULT [.IRAF]
$ IRAF
$ MKIRAF
```

Can be placed in LOGIN.COM file →

The **mkiraf** command initializes IRAF by creating a `login.cl` file and a subdirectory called `uparm`. After typing the **mkiraf** command, you will see the following:

```
% mkiraf
-- creating a new uparm directory
Terminal types: gterm=ttysw+graphics,vt640...
Enter terminal type:
```

Enter the type of terminal or workstation you will most often use with IRAF.¹ Generic terminal types that will work for most users are:

- `vt100` for most terminals.
- `xtermjhs` for most workstations running under X-Windows.

`xgterm` for sites that have installed X11 IRAF and IRAF v2.10.3 BETA or later.



You can change your terminal type at any time by typing `set term=new_type` during an IRAF session. You can also change your default type by editing the appropriate line in your `login.cl` file.

After you enter your terminal type, you will see the following output before getting your regular prompt:

```
A new LOGIN.CL file has been created in the current ...
You may wish to review and edit this file to change ...
```

The `login.cl` file is the *startup file* used by the IRAF command language (CL). It is similar to the `LOGIN.COM` file used by VMS or the `.login` file used by Unix. Whenever IRAF starts, it looks at the `login.cl` file. You can edit this file to customize your IRAF environment. In fact, you should look at it to make sure that everything in it is correct. In particular, there is a line starting with `set home =` that tells IRAF where to find your IRAF home directory. You should verify that this statement does, in fact, point to your IRAF directory. If you will be working with standard IRAF format images you should also insert a line saying `set imdir = "HDR$"`. The `imdir` setting is ignored when working with GEIS format images.

The `uparm` directory will contain your own copies of IRAF task parameters. This directory allows you to customize your IRAF

1. Users at STScI should consult the *STScI Site Guide for IRAF and STSDAS*.

environment by setting certain parameter values as defaults. Once you set up IRAF, you should rarely need to do it again, except when updated version of IRAF are installed.

A.1.2 Starting and Stopping an IRAF Session

To start an IRAF session:

1. Move to your IRAF home directory.
2. Type `c1`.

IRAF starts by displaying several lines of introductory text and then puts a prompt at the bottom of the screen. Figure A.1 is a sample IRAF startup screen.

Figure A.1:IRAF Startup Screen

```

NOAO Sun/IRAF Revision 2.11 Fri Aug 15 15:34:46 MST 1997
This is the EXPORT version of Sun/IRAF V2.11 for SunOS 4 and Solaris 2.5

Welcome to IRAF. To list the available commands, type ? or ??. To get
detailed information about a command, type 'help command'. To run a
command or load a package, type its name. Type 'bye' to exit a
package, or 'logout' to get out of the CL. Type 'news' to find out
what is new in the version of the system you are using. The following
commands or packages are currently defined:

  apropos      euv.          local.          spptools.
  ared.        fitsutil.    mem0.          stlocal.
  aspec.       focas.       newimred.      stsdas.
  c128.        ftools.     noao.          system.
  color.       hst_pipeline. obsolete.       tables.
  ctio.        images.     plot.          utilities.
  dataio.      imcnv.      proto.         vol.
  dbms.        language.   rvsao.         xray.
  digiphotx.   lists.          softools.

c1>

```

Startup Messages
Change from Day
to Day

Available Packages
and Tasks

To quit an IRAF session:

1. Type `logout`.

A.2 IRAF Basics

This section describes basic IRAF techniques such as:

- Loading packages (below).
- Running tasks and commands.
- Getting online help.
- Viewing and setting parameters (see appendix section A.2.4).

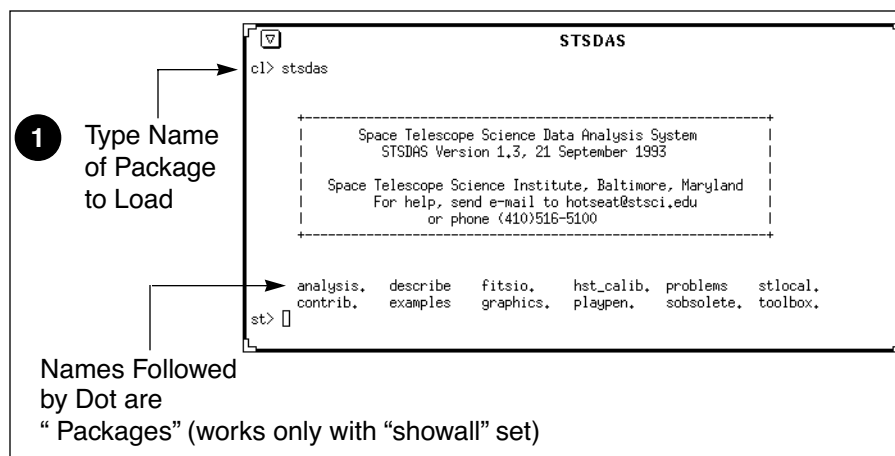
- Setting and using environment variables (see section A.2.5).
- File management
- Troubleshooting

A.2.1 Loading Packages

In IRAF jargon, an application is called a *task* and logically related tasks are grouped together in a *package*. Before you can use a task, you must load the package containing that task. To load a package, type the name of the package. The prompt will then change to the first two letters of the package name, and the screen will display the names of all the newly available tasks and subpackages. Even though the prompt has changed, previously loaded packages remain loaded, and all their tasks remain available.

Note that the standard way to specify a path through the IRAF package hierarchy to a task in a particular subpackage is to separate the package names with periods (e.g., **stdas.hst_calib.foc.focgeom.newgeom**).

Figure A.2:Loading Packages



Some helpful commands for managing packages are:

- ? - Lists tasks in the most recently-loaded package.
- ?? - Lists all tasks loaded, regardless of package.
- package - Lists names of all loaded packages.
- bye - Exits the current package.

A.2.2 Running Tasks

This section explains how to run tasks, background tasks, and system-level commands, and how to use piping and redirection.

Running a Task

The simplest way to run a task is to type its name or any unambiguous abbreviation of it. The task will then prompt you for the values of any required *parameters*, such as the names of input files. Alternatively, you can specify the values for the required *parameters* on the command line when you run the task. For example, if you want the task `imheader` to print header information on the file `myfile.hhh`, you can type

```
st> imhead myfile.hhh
```



IRAF does not require you to type the complete command name—only enough of it to make it unique. For example, `dir` is sufficient for `directory`.

Escaping System-Level Commands

To run an operating system-level command (i.e., Unix or VMS commands) from within the IRAF CL, precede the command with an exclamation point (!). This procedure is called *escaping* the command. For example:

```
st> !system_command
```

Piping and Redirection

You can run tasks in sequence if you desire, with the output of one task being used as the input for another. This procedure, called *piping*, and is done by separating commands with a vertical bar (|), using the following syntax:

```
st> task1 filename | task2
```

For example, if a particular task prints a large volume of textual output to the screen, you will often want to pipe it to `page`, which allows you to read the output one page at a time:

```
st> task1 filename | page
```

You can also redirect output from any task or command to a file by using the greater-than symbol (>) as follows:

```
st> command > outputfile
```

Background Tasks

To run a task as a background job, freeing your workstation window for other work, add an ampersand (&) to the end of the command line, like this:

```
st> taskname &
```

A.2.3 Getting Help

This section describes:

- How to use IRAF's on-line help facility.
- How to find a task that does what you want (see "Finding Tasks" on page A-8).

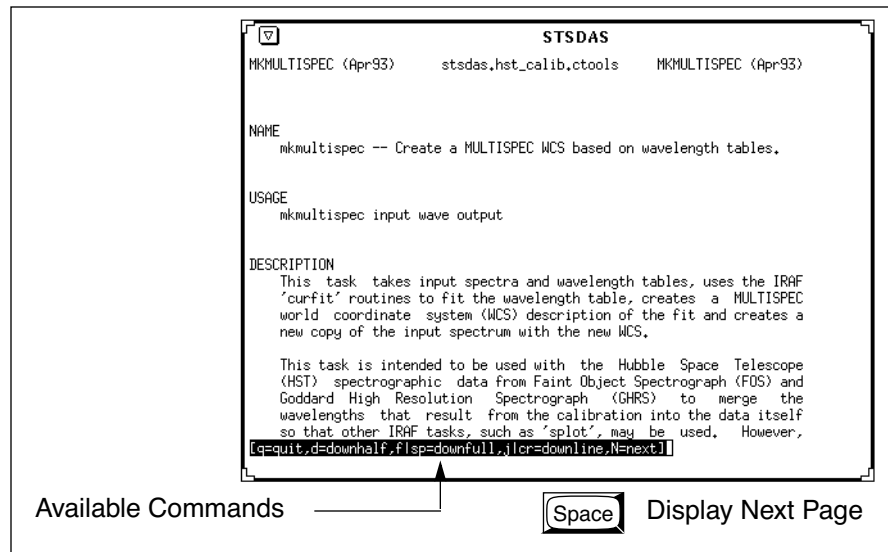
On-Line Help

You can get on-line help with any IRAF task or package by using the **help** command,² which takes as an argument the task or package name about which you want help. Wildcards are supported. For example, to display the on-line help for the STSDAS **mkmultispec** task, you would type:

```
fi> help mkmultispec
```

2. There is an optional *paging* front-end for help called **phelp**. For more information, type **help phelp** from within IRAF.

Figure A.3:Displaying On-line Help



Two STSDAS tasks that display only certain sections of the help file are also available:

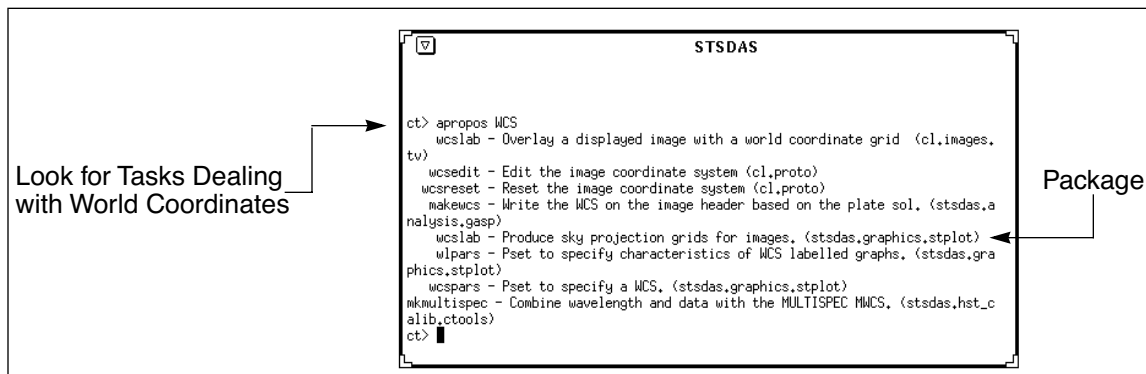
- **examples** - Displays only the examples for a task.
- **describe** - Displays only the description of the task.

Typing `help package` will produce one-line descriptions of each task in the package.

Finding Tasks

There are several ways to find a task that does what you need:

- Use `help package` to search through the IRAF/STSDAS package structure.
- Use the **apropos** task as shown in figure A.4 to search the online help database. This task looks through a list of IRAF and STSDAS package menus to find tasks that match a specified keyword. Note that the name of the package containing the task is shown in parentheses.
- Ask more experienced user, who can usually point you in the right direction.

Figure A.4:The **apropos** task Using apropos

A.2.4 Setting Parameters

Parameters specify the input information for IRAF tasks. They can be the names of input or output files, particular pixel numbers, keyword settings, or many other types of information that control the behavior of the task.

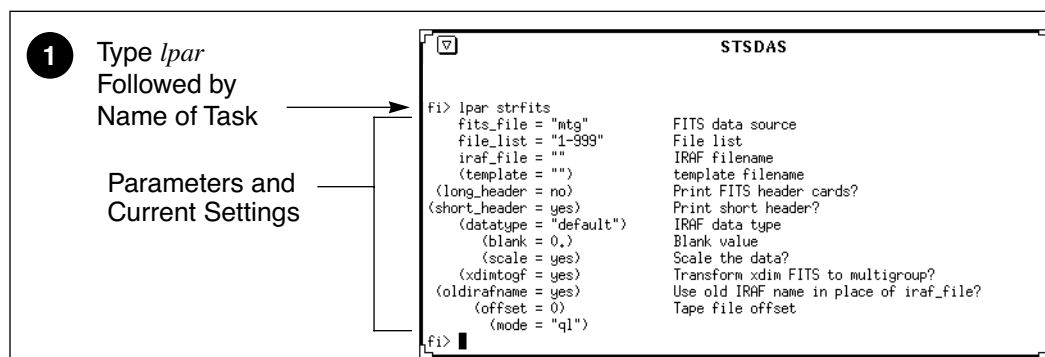
The two most useful commands for handling parameters are:

- **lparam** to display the current parameter settings (often abbreviated **lpar**).
- **eparam** to edit parameters (often abbreviated **epar**).

Viewing Parameters with lparam

The **lpar** command lists the current parameter settings for a given task (figure A.5).

Figure A.5:Displaying Parameter Settings with lpar

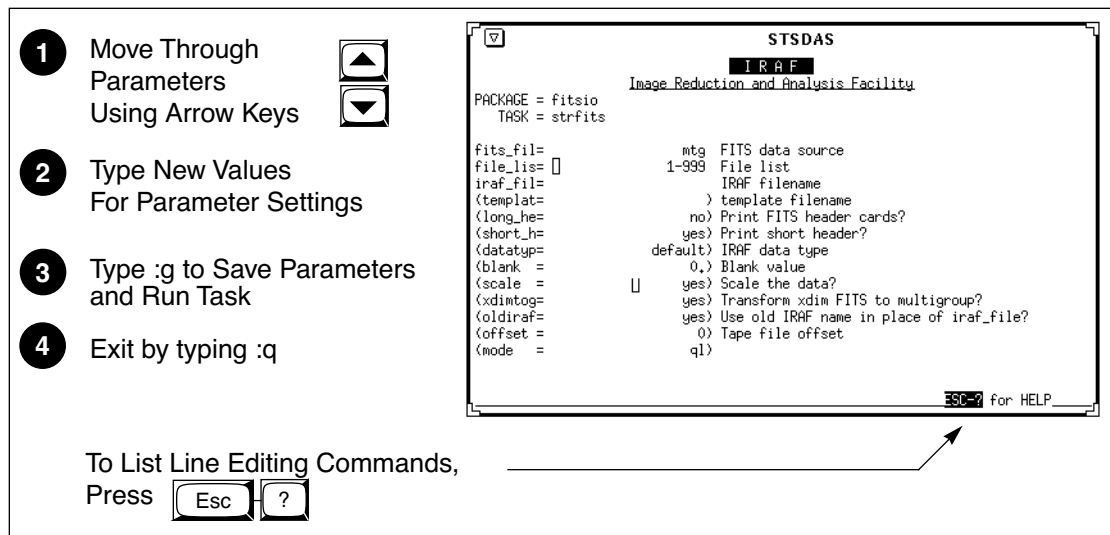


Setting parameters with eparam

The **epar** command is an interactive parameter set editor. It displays all of the parameters and their current settings on the screen. You can move around the screen using the arrow keys (also called *cursor* keys) and type

new settings for any parameters you wish to change. Figure A.6 shows a sample of the **epar** editor at work (invoked by typing `epar strfits`).

Figure A.6:Editing Parameters with `epar`



Parameter Data Types—What to Specify

Parameters are either *required* or *hidden*, and each parameter expects information of a certain *type*. Usually, the first parameter is required, and very often it expects a file name. Parameters are described in the online help for each task [include reference to help]. Hidden parameters, shown in parentheses in the online help and the **lpar** and **epar** listings, need not be specified at each execution because their default values frequently suffice.



Wise IRAF users will check the values of hidden parameters, as they often govern important aspects of a task's behavior.

If you specify the wrong type of information for a parameter, **epar** will usually display an error message saying something like “Parameter Value is Out of Range.” The message is displayed when you move to another parameter or if you press `[Return]`. Table A.1 lists the different parameter types.

Table A.1: Parameter Data Types

Type	Description
File Name	Full name of the file. Wild card characters (* and ?) are often allowed. Some tasks allow you to use special features when specifying file names, including “@” lists, IRAF networking syntax, and image section or group syntax. (See “File Management” below).
Integer	Whole number. Often the task will specify minimum or maximum values (see the help pages).
Real	Floating point numbers, can be expressed in exponential notation. Often will have minimum and maximum values.
Boolean	Logical “yes” or “no” values.
String	Any characters. Sometimes file names are specified as string.
Pset	Parameter set.

Restoring Parameter Default Values

Occasionally, IRAF (or you) will get confused by your parameter values. To alleviate this confusion, you can restore the default parameters with the **unlearn** command. You can use **unlearn** on either a task or on an entire package.



*The **unlearn** command generally will restore the parameters to reasonable values, a big help if you are no longer sure which parameter values you have changed in a complicated task.*

A.2.5 Setting Environment Variables

IRAF uses *environment variables* to define which devices are used for certain operations. For example, your terminal type, default printer, and the disk and directory used for storing images are all defined through environment variables. Environment variables are set using the **set** command and are displayed using the **show** command. Table A.2 lists some of the environment variables that you might want to customize.

Table A.2: Environment Variables

Variable	Description	Example of Setting
printer	Default printer for text	set printer = lp2
terminal	Terminal type	set term = xterm
stdplot	Default printer for all graphics output	set stdplot = ps2
stdimage	Default terminal display setting for image output (most users will want this set to either imt512 or imt800)	set stdimage = imt800
stdgraph	Default graphics device	set stdgraph = xterm
clobber	Allow or prevent overwriting of files	set clobber = yes
imtype	Default image type for output images. “imh” is original IRAF format, “hhh” is STSDAS GEIS format.	set imtype = “hhh”



If you are working with GEIS files, you should set imtype to “hhh”. If you are working with STIS and NICMOS data in FITS files, you can set imtype to “fits”

You can set your environment variables automatically each time you login to IRAF by adding the appropriate commands to your `login.cl` file. Use your favorite text editor to specify each variable on its own line. The **show** command with no arguments prints the names and current values of all environment variables.

A.2.6 File Management

This section describes:

- File formats commonly used with STSDAS and IRAF.
- Specification of file names.
- Navigation through directories.

File Formats

IRAF recognizes a number of different file structures. Among them are the standard HST file formats known as GEIS and FITS (see chapter 2 of the HST Introduction), both of which differ from the original IRAF format (OIF). GEIS is closer to OIF, in that two files are *always* used together as a pair:

- A *header file*, which consists of descriptive information. IRAF header files are identified by the suffix `.imh`. GEIS header files are in ASCII text format and are identified by the suffix `.hhh` or another suffix ending in “h”, such as `.c0h` or `.q1h`.
- A *binary data file*,³ consisting of pixel information. IRAF data file names end with a `.pix` suffix. STSDAS data files end with an suffix of `.hhd` or another suffix that ends with “d”, such as `.c0d` or `.q0d`.

STSDAS always expects both component files of a GEIS image to be kept together in the same directory. A single FITS file contains both the header information and the data.



When working with IRAF or STSDAS images, you need only specify the header file name—the tasks will automatically use the binary data file when necessary.

File Specification

Most tasks in IRAF and STSDAS operate on files and expect you to specify a file name for one or more parameters. Several types of special syntax can be used with certain tasks when specifying file names. These syntax features include:

- **Wild card characters**, often called *templates*, which are used to specify multiple files using pattern matching techniques. The wild cards are:
 - * Matches any number of characters, e.g.: `z*.c0h`
 - ? Matches any single character, e.g.: `z01x23x.c?h`

3. The binary data file format is host-dependent and may require translation before it can be moved to a computer using a different architecture.



When using wildcards with image-processing tasks, be sure to exclude the binary pixel files by ending your file name specification with an “h”, for example: `y.*?h`*

- **List files**, often called *@-files*, which are ASCII file that contain lists of file names, one per line. If your task supports the list file feature, you would type the name of your list file, preceded by the “@” character. For example: `@files.txt`
- **Image section** specification. Tasks that work with image data will often let you specify that you want to work on only a small area of the image rather than the entire image. To extract a particular image section, specify each axis range in square brackets, for example: `image.hhh[10:200,20:200]`
- **IRAF networking** specification. IRAF is capable of reading and writing files to and from remote systems on a network. This feature is often used with tasks in the **fitsio** and **convfile** packages, or with image display tasks. The *STSDAS Users Guide* and the online help (type `help networking`) describe how to enable this feature. To specify that you want to use the IRAF networking feature, type the remote host name followed by an exclamation point (!), followed by the file or device name. For example: `ra!mta`.

Directory Navigation

To navigate through directories, you can use the following commands:

- **path** or **pwd** - Lists the current working directory.
- **cd** *directory* - Move to the named directory.

A.2.7 Troubleshooting

There are a couple of easy things you can do to make sure that you don’t have a simple memory or parameter conflict—common causes of problems.

- Look at the parameter settings and make sure that you have specified reasonable values for every parameter.

- When you run an IRAF task for the first time in a session, IRAF stores the executable file in its *process cache*. If IRAF appears not to be running your tasks properly, you may need to use the **flprcache** command to clear the process cache. To do this type: `flpr`. Sometimes you will need to execute this command twice in succession.
- Occasionally, you may need to logout of the CL, restart IRAF, and try your command again.

If you still have a problem, contact the STScI Help Desk at help@stsci.edu

A.3 Getting IRAF and STSDAS

Both IRAF and STSDAS are provided free of charge to the astronomical community. You must have IRAF to run STSDAS. Detailed information about installing and retrieving STSDAS is found in the *STSDAS Site Manager's Installation Guide and Reference*. If you have any problems getting and installing STSDAS, TABLES, or any other packages or data described in this handbook, please contact the Help Desk by sending e-mail to: help@stsci.edu.

A complete description of how to install the **synphot** data files is provided in section A.3.2.

A.3.1 Retrieving the IRAF and STSDAS Software

There are three ways to get the software:

- Use the World Wide Web.
- Use anonymous FTP.
- Request a tape.

World Wide Web

The STSDAS World Wide Web page:

<http://stsdas.stsci.edu/STSDAS.html>

provides links and instructions for downloading the appropriate files to your local system or to display the software directory, from which you can select the series of smaller files.

Anonymous FTP

- **IRAF:** `iraf.noao.edu` (140.252.1.1)
- **STSDAS:** `ftp.stsci.edu` (130.167.1.2)

There are two points to remember when using FTP to retrieve STSDAS:

- You must retrieve and install the TABLES package before STSDAS.
- You should retrieve the README file from the directory `/software/stsdas/v2.0` and read it to find out which files you should retrieve.



You must have IRAF installed on your system to install TABLES and STSDAS. When you retrieve STSDAS, you must also retrieve the TABLES package, and TABLES must be installed first.

Instructions for installing STSDAS are available in the `doc` subdirectory of the directory where you find STSDAS. The complete instructions for installing STSDAS, TABLES, and all of the supporting software and reference files (including instrument reference files and the **synphot** dataset) are found in the *STSDAS Site Manager's Installation Guide and Reference*.

Registration

The software can also be registered and requested using on-line forms available through World Wide Web at the following URL:

<http://stsdas.stsci.edu/RegistForm.html>

When you request the STSDAS software, you can also ask for the appropriate version of IRAF, which will be requested for you— simply check the appropriate box on the form under “Do You Already Have IRAF Installed?” If you prefer to request the IRAF software independent of STSDAS, you can do so by sending e-mail to: iraf@iraf.noao.edu

A.3.2 Getting the Synphot Database

This manual sometimes refers to the **synphot** dataset, which must be available in order to run tasks in the STSDAS **synphot** package. These data files are not included with the STSDAS software and must be retrieved independently. To do this, you need to retrieve a series of compressed tar files from the STScI FTP site (<ftp.stsci.edu>) in the directory `software/stsdas/refdata/synphot`. After uncompressing and extracting the tar files (see below), you need to unpack the FITS files as described below.

The synthetic photometry data are read in similar way as the instrument datasets, using the script `unpack.c1` provided in the top directory. This script is run within IRAF to convert data from FITS format into the format used by the **synphot** task. This script assumes you have the logical

`crrefer` set up in your `extern.pkg` file (which is in the directory `$iraf/unix/hlib` (Unix) or `$iraf/vms/hlib` (VMS)) or have it set up in your session. You do this by placing the command below in `extern.pkg` or by typing it on the command line:

```
set crrefer = "/node/partition/stddata/synphot/"
```

Figure A.7 shows how to convert the files.

Figure A.7: Unpacking Synthetic Photometry Files

```
% cl
cl> cd /node/partition/stddata/synphot
cl> set crrefer = "/node/partition/stddata/synphot/"
cl> task $unpack = unpack.cl
cl> tables
ta> fitsio
fi> unpack
```

Just in case...

The "\$" is used because the task has no parameter file



Note that all three synphot files must be unloaded for the script to complete successfully.

A.3.3 Extracting the synphot Unix Tar Files

If you retrieved the **synphot** database as compressed tar files, you will need to copy them to an appropriate subdirectory and then expand and unpack the files. The `tar` and `compress` utilities that do this are commonly available on most Unix systems, but are not standard in the VMS environment. The examples shown below reflect Unix usage. If you are on a VMS system, you should consult with your systems support staff regarding the availability and usage of these commands. To process the files on a Unix system:

1. Get the compressed tar file that you want, as described in previous sections.
2. Make an appropriate subdirectory using the `mkdir` command.
3. Pipe the compressed tar file through the `uncompress` and `tar` files to expand and unpack the file.

The following example shows how to do this. The example assumes that you are putting the files in a subdirectory under `/usr/iraf/stdata` (note that the name of your file here is assumed to be `XXX.tar.Z`).

```
% pwd
/usr/iraf/stdata
% mkdir XXX
% mv XXX.tar.Z XXX/
% cd XXX
% cat XXX.tar.Z | uncompress | tar -xf -
```

APPENDIX B:

B.1 Rootnames / B-2

B.3 Associations / B-5

Diagram illustrating the structure of the filename `ipppssoot_sfx.fits`:

- Rootname**: `ipppssoot`
- Suffix (Data Type)**: `_sfx`
- Format (FITS)**: `.fits`

- For example, an FOC data file named `x3l80101t_d0f.fits` is a FITS file belong to the dataset with rootname `x3l80101t`, and its suffix `d0f` indicates that it contains raw science data.

In order to use IRAF/STSDAS tasks to work with data from instruments other than NICMOS and STIS, you will want to convert these FITS files into GEIS format. See section 2.2 in the HST Introduction for instructions on how to convert FITS files to GEIS files using **strfits**. Like FITS files, the names of GEIS files also derive from a file's rootname and suffix, and they look like this:

ippssoot.sfx

Generally the suffixes of GEIS files end either in “d”, indicating a binary data file, or “h”, indicating an ASCII header file. The two GEIS files `x3l80101t_d0h` and `x3l80101t_d0d` together contain the same information as the single FITS file `x3l80101t_d0f.fits`.



The identifier referred to here as a “suffix” has often been called an “extension” in the past. However, the individual pieces of FITS files are also known as “extensions” (see section 2.2.1 in the HST Introduction). For clarity, this handbook will use the term “extension” when referring to a component of a FITS file and the term “suffix” when referring to the three character identifier in a filename.

B.1 Rootnames

Rootnames of HST data files follow the naming convention defined in table B.1, which expands on the previous convention as follows: an initial “N” indicates a NICMOS exposure, an initial “O” indicates a STIS exposure, and the rootnames of files containing association products (see below) end in a number (0-8).

Table B.1: IPPPSSOOT Root File Names

Character	Meaning
I	Instrument used, will be one of: <i>E</i> - Engineering data <i>F</i> - Fine Guidance Sensors <i>N</i> - Near Infrared Camera and Multi-Object Spectrograph <i>O</i> - Space Telescope Imaging Spectrograph <i>S</i> - Engineering subset data <i>T</i> - Guide star position data <i>U</i> - Wide Field/Planetary Camera-2 <i>V</i> - High Speed Photometer <i>W</i> - Wide Field/Planetary Camera <i>X</i> - Faint Object Camera <i>Y</i> - Faint Object Spectrograph <i>Z</i> - Goddard High Resolution Spectrograph
PPP	Program ID; can be any combination of letters or numbers (46,656 combinations possible). There is a unique association between program ID and proposal ID.
SS	Observation set ID; any combination of letters or numbers (1,296 possible combinations).
OO	Observation ID; any combination of letters or numbers (1,296 possible combinations).
T	Source of transmission or association product number <i>M</i> - Merged real time and tape recorded <i>N</i> - Retransmitted merged real time and tape recorded <i>O</i> - Retransmitted real time (letter 'O') <i>P</i> - Retransmitted tape recorded <i>R</i> - Real time (not recorded) <i>T</i> - Tape recorded <i>0</i> - Primary association product (number zero) <i>1-8</i> - NICMOS background association product

B.2 Suffixes of Files Common to all Instruments

The three-character suffix of a data file (e.g., d0h) identifies the type of data that a file contains. Because the meanings of these suffixes change from instrument to instrument, please refer to the appropriate instrument-specific Data Structures chapter for their definitions. Several types of file suffixes are, however, common to all instruments.

OMS Files

Observatory Monitoring System (OMS) files, having suffixes `cm*` or `ji*`, contain Observation Logs describing how the HST spacecraft behaved during a given observation. OMS headers, which you can read with the IRAF task **imheader** (see section 2.3.3 in the HST Introduction),

are divided into groups of keywords that deal with particular topics such as SPACECRAFT DATA, BACKGROUND LIGHT, POINTING CONTROL DATA, and LINE OF SIGHT JITTER SUMMARY. The headers themselves provide short descriptions of each keyword. OMS tables and images record spacecraft pointing information as a function of time. For more information on OMS files, you can consult Appendix C or the STScI Observation Logs WWW pages at:

http://www.stsci.edu/ftp/instrument_news/Observatory/obslog/OL_1.html

PDQ Files

The suffix `pdq` denotes Post Observation Summary and Data Quality Comment files—*PDQ files*—which contain predicted as well as actual observation parameters extracted from the standard header and science headers. These files may also contain comments on any obvious features in the spectrum or image, as noted in the OPUS data assessment, or automatically extracted information about problems or oddities encountered during the observation or data processing. These comments may include correction to the keywords automatically placed in the OMS files.

OCX Files

The suffix `ocx` denotes Observer Comment Files—*OCX files*—which are produced by STScI personnel to document the results of real-time commanding or monitoring of the observation, along with keywords and comments. Prior to April 17, 1992, OCX files were not always archived separately and, in some cases, were prepended to the trailer file.

After early February 1995, OCX files were produced only when an observation was used to locate the target for an Interactive Target Acquisition. At this time, mission and spacecraft information were moved to the PDQ reports and the Observation Logs (OMS jitter image and jitter table).

Trailer Files

Trailer files (suffix `trl`) are FITS ASCII tables that log the processing of your data by the OPUS pipeline.



Note that trailer files are formatted with 132 columns

B.3 Associations

The STIS and NICMOS calibration pipelines sometimes produce single calibrated images from *associations* of many exposures. These associations allow HST pipeline processing to proceed further than it has in the past. For example, a NICMOS observer might specify a dithering pattern in a Phase II proposal. NICMOS would then take several exposures at offset positions, and the pipeline would combine them into a single mosaic. In this case, the original set of exposures constitutes the association, and the mosaic is the *association product*. Similarly, a STIS observer might specify a CR-SPLIT sequence in a Phase II proposal. STIS would gather several exposures at the same pointing, and the STIS pipeline would process this association of exposures into single image, free of cosmic rays, that would be the association product.

When you search the Archive with StarView for observations involving associations of exposures, your search will identify the final association product. The rootnames of association products always end in zero (see figure B.1 above.) If you request both Calibrated and Uncalibrated data from the Archive, you will receive both the association product and the exposures that went into making it. The corresponding association table, located in the file with suffix `asn` and the same rootname as the association product, lists the exposures belonging to the association. You can read this file using the STSDAS `tprint` or `tread` tasks (see table 3.1 in the HST Introduction). The exposure IDs in the association table share the same `ippss` sequence as the association rootname, followed by a base 36 number `nn` (`n = 0-9,A-Z`) that uniquely identifies each exposure, and a character `t` that denotes the data transmission mode (see figure B.1).

In practice, STIS and NICMOS store the exposures belonging to associations differently. The exposures belonging to a STIS association all reside in the same file, while the exposures belonging to a NICMOS association reside in separate datasets. See the relevant Data Structures chapters for more details.

Information on the exposures belonging to an association is also available through StarView (see chapter 1 of the HST Introduction). From the <Welcome> Screen, click on **[HST Instrument Searches]** to get the <HST Instruments> screen, and then click on the **[Associations]** button for the instrument of interest. You can then search for the various exposures belonging to an association by entering the rootname of the association in the Association ID field and clicking on **[Begin Search]**. An Association Results Screen will display the results of the search, which you can step through using the **[Step Forward]** button. Figure B.1 below gives an example of a NICMOS Association Results Screen. Note the differences between the association rootname and coordinates and those of the individual exposure.

Figure B.1:Association Results Screen from StarView

< NICMOS Association Results >			
File	Searches	Constraint	View Retrieve Customize Options Comments Help
Association ID:	N3S211010	Proposal ID:	862
Pattern:	NONE	PI (last name):	WAYNE BAGGETT
Member Name:	N3S211010	Target Name:	TARGET1
Member Type:	PROD-TARG	Start Time:	03/29/97 06:16:52
RA (RA ,2000):	17 59 09.185	Dec (Dec ,2000):	-61 35 02.000
Camera:	2	Orient:	50.364
Exp Len:	27.424	Numpos:	0
Filter:	F110W	Offset:	
Mode:	ACCUM	Dither Size:	0.000
Samp Seq:		Chop Size:	0.000
Aperture:	NIC2	Nread:	4
		Nsamp:	1
		Readout:	FAST
EXPOSURES			
Dataset Name:	N3S21106R	Position #:	
Exp. Start:		Exp. Flag:	
PAM Focus:	4.223		
RA (RA ,2000):	18 00 00.000	Dec (Dec ,2000):	-61 30 00.000
<div> <div>Step Forward</div> <div>Step Back</div> <div>Mark Dataset</div> <div>Retrieve Marked Data</div> <div>Scan Forward</div> <div>Scan Back</div> <div>Unmark Data</div> <div>Write Result to File</div> <div>Edit Search Constraints</div> <div>Mark All</div> <div>View Result as Table</div> <div>Record 1 of 1 (in progress)</div> <div>Unmark All</div> <div>Strategy</div> <div>Preview</div> <div>Overlay</div> <div>Exit Screen ^Z</div> </div>			
MESSAGE: More records available. Use record controls to view search results			

Observation Logs

In this appendix . . .

C.1 Observation Log Files / C-1 C.2 Retrieving Observation Logs / C-9 C.3 Using Observation Logs / C-10

This Appendix describes the *Observation Log Files*, also known as OMS or *jitter* files. These files record pointing, jitter, and other Pointing Control System (PCS) data taken during an HST observation. You can use them to assess the behavior of the HST spacecraft during your observation, and in particular, to evaluate the jitter of the spacecraft while it was taking data. Here we describe the contents and structure of the observation log files, how to retrieve them from the Archive, and how to work with the data they contain.

C.1 Observation Log Files

Observation log files associated with each HST dataset contain pointing and specialized engineering data taken during the observation. These data files are produced by the Observatory Monitoring System (OMS), an automated software system that interrogates the HST engineering telemetry and correlates the time-tagged engineering stream with HST's Science Mission Schedule (SMS), the seven-day command and event list that drives all spacecraft activities. This system reports the status of the instruments and observatory and flags discrepancies between planned and executed actions. OMS provides observers with information about guide star acquisition, pointing, and tracking that is not normally provided in the science headers.

The observation log files share the same rootname as the observation they are associated with, except for the final character, which for observation log files is always a "j" (see appendix B for more on the names of HST data files). When OMS was installed in October 1994, it initially generated files with the suffixes `cmh`, `cmj`, `cmi`, which contained header information, high time resolution pointing data, and three-second average pointing data, respectively (see table C.1). OMS observation logs changed to the `jih/jid/jif` image format after August 1995, at which time the `cmi` table was renamed `jit` to keep the naming convention consistent. In the OMS version of August 1995, `cmj` tables were replaced with a jitter image, which is a two-dimensional histogram of jitter excursions during the observation. The suffixes of the GEIS jitter image are `jih` for the header and `jid` for the image data. The `jit` table accompanies the jitter image. The header file of the image replaces the `cmh` file but includes the same information with the addition of some image-related keywords.



A detailed description of the observation log files can be found on-line:
http://www.stsci.edu/instruments/observatory/obslog/OL_1.html.

Table C.1: OMS Observation Log Files

Suffix	Contents
<i>October 1994 to August 1995</i>	
<code>cmh</code>	OMS header
<code>cmj</code>	High time resolution (IRAF table)
<code>cmi</code>	Three-second averages (IRAF table)
<code>_cmh.fits</code>	Archived FITS file of <code>cmh</code>
<code>_cmj.fits</code>	Archived FITS file of <code>cmj</code>
<code>_cmi.fits</code>	Archived FITS file of <code>cmi</code>
<i>August 1995 to February 1997</i>	
<code>jih/jid</code>	Two-dimensional histogram and header (GEIS)
<code>jit</code>	Three-second averages (IRAF table) ¹
<code>_jif.fits</code>	Archived FITS file which bundles the <code>jih/jid</code> files.
<code>_jit.fits</code>	Archived FITS file of <code>jit</code> .
<i>February 1997 onward</i>	
<code>_jif.fits</code>	Two-dimensional histogram (FITS)
<code>_jit.fits</code>	Three-second averages table (FITS)

1. After May 11, 1995, the `jit` tables for exposures shorter than 6 seconds contain higher-resolution, one-second average pointing data.



Pointing and tracking information prior to October 1994 is not routinely available. Interested observers with data from this epoch, can send E-mail to help@stsci.edu.

C.1.1 Observation Log File Contents (October 1994 version)

Observation logs created between October 1994 and August 1995 contain:

- *rootnamej.cmh*: This ASCII header file contains the time interval, the rootname, averages of the pointing and spacecraft jitter, the guiding mode, guide star information, and alert or failure keywords. Figure C.1 shows a representative observation log header file.
- *rootnamej.cmj*: This table presents the data at the highest time resolution for the telemetry mode in use. It contains the reconstructed pointing, guide star coordinates, derived jitter at the instrument aperture, and guiding-related flags. The intent is: (1) to provide high-time resolution jitter data for deconvolution or for assessing small aperture pointing stability, and (2) to display the slew and tracking anomaly flags with the highest resolution. Table C.2 lists the table column heading, units and a brief definition.
- *rootnamej.cmi*: This table contains data that were averaged over three-second intervals. It includes the same information as the .cmj table and also includes orbital data (e.g., latitude, longitude, limb angle, magnetic field values, etc.) and instrument-specific items. It is best suited for a quick-look assessment of pointing stability and for studying trends in telescope or instrument performance with orbital environment. Table C.3 lists the table column heading, units and a brief definition.
- *rootnamej_cmi/j/h.fits*: The above three GEIS files are actually archived as FITS files. They may be worked with as such, or run through the STSDAS task **strfits**, to convert them.
- Observation Log File Contents (August 1995 version)
- The contents of observation log files created between August 1995 and February 1997 are as follows:

- *rootnamej.jih*: This GEIS header file, the analog to the *cmh* file, contains the time interval, the rootname, averages of the pointing and spacecraft jitter, the guiding mode, guide star information, and alert or failure keywords. Figure C.1 shows a representative observation log header file.
- *rootnamej.jid*: This GEIS image—a significant enhancement of the old *cmj* file—presents a two-dimensional histogram of the pointing fluctuations during the observation. You can display it to visualize the spacecraft stability during your observation, and is information for deconvolutions and PSF analyses.
- *rootnamej.jit*: This table, the analog to the *cmi* table, contains data that were averaged over three-second intervals. Its content is identical (see table C.3).
- *rootnamej_jif.fits*: FITS file that is actually the de-archived product. This file can be converted to the *jih/jid* GEIS file via the **strfits** routine.
- *rootnamej_jit.fits*: The de-archived FITS file corresponding to the *jit* IRAF table. It can be converted via **strfits**.

C.1.2 Observation Log File Contents (February 1997 version)

The contents of observation log files created since February 1997 are as follows:

- *rootnamej_jif.fits*: The de-archived FITS file. Unlike the previous OMS epoch, this FITS file does not bundle a GEIS file and cannot be converted with **strfits**. This was done to more closely correlate the observation log files with the STIS and NICMOS FITS files with extensions and associations. OMS will normally put all associated observation logs into a single file, to correspond to the associated science exposures. However, if even one science exposure is orphaned (not associated) then an individual observation log FITS file will be produced for every exposure in that association. For a description of STIS and NICMOS association files, see appendix B. All of the information contained in the old *cmh/jih* ASCII header is now available as keywords in the FITS files.
- *rootnamej_jit.fits*: The FITS file containing the table information. The comments for the *_jif* file apply here as well.

Table C.2: Contents of .cmj Table

Parameter	Units	Description
seconds	seconds	Time since window start
V2 dom	arcseconds	Dominant FGS V2 coordinate
V3 dom	arcseconds	Dominant FGS V3 coordinate
V2 roll	arcseconds	Roll FGS V2 coordinate
V3 roll	arcseconds	Roll FGS V3 coordinate
SI V2	arcseconds	Jitter at aperture reference
SI V3	arcseconds	Jitter at aperture reference
RA	degrees	Right ascension of aperture reference
DEC	degrees	Declination of aperture reference
Roll	degrees	Angle between North and +V3
DayNight	0,1 flag	Day (0) or night (1)
Recenter	0,1 flag	Recentering status
TakeData	0,1 flag	Vehicle guiding status
SlewFlag	0,1 flag	Vehicle slewing status

Figure C.1: A Representative .jih or .cmh Header

```

dSIMPLE = F / data conforms to FITS standard !
BITPIX = 32 / bits per data value !
DATATYPE= 'INTEGER*4' / datatype of the group array !
NAXIS = 2 / number of data axes !
NAXIS1 = 64 / length of the 1st data axis !
NAXIS2 = 64 / length of the 2nd data axis !
GROUPS = T / image is in group format !
GCOUNT = 1 / number of groups !
PCOUNT = 0 / number of parameters !
PSIZE = 0 / bits in the parameter block !
OMS_VER = '16.2C' / OMS version used to process this observation
PROCTIME= '1994.133:06:24:18.35' / date-time OMS processed observation
/ date-times format (yyyy.ddd:hh:mm:ss.ss)

/ IMAGE PARAMETERS
CRVAL1 = 0.0 / right ascension of zero-jitter pixel (deg)
CRVAL2 = 0.0 / declination of zero-jitter pixel (deg)
CRPIX1 = 32 / x-coordinate of zero-jitter pixel
CRPIX2 = 32 / y-coordinate of zero-jitter pixel
CTYPE1 = 'RA---TAN' / first coordinate type
CTYPE2 = 'DEC--TAN' / second coordinate type
CD1_1 = 0.0 / partial of ra w.r.t. x (deg/pixel)
CD1_2 = 0.0 / partial of ra w.r.t. y (deg/pixel)
CD2_1 = 0.0 / partial of dec w.r.t. x (deg/pixel)
CD2_2 = 0.0 / partial of dec w.r.t. y (deg/pixel)
COORDSYS= 'WFPC2' / image coordinate system
XPIXINC = 2.0 / plate scale along x (mas per pixel)
YPIXINC = 2.0 / plate scale along y (mas per pixel)
PARITY = -1 / parity between V2V3 frame and image frame
BETA1 = 134.72 / angle from +V3 to image +x (toward +V2)
BETA2 = 224.72 / angle from +V3 to image +y (toward +V2)

/ OBSERVATION DATA
PROPOSID= 05233 / PEP proposal identifier
PROGRMID= '288' / program id (base 36)
OBSET_ID= '02' / observation set id
OBSRVTN= '03' / observation number (base 36)
TARGNAME= 'NGC3379-PO' / proposer's target name
STARTIME= '1994.133:06:24:18.35' / predicted observation window start time
ENDTIME = '1994.133:06:39:18.35' / predicted observation window end time
SOGSID = 'U2880203' / SOGS observation name !

/ SCIENTIFIC INSTRUMENT DATA
CONFIG = 'WFPC2' / proposed instrument configuration
PRIMARY = 'SINGLE' / single, parallel-primary, parallel-secondary
OPERATE = '1994.133:06:22:46.91' / predicted time instr. entered operate mode
TLMFORM = 'PN' / telemetry format
APER_TURE= 'UWFALL' / aperture name
APER_V2 = 1.565 / V2 aperture position in vehicle frame (arcsec)
APER_V3 = 7.534 / V3 aperture position in vehicle frame (arcsec)

/ SPACECRAFT DATA
ALTITUDE= 593.23 / average altitude during observation (km)
LOS_SUN = 106.08 / minimum line of sight to Sun (deg)
LOS_MOON= 77.11 / minimum line of sight to Moon (deg)
SHADOENT= '1994.133:05:11:29.00' / predicted Earth shadow last entry
SHADOEXT= '1994.133:05:42:45.00' / predicted Earth shadow last exit
LOS_SCV = 12.46 / minimum line of sight to S/C veloc. (deg)
LOS_LIMB= 58.0 / average line of sight to Earth limb (deg)

/ BACKGROUND LIGHT
ZODMOD = 22.3 / zodiacal light - model (V mag/arcsec2)
EARTHMOD= 20.2 / peak Earth stray light - model (V mag/arcsec2)
MOONMOD = 35.5 / moon stray light - model (V mag/arcsec2)
GALACTIC= -1.0 / diffuse galactic light - model (V mag/arcsec2)

/ POINTING CONTROL DATA
GUIDECMD= 'FINE LOCK' / commanded guiding mode
GUIDEACT= 'FINE LOCK' / actual guiding mode at end of GS acquisition
GSD_ID = '0084900235' / dominant guide star id
GSD_RA = 161.70720 / dominant guide star RA (deg)
GSD_DEC = 12.45407 / dominant guide star DEC (deg)

```

Figure C.2:Representative .jih or .cmh Header

```

GSD_MAG =          12.867 / dominant guide star magnitude
GSR_ID   = '0085201189'   / roll guide star id
GSR_RA   =          161.93314 / roll guide star RA (deg)
GSR_DEC  =          12.78141 / roll guide star DEC (deg)
GSR_MAG  =          12.977 / roll guide star magnitude
GSACQ    = '1994.133:06:31:02.92' / actual time of GS acquisition completion
PREDGSEP=          1420.775 / predicted guide star separation (arcsec)
ACTGSSEP=          1421.135 / actual guide star separation (arcsec)
GSSEPRMS=           3.8 / RMS of guide star separation (milli-arcsec)
NLOSSES  =           0 / number of loss of lock events
LOCKLOSS=           0.0 / total loss of lock time (sec)
NRECENT  =           0 / number of recentering events
RECENTR  =           0.0 / total recentering time (sec)

/ LINE OF SIGHT JITTER SUMMARY
V2_RMS   =           4.5 / V2 axis RMS (milli-arcsec)
V2_P2P   =          51.6 / V2 axis peak to peak (milli-arcsec)
V3_RMS   =           20.9 / V3 axis RMS (milli-arcsec)
V3_P2P   =          267.3 / V3 axis peak to peak (milli-arcsec)
RA_AVG   =          161.85226 / average RA (deg)
DEC_AVG  =          12.58265 / average dec (deg)
ROLL_AVG =          293.01558 / average roll (deg)

/ PROBLEM FLAGS, WARNINGS and STATUS MESSAGES
/ (present only if problem exists)
ACQ2FAIL= '          ' T' / target acquisition failure
GSFAIL   = 'DEGRADED' T' / guide star acquisition failure (*1)
TAPEDROP= '          ' T' / possible loss of science data
TLM_PROB= '          ' T' / problem with the engineering telemetry
TM_GAP   =          404.60 / duration of missing telemetry (sec)
SLEWING  = '          ' T' / slewing occurred during this observation
TAKEDATA= '          ' F' / take data flag NOT on throughout observation
SIPROBnn= '          ' / problem with specified science instrument (*2)

END

```

notes

*1 - GSFAIL appears only once in a single header file.

The following table lists all current possible values for the GSFAIL keyword:

GSFAIL	DEGRADED	/ guide star acquisition failure
	IN PROGR	/ guide star acquisition failure
	SSLEXP	/ guide star acquisition failure
	SSLEXS	/ guide star acquisition failure
	NOLOCK	/ guide star acquisition failure
	SREXCS?	/ guide star acquisition failure
	SREXCS1	/ guide star acquisition failure
	SREXCS2	/ guide star acquisition failure
	SREXCS3	/ guide star acquisition failure
	SREXCP?	/ guide star acquisition failure
	SREXCP1	/ guide star acquisition failure
	SREXCP2	/ guide star acquisition failure
	SREXCP3	/ guide star acquisition failure
	UNKNOWN	/ guide star acquisition failure
	VEHSAFE	/ guide star acquisition failure

*2 - The SIPROBnn keywords appear in the header file with nn = 01 - 99.

The following table lists all current possible values for the SIPROBnn keyword:

SIPROBnn	DCF_NUM unchanged	/ This observation may not have been taken
	FOS Safing!	/ This observation affected when FOS Safed!
	HRS Safing!	/ This observation affected when HRS Safed!
	WFII Safing!	/ This observation affected when WFII Safed!
	FOC Safing!	/ This observation affected when FOC Safed!
	Shut	/ FOS aperture door is not Open!
	FAILED	/ FGS astrometry target acquisition failed

Table C.3: Contents of .jit or .cmi Table, Three-Second Averaging

Parameter	Units	Description
seconds	seconds	Time since window start
V2 dom	arcseconds	Dominant FGS V2 coordinate
V3 dom	arcseconds	Dominant FGS V3 coordinate
V2 roll	arcseconds	Roll FGS V2 coordinate
V3 roll	arcseconds	Roll FGS V3 coordinate
SI V2 AVG	arcseconds	Mean jitter in 3 seconds
SI V2 RMS	arcseconds	rms jitter in 3 seconds
SI V2 P2P	arcseconds	Peak jitter in 3 seconds
SI V3 AVG	arcseconds	Mean jitter in 3 seconds
SI V3 RMS	arcseconds	rms jitter in 3 seconds
SI V3 P2P	arcseconds	Peak jitter in 3 seconds
RA	degrees	Right ascension of aperture reference
DEC	degrees	Declination of aperture reference
Roll	degrees	Angle between North and +V3
LimbAng	degrees	Angle between earth limb and target
TermAng	degrees	Angle between terminator and target
LOS_Zenith	degrees	Angle between HST zenith and target
Latitude	degrees	HST subpoint latitude
Longitude	degrees	HST subpoint longitude
Mag V1,V2,V3	degrees	Magnetic field along V1, V2, V3
EarthMod	V Mag/arcsec ²	Model earth background light
SI_Specific	–	Special science instrument data
DayNight	0,1 flag	Day (0) or night (1)
Recenter	0,1 flag	Recentering status
TakeData	0,1 flag	Vehicle guiding status
SlewFlag	0,1 flag	Vehicle slewing status

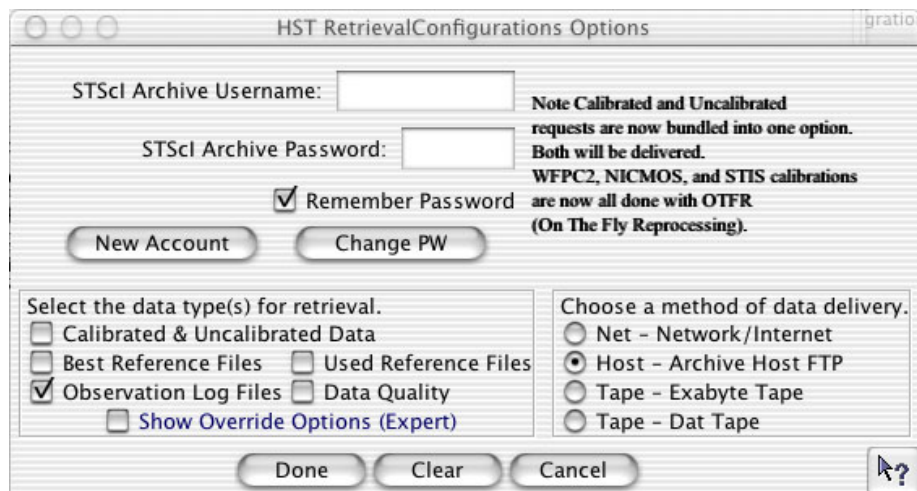
C.2 Retrieving Observation Logs

You can retrieve observation log files for data taken after October 20, 1994 from the HST Archive using StarView as described in chapter 1 of the HST Introduction. Unlike science data, which generally has a one-year proprietary period, observation log files become public as soon as they are archived.

The easiest way to get OMS files through StarView is to identify the observation of interest and proceed with your request as described in chapter 1 of the HST Introduction, until you reach the "HST Retrieval Configurations Options" screen, reproduced in figure C.3. You can then check the Observation Log Files box, along with any other desired boxes, and continue with your request. StarView will then deliver the associated observation log files.

For observations logged between October 1994 to August 1995, you will be delivered the `cmi`, `cmj`, and `cmh` files in FITS form (e.g., `_cmi.fits`). Observations archived from August 1995 to February 1997 will return `_jif.fits` and `_jit.fits` files. These, and the earlier FITS files can be worked with as such, or converted to their GEIS counterparts via the STSDAS **strfits** task. However, as of February 1997, the `_jif.fits` and `_jit.fits` files are standard FITS files with extensions and cannot be converted to GEIS.

Figure C.3:Choosing Observation Log Files in StarView



C.3 Using Observation Logs

Here are some simple examples of what can be learned from the observation log files. Note that for FITS format observation logs, current versions of STSDAS tools will handle the files with extensions properly. Keywords can be viewed with tools such as **imheader** or **hedit**, and data viewed, plotted, or displayed using the same tasks one might have for the GEIS files. For more information on FITS file structures, see chapter 2 of the HST Introduction.

C.3.1 Guiding Mode

Unless requested, all observations will be scheduled with FINE LOCK guiding, which may be one or two guide stars (dominant and roll). The spacecraft may roll slightly during an observation if only one guide star is acquired. The amount of roll depends upon the gyro drift at the time of the observation, the location during an orbit, and the lever arm from the guide star to the center of the aperture.

There are three commanded guiding modes: FINE LOCK, FINE LOCK/GYRO, and GYRO. OMS header keywords GUIDECMD (commanded guiding mode) and GUIDEACT (actual guiding mode) will usually agree. If there was a problem, they won't agree and the GUIDEACT value will be the guiding method actually used during the exposure. If the acquisition of the second guide star fails, the spacecraft guidance, GUIDEACT, may drop from FINE LOCK to FINE LOCK/GYRO, or even to GYRO, which may result in a target rolling out of an aperture. Check the OMS header keywords to verify that there was no change in the requested guiding mode during the observation.



Until new flight software (version FSW 9.6) came online in September 1995, if the guide star acquisition failed, the guiding dropped to COARSE track. After September 1995, if the guide star acquisition failed, the tracking did not drop to COARSE track. Archival researchers may find older datasets that were obtained with COARSE track guiding.

The dominant and roll guide star keywords (GSD and GSR) in the OMS header can be checked to verify that two guide stars were used for guiding, or in the case of an acquisition failure, to identify the suspect guide star. The dominant and roll guide star keywords identify the stars that were scheduled to be used, and in the event of an acquisition failure, may not be

the stars that were actually used. The following list of `cmh` keywords is an example of two star guiding.

```
GSD_ID = '0853601369' / Dominant Guide Star ID
GSD_RA =          102.42595 / Dominant Guide Star RA (deg)
GSD_DEC =        -53.41362 / Dominant Guide Star DEC (deg)
GSD_MAG =          11.251 / Dominant Guide Star Magnitude
GSR_ID = '0853602072' / Roll Guide Star ID
GSR_RA =          102.10903 / Roll Guide Star RA (deg)
GSR_DEC =        -53.77683 / Roll Guide Star DEC (deg)
GSR_MAG =          12.426 / Roll Guide Star Magnitude
```

If you suspect that a target has rolled out of the aperture during an exposure, you can quickly check the counts in each group of the raw science data. As an example, the following IRAF commands can be used to determine the counts in each group.

```
cl> grlist z2o4040dt.d0h 1-24 > groups.lis
cl> imstat @groups.lis
```

Some observations can span several orbits. If during a multiple orbit observation the guide star reacquisition fails, the observation may be terminated with possible loss of observing time, or switch to other less desirable guiding modes. The `GSACQ` keyword in the `cmh` header will state the time of the last successful guide star acquisition.

```
GSACQ = '136:14:10:37.43' / Actual time of GS Acquisition Completion
```

C.3.2 Guide Star Acquisition Failure

The guide star acquisition at the start of the observation set could fail if the FGS fails to lock onto the guide star. The target may not be in the aperture, or maybe only a piece of an extended target is in the aperture. The jitter values will be increased because `FINE LOCK` was not used. The following list of `cmh` header keywords indicate that the guide star acquisition failed.

```
V3_RMS =          19.3 / V3 Axis RMS (milli-arcsec)
V3_P2P =          135.7 / V3 Axis peak to peak (milli-arcsec)
GSFAIL = '          DEGRADED' / Guide star acquisition failure!
```

The observation logs for all of the following observations in the observation set will have the “`DEGRADED`” guide star message. This is not a Loss of Lock situation but an actual failure to acquire the guide star in

the desired guiding mode. For the example above, the guiding mode dropped from FINE LOCK to COARSE TRACK.

GUIDECMD= 'FINE LOCK	' / Commanded Guiding mode
GUIDEACT= 'COARSE TRACK	' / Actual Guiding mode at end of GS acquisition

If the observational dataset spans multiple orbits, the guide star will be re-acquired, but the guiding mode will not change from COARSE TRACK. In September 1995, the flight software was changed so that COARSE TRACK is no longer an option. The guiding mode drops from two guide star FINE LOCK to one guide star FINE LOCK , or to GYRO control.

C.3.3 Moving Targets and Spatial Scans

A type 51 slew is used to track moving targets (planets, satellites, asteroids, and comets). Observations are scheduled with FINE LOCK acquisition, i.e., with two or one guide stars. Usually, a guide star pair will stay within the pickle during the entire observation set, but if two guide stars are not available, a single guide star may be used, assuming the drift is small or the proposer says that the roll is not important for that particular observing program. An option during scheduling is to drop from FGS control to GYRO control when the guide stars move out of the FGS. Also, guide star handoffs (which are not a simple dropping of the guide stars to GYRO control) will affect the guiding and may be noticeable when the jitter ball is plotted.

The jitter statistics are accumulated at the start of the observation window. Moving targets and spatial scan motion will be seen in the jitter data and image. Therefore, the OMS header keywords V2_RMS and V3_RMS values (the root mean square of the jitter about the V2 and V3 axis) can be quite large for moving targets. Also, a special anomaly keyword (SLEWING) will be appended to the OMS header stating movement of the telescope during the observation. This is expected for observing moving targets. The following list of .cmh header keywords is an example of expected values while tracking a moving target.

	/ LINE OF SIGHT JITTER SUMMARY
V2_RMS =	3.2 / V2 Axis RMS (milli-arcsec)
V2_P2P =	17.3 / V2 Axis peak to peak (milli-arcsec)
V3_RMS =	14.3 / V3 Axis RMS (milli-arcsec)
V3_P2P =	53.6 / V3 Axis peak to peak (milli-arcsec)
RA_AVG =	244.01757 / Average RA (deg)
DEC_AVG =	-20.63654 / Average DEC (deg)
ROLL_AVG=	280.52591 / Average Roll (deg)
SLEWING =	' T' / Slewing occurred during this observation

C.3.4 High Jitter

The spacecraft may shake during an observation, even though the guiding mode is FINE LOCK. This movement may be due to a micro-meteorite hit, jitter at a day-night transition, or for some other unknown reasons. The FGS is quite stable and will track a guide star even during substantial spacecraft motion. The target may move about in an aperture, but the FGS will continue to track guide stars and reposition the target into the aperture. For most observations, the movement about the aperture during a spacecraft excursion will be quite small, but sometimes, especially for observations with the spectrographs, the aperture may move enough that the measured flux for the target will be less than a previous group. Check the OMS header keywords (V2_RMS, V3_RMS) for the root mean square of the jitter about the V2 and V3 axis. The following list of .cmh header keywords is an example of typical guiding rms values.

	/ LINE OF SIGHT JITTER SUMMARY
V2_RMS =	2.6 / V2 Axis RMS (milli-arcsec)
V2_P2P =	23.8 / V2 Axis peak to peak (milli-arcsec)
V3_RMS =	2.5 / V3 Axis RMS (milli-arcsec)
V3_P2P =	32.3 / V3 Axis peak to peak (milli-arcsec)

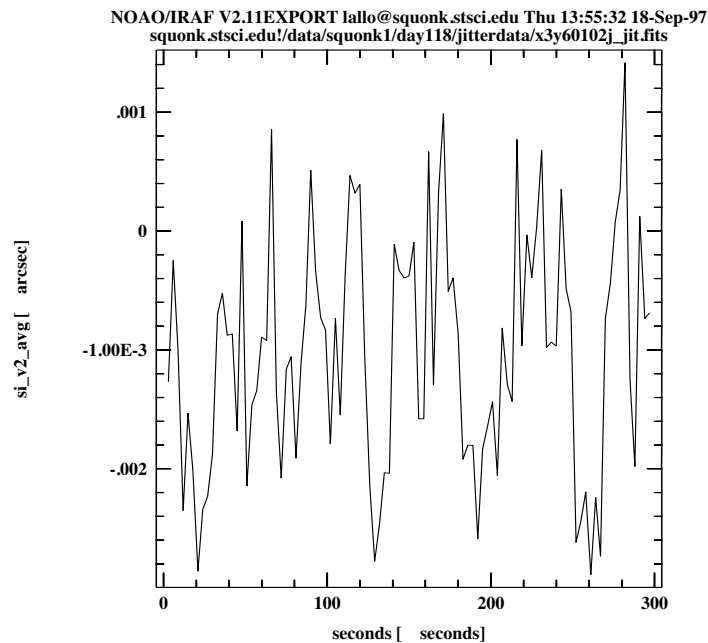
Recentering events occur when the spacecraft software decides that shaking is too severe to maintain lock. The FGS will release guide star control and within a few seconds reacquire the guide stars. It is assumed the guide stars are still within the FGS field of view. During the recentering time, INDEF will be written to the OMS table. Recentering events are tracked in the OMS header file.

Be careful when interpreting “Loss of Lock” and “Recentering” events that occur at the very beginning or at the end of the OMS window. The OMS window is larger than the observation window. These events might not affect the observation since the observation start time will occur after the guide stars are acquired (or re-acquired), and the observation stop time may occur before the “Loss of Lock” or “Recentering” event that occurred at the end of an OMS window.

The **sgraph** command in the **stdas.graphics.stplot** package will plot time vs. jitter along the direction of HST’s V2 axis (see figure C.4):

```
cl> sgraph "x3y60102j_jit.fits seconds si_v2_avg"
```

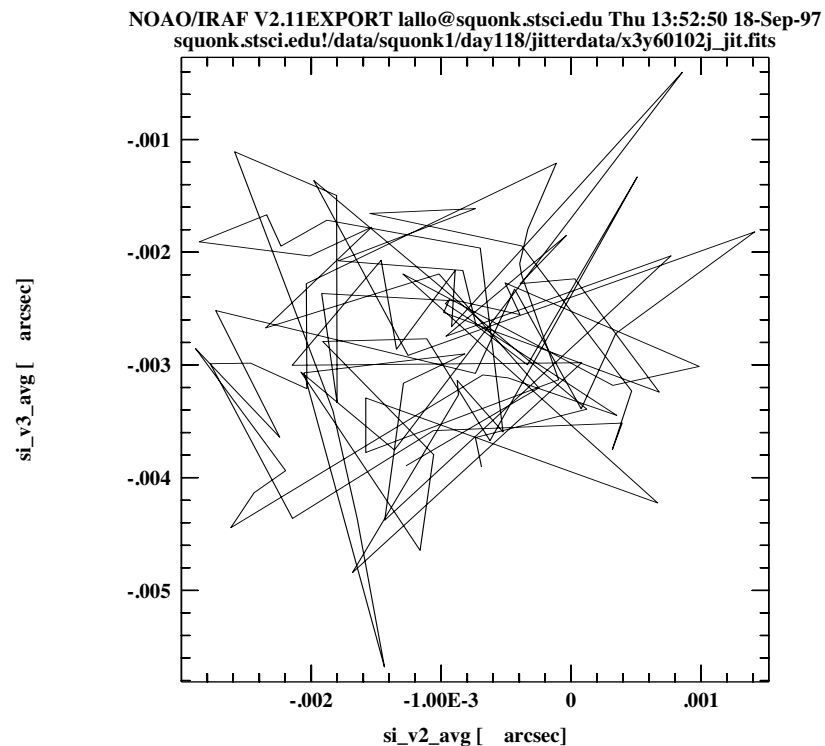
Figure C.4:Plotting Jitter Along V3 Axis



To get an idea of pointing stability, you can create a *jitter ball* by plotting jitter along the V2 axis vs. jitter along the V3 axis (see figure C.5):

```
st> sgraph "x3660102j_jit.fits si_v2_avg si_v3_avg"
```

Figure C.5:Plotting V2 vs. V3 Jitter



The **tstatistics** task can be used to find the mean value of the **si_v3_avg** column—the amount of jitter (in arcseconds) in the direction of the V3. This value can be used to model jitter in a PSF. In this example, the mean jitter is ~3 mas, which is typical for post-servicing mission data:

Figure C.6:Averaging a Column with tstatistics

```
tt> tstat u26m0801j.cmi si_v3_avg
# u26m0801j.cmi si_v3_avg
#
nrows      mean      stddev      median      min      max
  11    -0.003006443888  0.00362533  -7.17163E-4  -0.00929515  0.00470988
```



Understanding and interpreting the meaning of the table columns and header keywords is critical to understanding the observation logs. Please read the available documentation and contact the STScI Help Desk (help@stsci.edu.) if you have any questions about the files. Documentation is available via the WWW at: http://www.stsci.edu/instruments/observatory/obslog/OL_1.html.

Index

A

- ACCUM mode
 - STIS, data storage STIS:2—6
- accuracy
 - astrometric, improving INTRO:3—11
- acquisition failure
 - guide stars APP:C—11
- algorithm
 - calstis STIS:3—3
- analog-to-digital conversion
 - STIS STIS:3—16
- analysis
 - images, general in STSDAS INTRO:3—9
 - spectra, general in STSDAS INTRO:3—24
 - spectra, general tasks in IRAF
 - INTRO:3—30
 - spectra, STIS INTRO:3—27
 - STIS, software tools STIS:5—1
- analysis package
 - image analysis INTRO:3—4
- apphot package
 - aperture photometry INTRO:3—17
- archive
 - file names INTRO:2—1
- arithmetic
 - imset, msarith task INTRO:3—15
 - spectra, splot task INTRO:3—31
- array
 - FITS table INTRO:2—11
- astrometry
 - basic, in STSDAS INTRO:3—10
 - improving accuracy INTRO:3—11
 - tasks in STSDAS INTRO:3—11

B

- background
 - running tasks in APP:A—7
- bar
 - see fiducials STIS:1—3
- basic2d task
 - STIS 2-D image reduction STIS:3—2
- bias correction
 - electronic, STIS STIS:3—18
 - STIS STIS:3—17
- buffer
 - STIS STIS:1—8

C

- calibration
 - software, STSDAS INTRO:3—1
 - STIS STIS:3—1, STIS:3—38
 - STIS keywords STIS:2—13
 - STIS reference file keywords STIS:2—15
 - switches
 - STIS STIS:3—40
 - switches, see also "chcalpar"
- calstis task
 - development STIS:3—44
 - environment variables, setting STIS:3—40
 - process STIS:3—3
 - recalibrating data STIS:3—39
 - STIS calibration STIS:3—1
- CCD
 - STIS, cosmic ray rejection STIS:3—19
 - STIS, described STIS:1—5
- chcalpar task
 - STIS calibration switches STIS:3—40

command
 see "task"
 commands
 splot cursor INTRO:3—33
 conversion
 counts to flux or magnitude INTRO:3—18
 flux to wavelength, resample task
 INTRO:3—24
 cosmic ray
 rejection, STIS STIS:3—19
 counts
 flux conversion INTRO:3—18
 magnitude conversion INTRO:3—18
 cursor
 splot commands INTRO:3—33

D

dark
 STIS STIS:3—22
 data
 analysis software, STSDAS INTRO:3—1
 parameter types, IRAF APP:A—10
 STIS, described STIS:2—1—STIS:2—23
 storage, STIS STIS:1—8
 data quality
 PDQ files APP:B—4
 STIS STIS:2—16
 STIS, calibration STIS:3—25
 STIS, flags STIS:2—17
 database
 synphot APP:A—16
 dataset
 see also "imset"
 detector
 STIS MAMA STIS:1—5
 STIS, described STIS:1—5
 disconlab task
 position display INTRO:3—10
 dispersion
 coefficients, STIS STIS:3—23
 display
 display task INTRO:3—4
 image INTRO:3—4
 SAOimage INTRO:3—6
 spectra INTRO:3—20

display task
 images in STSDAS INTRO:3—5
 distortion STIS:4—3
 documentation
 IRAF INTRO:3—37
 STSDAS INTRO:3—37
 Doppler shift
 STIS MAMA STIS:3—24

E

echelle spectra
 echplot task INTRO:3—22
 echplot task
 plot echelle spectra
 INTRO:3—22, STIS:5—2
 engineering data
 OMS logs APP:C—1
 environment variable
 IRAF APP:A—11
 eparam task
 editing parameters APP:A—9
 error image
 STIS STIS:2—16
 extension
 FITS file INTRO:2—3
 FITS, appending INTRO:2—8
 extracted spectra
 STIS, storage STIS:2—8

F

file
 PostScript, creating INTRO:3—23
 files
 data formats APP:A—13
 data quality (PDQ) APP:B—4
 FITS, working with INTRO:2—5
 naming conventions INTRO:2—1
 naming conventions, STIS STIS:2—3
 observation log APP:B—3
 observer comments (OCX) APP:B—4
 rootname APP:B—3
 specifying, STIS INTRO:3—21
 trailer APP:B—4

FINE LOCK

- guidance APP:C—10

FITS

- files, working with INTRO:2—3
 - INTRO:2—12
- format, described INTRO:2—3
- GEIS files in INTRO:2—13
- table INTRO:2—9
- table, array in cell INTRO:2—11
- waiver INTRO:2—17

fitting package

- fit spectra INTRO:3—34
- tasks in INTRO:3—34

flag

- see "data quality"

flatfield

- STIS STIS:3—26

flux

- combine with wavelength,
 - mkmultispec INTRO:3—25
- correction, STIS STIS:3—27
- from counts INTRO:3—18

format

- IRAF and STSDAS files APP:A—13

FOS

- display spectra INTRO:3—20

fwplot task

- spectra display INTRO:3—20

G**GEIS format**

- described APP:A—13
- header file INTRO:2—15
- working with INTRO:2—15

geometric correction

- STIS STIS:3—28

geometric distortion

- correction INTRO:3—10

GHR

- display spectra INTRO:3—20

grating wheel

- described, STIS STIS:1—6

group

- number in image INTRO:2—14
- working with INTRO:3—8

grspec task

- plot groups INTRO:3—21

guidance mode

- observation log APP:C—10

guide stars

- acquisition APP:C—10
- acquisition failure APP:C—11
- dominant roll APP:C—10
- number used APP:C—10

H**hardcopy**

- see "print" or "paper products"

header

- file, GEIS INTRO:2—15
- keyword, inheritance in FITS
 - INTRO:2—7

header data unit

- FITS file INTRO:2—3

hedit task

- set STIS keywords STIS:3—42

heliocentric reference

- STIS STIS:3—30

help

- STSDAS and IRAF tasks
 - INTRO:3—2, APP:A—7

hole in the mirror

- calibration system, STIS STIS:1—7

I**icons**

- used in this manual STIS:2—4

igi

- plotting with INTRO:3—23
- printing plots INTRO:3—22

image

- display INTRO:3—4, INTRO:3—5
- GEIS file INTRO:2—16
- plot data, implot INTRO:3—12
- section INTRO:3—8
- see also "FITS"
- STIS calibration STIS:3—16
- STSDAS tasks INTRO:3—2
- working with INTRO:3—2

Image Reduction and Analysis Facility
see "IRAF"

image set
see "imset"

imcopy task
FITS files INTRO:2—8

imexamine task
image display and plot INTRO:3—13

imgtools package
multigroup GEIS images INTRO:3—2

implot task
plot image data INTRO:3—12

imset
combination, msjoin task INTRO:3—16
extraction, mssplit task INTRO:3—16
statistics, msstatistics task INTRO:3—16
STSDAS tasks for INTRO:3—14

imtab task
header to table INTRO:3—26

infostis task
display STIS keywords STIS:2—10

insert mechanism
calibration system
STIS STIS:1—7

inttag task
STIS time-tag data STIS:3—43

IPPSSOOT
see "files, naming conventions"

IRAF
basics APP:A—1
described INTRO:3—1, APP:A—1
documentation INTRO:3—37
obtaining APP:A—15
parameter, data type APP:A—10
piping APP:A—6
psikern, PostScript INTRO:3—23
setup APP:A—2
spectra analysis INTRO:3—30
tasks available APP:A—1

J

jitter
effect on target lock APP:C—13
images from OMS APP:C—1
plotting APP:C—14

K

keystroke
commands STIS:2—3
keywords
FITS header INTRO:2—7
see also "header"
STIS calibration switches STIS:2—13
STIS header STIS:2—9
STIS reference files STIS:2—15

L

lparam task
viewing parameters APP:A—9

M

magnitude
from counts INTRO:3—18
MAMA
described, STIS STIS:1—5
Doppler shift STIS:3—24
small scale geometric distortion
STIS:3—33

math
see "arithmetic"

mkiraf command
IRAF setup APP:A—3
mkmultispec task INTRO:3—25
mode selection mechanism
see "grating wheel"

moving target
acquisition APP:C—12

msarith task
imset arithmetic INTRO:3—15

mscombine task
combine imset INTRO:3—15

msjoin task
combine imset INTRO:3—16

mssplit task
extract imset INTRO:3—16

msstatistics task
imset statistics INTRO:3—16

mstools package
FITS image extensions INTRO:3—4
STIS data STIS:5—1

multispec format
described INTRO:3—25

N

naming conventions
files, HST data INTRO:2—1
nfit1d task INTRO:3—34
ngaussfit task INTRO:3—34
NICMOS
CCD, described STIS:1—5
imset, STSDAS tasks INTRO:3—14

O

observation log
files APP:B—3
Observation Monitoring System APP:C—1
observer comment file
described APP:B—4
OCX file
observer comments APP:B—4
OMS
observation log files APP:B—3
OPUS
see "pipeline"

P

package
IRAF concept APP:A—5
STSDAS, structure INTRO:3—2, IN-
TRO:3—3
paper products
STIS STIS:2—20
parameter
data types APP:A—10
see also "eparam" and "lparam"
setting, IRAF APP:A—9
PDQ file
described APP:B—4
pickup
STIS STIS:1—7
photometry
basic, in STSDAS INTRO:3—17
synthetic INTRO:3—19

pipe
IRAF task APP:A—6
pipeline
files produced by INTRO:2—1
STIS calibration STIS:3—1
STIS, output STIS:2—1
pixel coordinate
converting to RA and Dec INTRO:3—10
pixel data
GEIS file INTRO:2—16
plot
igi task INTRO:3—23
pointing stability APP:C—13
position
RA and Dec in STSDAS INTRO:3—10
PostScript
psikern, IRAF INTRO:3—23
pp_dads task
generate paper products STIS:2—20
print
plots, igi INTRO:3—22
proposal
keywords compared to Phase II, STIS
STIS:2—11
psikern
PostScript IRAF kernel INTRO:3—23

Q

quality
data, see "data quality"

R

recalibration
see also "calibration" STIS:3—38
STIS STIS:3—38
recentering
jitter APP:C—13
reference files
retrieving, STIS STIS:3—39
STIS, keywords STIS:2—15
resample task
flux to wavelength INTRO:3—24
resolution **STIS:1—3**
rootname
see "files, naming conventions"

S

SAOimage

display image INTRO:3—6

scattered light STIS:4—2

section

image INTRO:3—8

sgraph task

plot group INTRO:3—20

plot STIS spectra INTRO:3—21

slit wheel

described, STIS STIS:1—6

small scale geometric distortion

MAMA STIS:3—33

software

IRAF APP:A—1, APP:A—15

see also "IRAF" and "STSDAS" STIS:3—44

STIS analysis STIS:5—1

STSDAS INTRO:3—1, APP:A—15

Space Telescope Science Data Analysis

System

see "STSDAS"

specfit task

fit models to spectrum INTRO:3—37

spectra

analysis tasks, IRAF INTRO:3—30

analysis, STSDAS INTRO:3—24

analysis, tasks in STSDAS INTRO:3—31,
STIS:5—2

display INTRO:3—20

display, STIS INTRO:3—21

fitting INTRO:3—34

locate, STIS STIS:3—35

specfit task INTRO:3—37

spectroscopy

slitless STIS:1—3

splot task

cursor commands INTRO:3—33

plot spectra INTRO:3—31

stability STIS:4—2, STIS:4—3

StarView

observation logs APP:C—9

STIS

ACCUM data, storage STIS:2—6

analysis tasks STIS:5—3

analysis, preparing INTRO:3—27

bias level correction STIS:3—18

calibrating, when to STIS:3—38

calibration STIS:3—1

calibration, image STIS:3—16

dark signal STIS:3—22

data quality STIS:2—16

data quality, calibration STIS:3—25

data storage STIS:1—8

described STIS:1—1

detectors STIS:1—5

display spectra INTRO:3—21

echelle spectra, plotting INTRO:3—22

error array STIS:2—16

file names STIS:2—3

flatfield STIS:3—26

flux correction STIS:3—27

geometric correction STIS:3—28

header keywords STIS:2—9

hole in the mirror STIS:1—7

imset, STSDAS tasks for INTRO:3—14

insert mechanism STIS:1—7

keywords, calibration STIS:2—13

MAMA STIS:1—5

one-dimensional extracted

spectra STIS:2—8

paper products STIS:2—20

pickup STIS:1—7

proposal keywords, Phase II STIS:2—11

target acquisition STIS:1—7

target acquisition data STIS:5—12

time-tag data, storage STIS:2—8

wavecal STIS:1—7

STSDAS INTRO:3—1

astrometry INTRO:3—11

astrometry in INTRO:3—10

described APP:A—1

documentation INTRO:3—37

image analysis tasks INTRO:3—14

image display INTRO:3—4, INTRO:3—5

image, display INTRO:3—8

image, section INTRO:3—8

images INTRO:3—2

imset tasks INTRO:3—14

obtaining APP:A—15

organization of INTRO:3—2, INTRO:3—3

photometry in INTRO:3—17

- spectra analysis tasks
 - INTRO:3—31, STIS:5—2
- synphot, database APP:A—16
- synthetic photometry INTRO:3—19
- tables INTRO:3—4
- suffix
 - see "files, naming conventions"
- synphot
 - database, obtaining APP:A—16
- synphot package
 - synthetic photometry INTRO:3—19
- synthetic photometry
 - see "synphot"

T

- table
 - FITS INTRO:2—9
 - STSDAS INTRO:3—4
- target acquisition STIS:4—3
 - moving target APP:C—12
 - STIS STIS:1—7
 - STIS, analysis STIS:5—12
- task
 - IRAF concept APP:A—6
- time-tag data
 - inttag task STIS:3—43
 - STIS, storage of STIS:2—8
- tomultipsec task
 - extract STIS spectral orders INTRO:3—27
- trailer file
 - described APP:B—4
- transformation
 - pixel coordinates to RA and Dec
INTRO:3—10
- tread task
 - display STIS association table STIS:2—4
- ttools package
 - STSDAS tables INTRO:3—4
- txtable task
 - extract arrays INTRO:3—29
- typographic conventions
 - in this manual STIS:2—3

U

- unit
 - counts to flux/mag, conversion
INTRO:3—18
- user support
 - help desk ii

V

- variable
 - IRAF, environment APP:A—11

W

- waiver FITS format INTRO:2—17
- warning
 - types in this document STIS:2—4
- wavecal STIS:4—2
 - STIS STIS:1—7
- wavelength
 - combine with flux, mkmultispec
INTRO:3—25
- Wavelengths STIS:3—23
- World Coordinate System
 - mkmultispec task INTRO:3—25
- world wide web
 - software, obtaining APP:A—15

X

- xy2rd task
 - pixel coordinates to RA and Dec
INTRO:3—10

