

TO: Distribution DATE: July 24, 1986
 FROM: Keith Horne
 SUBJECT: Proposed Changes to SO-11

This is the proposed new chapter in SO-11 to replace the CDBS requirements for "absolute flux calibration" and "photometric transformations" with requirements for an observatory throughput calibration based on synthetic photometry techniques.

Contents

16 Throughput Calibration of HST Observations **2**

16.1 Synthetic Photometry Package 3

 16.1.1 Properties of Passbands 3

 16.1.2 Properties of Passbands and Spectra 5

 16.1.3 Magnitudes and Units Conversions 6

16.2 Throughput Calibration Databases and Supporting Software 9

 16.2.1 Component Throughput Functions 9

 16.2.2 The Observatory Graph and Dynamic Throughput Generator 10

 16.2.3 Calibration Target Spectra 13

 16.2.4 Calibration Observations and the Comparison of Observed and Predicted Responses 15

References

Horne,K., Burrows,C. & Koornneef,J. 1986 "Dynamic Generation of Throughput Functions: A Unified Approach", STScI memo presented at the ISB meeting of 16 May 1986.

Koornneef,J., Bohlin,R., Buser,R., Horne.K. & Turnshek,D. 1986, "Synthetic Photometry and the Calibration of the Hubble Space Telescope", to appear in *Highlights of Astronomy*, Vol 7, ed. J.P.Swings (STScI preprint No.91; also distributed at the 16 May 1986 ISB meeting).

16 Throughput Calibration of HST Observations

The throughput calibration of the HST observatory will be based on the techniques of synthetic photometry as follows. The passband throughput function, $P(\lambda)$, is by definition the expected response of the detector to a target photon of wavelength λ that enters the 240 cm diameter aperture of the telescope. A model of the HST observatory configuration will be used to generate the passband throughput function for every HST observation. The sensitivity, which is used to express the observed response in terms of an absolute energy flux density, will be calculated from the corresponding passband. The accuracy of the throughput model will be monitored by comparing observed responses on calibration targets against predicted responses calculated from the corresponding passband functions and target spectra. Discrepancies will be resolved by adjusting the passbands and target spectra until the predicted and observed responses agree to within observational uncertainties. This calibration process will be applied uniformly across all science instruments and observing modes.

Section 16.1 provides specifications for the CDBS library of utility subroutines that will be used to perform the simple numerical integrals that are involved in synthetic photometry, in particular the calculation of sensitivities from passbands and of predicted count rates from passbands and target spectra.

Section 16.2 provides specifications for the four CDBS databases and associated software that will be used to generate and maintain the throughput calibration of the HST observatory.

Section 16.2.1 describes the **components database**, which provides passband throughput functions for all optical elements in the HST observatory.

Section 16.2.2 describes the **observatory graph database**, which provides the information needed to determine which optical components are present in the optical beam, and the **dynamic throughput generator**, which combines the passbands of individual components to generate the passband for any specified observing mode.

Section 16.2.3 describes the **target spectrum database**, which provides absolute flux distributions for each of the HST calibration targets.

Section 16.2.4 describes the **observations database**, which provides access to observations of calibration targets (including spectrophotometry and broadband photometry from ground-based, IUE, and other sources as well as the observations with HST), and software for comparing observed and predicted responses, which will be used to monitor the throughput calibration and to generate updates to the observatory graph and components databases.

16.1 Synthetic Photometry Package

This section describes the library of synthetic photometry subroutines that will be used to calculate sensitivities from passbands and predicted responses from passbands and spectra. The following constants and definitions will be used:

- $h = 6.6262 \times 10^{-27}$ is Planck's constant in *erg s*.
- $c = 2.997925 \times 10^{10}$ is the speed of light in *cm s⁻¹*.
- D = telescope diameter in *cm* (= 240 for HST).
- $\Lambda = 10^{10}$ = the number of \AA in a *cm*.
- λ = wavelength in \AA .
- f_λ = flux density in *erg cm⁻²s⁻¹\AA⁻¹*.
- f_ν = flux density in *erg cm⁻²s⁻¹Hz⁻¹*.
- f_{mJy} = flux density in *mJy* = 10^{-26} *erg cm⁻²s⁻¹Hz⁻¹*.
- $ST = -2.5 \log [f_\lambda] - ST_0$.
- $AB = -2.5 \log [f_\nu] - AB_0$.
- $ST_0 = 21.10$ = zero-point of ST magnitude system.
- $AB_0 = 48.60$ = zero-point of AB magnitude system.

Note that the adopted zero-points of the *AB* and *ST* magnitude systems are chosen so that Vega has a magnitude near 0 in the Johnson *V* passband.

16.1.1 Properties of Passbands

This subsection covers utilities that compute sensitivities, characteristic wavelengths and bandwidths, which involve numerical integrals over wavelength weighted by the passband throughput function.

The inputs to these routines are as follows:

D is the telescope diameter in *cm* ($D = 240$ for the HST).

M is the number of wavelengths.

$\lambda(M)$ is the array of wavelengths (in \AA) at which the passband throughput is supplied.

$P(M)$ is the array of dimensionless throughputs giving at each wavelength the expected number of counts that the detector registers for each photon that is incident on the telescope entrance aperture.

N is the number of spectral pixels (1 for broadband detectors).

$\lambda^-(N)$, $\lambda^+(N)$ give (in \AA) the lower and upper wavelength limits for each spectral pixel of the detector. The throughput of each pixel is assumed to vanish outside these limits.

The error in calculating the numerical integrals specified below must never be larger than 0.1 %. Accordingly, the wavelength sets should be finely sampled so that linear interpolation represents with the required accuracy the passband and spectrum at intermediate wavelengths. If a nonlinear interpolation scheme is used, care must be taken to ensure that the interpolated values are nowhere negative.

UNITFLAM($D, M, \lambda(M), P(M), N, \lambda^-(N), \lambda^+(N), \text{output}$) The inverse sensitivity,

$$U_\lambda(i) = \left[\frac{\pi D^2}{4hc\Lambda} \int_{\lambda^-(i)}^{\lambda^+(i)} P(\lambda)\lambda d\lambda \right]^{-1},$$

is computed for each pixel i . U_λ is the flux density in $\text{erg cm}^{-2}\text{s}^{-1}\text{\AA}^{-1}$ that is required to produce a unit response of $1 \text{ count s}^{-1}\text{pixel}^{-1}$, and is used to convert an observed count rate C to the corresponding flux density: $f_\lambda = CU_\lambda$.

UNITSTMAG($D, M, \lambda(M), P(M), N, \lambda^-(M), \lambda^+(M), \text{output}$) The reference ST magnitude,

$$STUNIT(i) = -2.5 \log [U_\lambda(i)] - ST_0,$$

is calculated for each pixel i . $STUNIT$ is the ST magnitude required to produce a unit response of $1 \text{ count s}^{-1}\text{pixel}^{-1}$, and is used to convert an observed count rate C to the corresponding ST magnitude: $ST = STUNIT - 2.5 \log C$.

UNITFNU($D, M, \lambda(M), P(M), N, \lambda^-(M), \lambda^+(M), \text{output}$) The inverse sensitivity,

$$U_\nu(i) = \left[\frac{\pi D^2}{4h} \int_{\lambda^-(i)}^{\lambda^+(i)} P(\lambda) \frac{d\lambda}{\lambda} \right]^{-1},$$

is computed for each pixel i . U_ν is the flux density in $\text{erg cm}^{-2}\text{s}^{-1}\text{Hz}^{-1}$ that is required to produce a unit response of $1 \text{ count s}^{-1}\text{pixel}^{-1}$, and is used to convert an observed count rate C to the corresponding flux density: $f_\nu = CU_\nu$.

UNITSTMAG($D, M, \lambda(M), P(M), N, \lambda^-(M), \lambda^+(M), \text{output}$) The reference AB magnitude,

$$ABUNIT(i) = -2.5 \log [U_\nu(i)] - AB_0,$$

is computed for each pixel i . $ABUNIT$ is the AB magnitude that is required to produce a unit response of $1 \text{ count s}^{-1}\text{pixel}^{-1}$, and $ABUNIT$ is used to convert an observed count rate C to the corresponding AB magnitude: $AB = ABUNIT - 2.5 \log C$.

PIVLAM($M, \lambda(M), P(M), N, \lambda^-(M), \lambda^+(M), \text{output}$) The pivot wavelength,

$$\lambda_P(i) = \sqrt{\frac{\int_{\lambda^-(i)}^{\lambda^+(i)} P(\lambda)\lambda d\lambda}{\int_{\lambda^-(i)}^{\lambda^+(i)} P(\lambda)d\lambda/\lambda}},$$

is computed for each pixel i . The pivot wavelength used for exact conversions between flux densities based on ν and λ , as in

$$f_\nu = \frac{\lambda_P^2}{c\Lambda} f_\lambda.$$

BARLAM(M , $\lambda(M)$, $P(M)$, N , $\lambda^-(M)$, $\lambda^+(M)$, **output**) The bar wavelength,

$$\bar{\lambda}(i) = \exp \left\{ \frac{\int_{\lambda^-(i)}^{\lambda^+(i)} \ln(\lambda) P(\lambda) \frac{d\lambda}{\lambda}}{\int_{\lambda^-(i)}^{\lambda^+(i)} P(\lambda) \frac{d\lambda}{\lambda}} \right\},$$

is computed for each pixel i . This is a robust characteristic wavelength that is defined in the WF/PC Instrument Handbook.

RMSLAM(M , $\lambda(M)$, $P(M)$, N , $\lambda^-(M)$, $\lambda^+(M)$, **output**) The effective Gaussian RMS bandwidth,

$$\lambda_{\text{RMS}}(i) = \bar{\lambda} \sqrt{\frac{\int_{\lambda^-(i)}^{\lambda^+(i)} \left[\ln \left(\frac{\lambda}{\bar{\lambda}} \right) \right]^2 P(\lambda) \frac{d\lambda}{\lambda}}{\int_{\lambda^-(i)}^{\lambda^+(i)} P(\lambda) \frac{d\lambda}{\lambda}}},$$

is computed for each pixel i .

FWHMLAM(M , $\lambda(M)$, $P(M)$, N , $\lambda^-(M)$, $\lambda^+(M)$, **output**) The effective Gaussian full-width-half-maximum bandwidth,

$$\lambda_{\text{FWHM}}(i) = \sqrt{8 \ln(2)} \lambda_{\text{RMS}},$$

is computed for each pixel i .

16.1.2 Properties of Passbands and Spectra

This subsection covers utilities that compute quantities involving both passbands and spectra, including predicted count rates, mean flux densities, and effective wavelengths.

RATEFLAM(D , M , $\lambda(M)$, $P(M)$, $f_{\lambda}(M)$, N , $\lambda^-(M)$, $\lambda^+(M)$, **output**) The predicted count rate,

$$C^{\text{pred}}(i) = \frac{\pi D^2}{4hc\Lambda} \int_{\lambda^-(i)}^{\lambda^+(i)} f_{\lambda}(\lambda) P(\lambda) \lambda d\lambda,$$

is computed for each pixel i from the passband and an f_{λ} spectrum.

RATEFNU(D , M , $\lambda(M)$, $P(M)$, $f_{\nu}(N)$, N , $\lambda^-(M)$, $\lambda^+(M)$, **output**) The predicted count rate,

$$C^{\text{pred}}(i) = \frac{\pi D^2}{4h} \int_{\lambda^-(i)}^{\lambda^+(i)} f_{\nu}(\lambda) P(\lambda) \frac{d\lambda}{\lambda},$$

is computed for each pixel i from the passband and an f_{ν} spectrum.

AVGFLAM(D , M , $\lambda(M)$, $P(M)$, $f_{\lambda}(M)$, N , $\lambda^-(M)$, $\lambda^+(M)$, **output**) Average flux density,

$$f_{\lambda}^{\text{pred}}(i) = \frac{\int_{\lambda^-(i)}^{\lambda^+(i)} f_{\lambda}(\lambda) P(\lambda) \lambda d\lambda}{\int_{\lambda^-(i)}^{\lambda^+(i)} P(\lambda) \lambda d\lambda},$$

is computed for each pixel i from the passband and an f_λ spectrum.

AVGFNU($D, M, \lambda(M), P(M), f_\nu(N), N, \lambda^-(M), \lambda^+(M), \text{output}$) Average flux density,

$$f_\nu^{\text{pred}}(i) = \frac{\int_{\lambda^-(i)}^{\lambda^+(i)} f_\nu(\lambda) P(\lambda) \frac{d\lambda}{\lambda}}{\int_{\lambda^-(i)}^{\lambda^+(i)} P(\lambda) \frac{d\lambda}{\lambda}},$$

is computed for each pixel i from the passband and an f_ν spectrum.

EFFLAMFLAM($D, M, \lambda(M), P(M), f_\lambda(M), N, \lambda^-(M), \lambda^+(M), \text{output}$) The effective wavelength,

$$\lambda_{\text{eff}} = \frac{\int_{\lambda^-(i)}^{\lambda^+(i)} f_\lambda(\lambda) P(\lambda) \lambda^2 d\lambda}{\int_{\lambda^-(i)}^{\lambda^+(i)} f_\lambda(\lambda) P(\lambda) \lambda d\lambda},$$

is calculated for each pixel i from the passband and an f_λ spectrum.

EFFLAMFNU($D, M, \lambda(M), P(M), f_\nu(N), N, \lambda^-(M), \lambda^+(M), \text{output}$) The effective wavelength,

$$\lambda_{\text{eff}} = \frac{\int_{\lambda^-(i)}^{\lambda^+(i)} f_\nu(\lambda) P(\lambda) d\lambda}{\int_{\lambda^-(i)}^{\lambda^+(i)} f_\nu(\lambda) P(\lambda) \frac{d\lambda}{\lambda}},$$

is calculated for each pixel i from the passband and an f_ν spectrum.

16.1.3 Magnitudes and Units Conversions

The following subroutines provide for easy conversion of observations among various spectral flux densities and magnitude systems that are in common use. Each routine converts a set of N observations, which might be different pixels of a spectrum, or a set of broadband filter observations. Pivot wavelengths should be supplied as input to those routines that perform conversions between f_λ and f_ν (see the definition of **PIVLAM** above), for otherwise the conversion will not be exact.

FLAMTOFNU($N, \lambda_P(N), f_\lambda(N), \text{output}$)

$$f_\nu(i) = \frac{\lambda_P^2(i)}{c\Lambda} f_\lambda(i)$$

FLAMTOMJY($N, \lambda_P(N), f_\lambda(N), \text{output}$)

$$f_{\text{mJy}}(i) = 10^{26} \frac{\lambda_P^2(i)}{c\Lambda} f_\lambda(i)$$

FLAMTOSTMAG($N, f_\lambda(N)$, output)

$$ST(i) = -2.5 \log [f_\lambda(i)] - ST_0$$

FLAMTOABMAG($N, \lambda_P(N), f_\lambda(N)$, output)

$$AB(i) = -2.5 \log \left[\frac{\lambda_P^2(i)}{c\Lambda} f_\lambda(i) \right] - AB_0$$

FNUTOFLAM($N, \lambda_P(N), f_\nu(N)$, output)

$$f_\lambda(i) = \frac{c\Lambda}{\lambda_P^2(i)} f_\nu(i)$$

FNUTOMJY($N, f_\nu(N)$, output)

$$f_{mJy}(i) = 10^{26} f_\nu(i)$$

FNUTOSTMAG($N, \lambda_P(N), f_\nu(N)$, output)

$$ST(i) = -2.5 \log \left[\frac{c\Lambda}{\lambda_P^2(i)} f_\nu(i) \right] - ST_0$$

FNUTOABMAG($N, f_\nu(N)$, output)

$$AB(i) = -2.5 \log [f_\nu(i)] - AB_0$$

MJYTOFLAM($N, \lambda_P(N), f_{mJy}(N)$, output)

$$f_\lambda(i) = 10^{-26} \frac{c\Lambda}{\lambda_P^2(i)} f_{mJy}(i)$$

MJYTOFNU($N, f_{mJy}(N)$, output)

$$f_\nu(i) = 10^{-26} f_{mJy}(i)$$

MJYTOSTMAG($N, \lambda_P(N), f_{mJy}(N)$, output)

$$ST(i) = -2.5 \log \left[10^{-26} \frac{c\Lambda}{\lambda_P^2(i)} f_{mJy}(i) \right] - ST_0$$

MJYTOABMAG($N, f_{mJy}(N)$, output)

$$AB(i) = -2.5 \log [10^{-26} f_{mJy}(i)] - AB_0$$

STMAGTOFLAM($N, ST(N)$, output)

$$f_\lambda(i) = 10^{-0.4(ST(i) + ST_0)}$$

STMAGTOFNU($N, \lambda_P(N), ST(N), \text{output}$)

$$f_\nu(i) = \frac{\lambda_P^2(i)}{c\Lambda} 10^{-0.4(ST(i) + ST_0)}$$

STMAGTOMJY($N, \lambda_P(N), ST(N), \text{output}$)

$$f_{\text{mJy}}(i) = 10^{26} \frac{\lambda_P^2(i)}{c\Lambda} 10^{-0.4(ST(i) + ST_0)}$$

STMAGTOABMAG($N, \lambda_P(N), ST(N), \text{output}$)

$$AB(i) = ST(i) - AB_0 + ST_0 - 2.5 \log \left[\frac{\lambda_P^2(i)}{c\Lambda} \right]$$

ABMAGTOFLAM($N, \lambda_P(N), AB(N), \text{output}$)

$$f_\lambda(i) = \frac{c\Lambda}{\lambda_P^2(i)} 10^{-0.4(AB(i) + AB_0)}$$

ABMAGTOFNU($N, AB(N), \text{output}$)

$$f_\nu(i) = 10^{-0.4(AB(i) + AB_0)}$$

ABMAGTOMJY($N, AB(N), \text{output}$)

$$f_{\text{mJy}}(i) = 10^{-26} 10^{-0.4(AB(i) + AB_0)}$$

ABMAGTOSTMAG($N, \lambda_P(N), AB(N), \text{output}$)

$$ST(i) = AB(i) + AB_0 - ST_0 - 2.5 \log \left[\frac{c\Lambda}{\lambda_P^2(i)} \right]$$

16.2 Throughput Calibration Databases and Supporting Software

The throughput calibration program based on synthetic photometry requires four databases:

1. component throughput functions,
2. the observatory graph,
3. calibration target spectra,
4. calibration observations.

This section describes the purpose and contents of these databases and the associated software tools that provide for accessing, updating and verifying their contents.

16.2.1 Component Throughput Functions

The components database provides throughput functions for every optical component in the HST. These components include mirrors, windows, apertures, lenses, polarizers, filters, gratings, prisms, detectors, and any other parameters that affect the sensitivity. The throughputs are dimensionless; they represent at each wavelength the number of photons exiting from the component (or the number of counts registered in the case of a detector) per photon incident upon the component.

Each member of the database corresponds to a specific component, and gives:

M , the number of wavelengths.

$\lambda(M)$, the wavelength set, in \AA (different wavelength sets may be used for different components).

$P(M)$, the throughput at each wavelength.

$\sigma_P(M)$, the throughput uncertainty at each wavelength.

comments documenting origin of the data.

Expert input is required to fill the components database. The database will be filled initially with throughputs from laboratory measurements, manufacturers specifications, or theoretical calculations. The throughput functions will be revised and new components will be installed in the database in order to ensure that predicted count rates agree with the count rates observed on HST calibration targets.

Software required to access, update and verify the components database:

LOADFILT(M , $\lambda(M)$, $P(M)$, $\sigma_P(M)$, **NAME**) Access throughput data on its original wavelength set.

Input: components database, name of component, maximum number of wavelengths M .

Output: number of wavelengths M , wavelengths $\lambda(i)$, throughputs $P(i)$, uncertainties $\sigma_P(i)$.

EVALFILT(M , $\lambda(M)$, $P(M)$, $\sigma_P(M)$, **NAME**) Access throughput data at user-specified wavelengths.

Input: components database, name of component, number of wavelengths M , wavelengths $\lambda(i)$.

Procedure: use LOADFILT to access throughput data, interpolate to desired wavelengths.

Output: throughputs $P(i)$ and uncertainties $\sigma_P(i)$ interpolated to the requested wavelengths.

NEWFILT: Install new filter in the database.

Input: components database, name of component, wavelengths, throughputs, uncertainties.

Procedure: abort if proposed name matches existing component, create new member, record of activity in database history log.

Output: new member in the database, record of activity in database history log.

EDITFILT: Update (create a revised version of) an existing member.

Input: components database, name of component, wavelengths, throughputs, uncertainties.

Output: revised version of member in the database, record of activity in database history log.

PLOTFILT: Plot throughput data and uncertainty against wavelength.

Input: components database, list of component names.

Output: plot of $P(\lambda)$ and $P(\lambda) \pm \sigma_P(\lambda)$, with vertical line marking pivot wavelength λ_p , for each specified component.

16.2.2 The Observatory Graph and Dynamic Throughput Generator

The observatory graph database provides the information needed to determine for a specified observing mode which optical components are present along the optical path. The graph may be visualized as a set of *nodes* that are connected together by a network of *links*. Each node represents a place along one of the many possible paths that light can take through the HST observatory. Each link represents a segment of the optical path starting from one node, designated the *entry node*, through a specific optical component, and on to the next node, the *exit node*. The graph is thus somewhat like a tree, except that different paths through the observatory graph can reconnect, while the branches of a tree, by definition, do not.

Each member of the observatory graph database represents one of the links, and gives the following quantities:

entry node number,
exit node number.
pointer to a member of the components database,
keyword.
comments.

The entry and exit node numbers identify the nodes that are connected by this link, and the direction in which the light passes. The pointer associates this link with the components database member that provides the throughput data for the optical component on this segment of the light path. The keywords attached to the links are used to trace the light path through the observatory graph. A path that arrives at the entry node can proceed to the exit node if the keyword associated with the link matches one of the keywords in the list that specifies the observing mode. Expert input is needed to fill the observatory graph database.

Software required to access, update and verify the observatory graph database:

PATHFINDER: uses mode keywords to trace a path through the graph.

Input: graph database, character string with keywords specifying observing mode.

Procedure: extract mode keywords from character string,
load graph links, sort links by entry node number,
start at smallest entry node number,
find all possible paths with link keyword matching a mode keyword,
abort if path is ambiguous,
move to exit node if path is unique,
move to exit node along default path,
finish if trapped.

Output: List of pointers to members of components database.

EVALBAND(M , $\lambda(M)$, $P(M)$, $\sigma_P(M)$, **MODE)** Generates the passband throughput function for a specified mode. This is the **dynamic throughput generator**, which has widespread application outside CDBS and hence should be made available to such programs as SDAS, RSDP, RPSS, PEPSI exposure time calculators, and other stand-alone user programs.

Input: components database, observatory graph database, character string **MODE** with keywords specifying the observing mode, number of wavelengths M , wavelength set $\lambda(M)$ on which the throughput is to be evaluated.

Procedure: Send mode keywords to **PATHFINDER** to trace a path through the observatory graph and thereby to generate pointers to the required component throughput data.

Use **EVALFILT** to load the throughput data for each component j and to interpolate to the required wavelength set $\lambda(i)$, thereby obtaining $P_j(i)$ and $\sigma_{P_j}(i)$.

Compute for each wavelength $\lambda(i)$ the corresponding total throughput for the mode,

$$P(i) = \prod_j P_j(i).$$

Compute for each wavelength $\lambda(i)$ the corresponding throughput uncertainty,

$$\sigma_P(i) = P(i) \sqrt{\sum_j \left[\frac{\sigma_{P_j}(i)}{P_j(i)} \right]^2}.$$

This formula assumes that fractional throughput uncertainties are small, and that the throughput errors of different components are independent.

Output: Mode throughput function, $P(i)$,
uncertainties, $\sigma_P(i)$ at the requested wavelengths.

NEWLINK: Install a new link in the graph.

Input: graph database, components database, entry node, exit node, pointer, keyword

Procedure: abort if proposed entry node and keyword are identical with an existing link.
abort if pointer misses all members of components database,
create new link in the graph database,
record activity in database history log.

Output: new member in the database, record of activity in database history log.

BADLINK: Delete an existing link from the graph.

Input: graph database, entry node, exit node, keyword

Procedure: load graph links,
find all links with specified entry mode,
exit mode, and keyword
abort if 0 or 1 link found,
delete the specified link,
record activity in database history log.

Output: member removed from the database, record of activity in database history log.

Test for completeness of the throughput data:

Input: graph database, components database.

Output: list (ordered by entry node) of all graph links that point to components that are absent from the components database.

Test for ambiguous paths: (The process that installs new links is designed to prevent this.)

Input: graph database

Output: list (ordered by entry node) of all graph links that have identical entry nodes and keywords

LISTGRAPH: Makes a listing of the graph.

Input: graph database, starting node number.

Output: listing of all graph links or of those links that are “downstream” from the specified node.

EDITGRAPH: Interactive graph editor.

Input: graph database, components database.

Procedure: interactive program allowing a user to
view on terminal screen the forward and backward links from any given node,
move forward or backward through the instrument graph,
jump to a distant node,
plot component throughput functions,
run the graph testing procedures defined above,
install and delete links.

Output: possible hardcopy plots, lists from graph test procedures, new members or members removed from the database, record of activity in database history log.

PLOTGRAPH: Makes a picture of the observatory graph. (useful for verification purposes but not absolutely necessary.)

Input: graph database, starting node number.

Output: a graphical representation of the entire graph or of all links that are “downstream” from the specified node.

16.2.3 Calibration Target Spectra

The spectrum database provides for each of the HST calibration targets the energy flux density as a function of wavelength. These spectra are used to compute predicted count rates whenever a calibration target is observed. Note that spectra here are *not* the same as the spectrophotometric observations that will be present for many of the calibration targets in the observations database.

Each member of the spectrum database gives the spectrum and other information about one of the calibration targets:

M number of wavelengths

$\lambda(M)$ array of wavelengths.

$ST(M)$ the ST magnitude at each wavelength.

$\sigma_{ST}(M)$ uncertainty in the ST magnitude at each wavelength.

Other optional information (such as epoch, right ascension, declination, proper motion, radial velocity)

comments.

Expert input is required to define the wavelength sets and the ST magnitudes for each target. Different wavelength sets may be used for different targets, and the wavelength spacing need not be uniform, but will be everywhere fine enough that linear interpolation suffices to represent the spectrum at intermediate wavelengths. The wavelength sets and ST magnitudes will be updated as knowledge of the calibration target spectra improves. New calibration targets will be installed.

Software required to access, update and verify the spectrum database is very similar to that described in Section 16.2.1 for the components database.

LOADSPEC(M , $\lambda(M)$, $ST(M)$, $\sigma_{ST}(M)$, **TARGET**) Access spectra on their original wavelength set.

Input: spectrum database, name of target, maximum number of wavelengths M .

Output: number of wavelengths M , wavelengths $\lambda(i)$, ST magnitudes $ST(i)$, uncertainties $\sigma_{ST}(i)$.

EVALSPEC(M , $\lambda(M)$, $ST(M)$, $\sigma_{ST}(M)$, **TARGET**) Access spectra at user-specified wavelengths.

Input: spectrum database, name of target, number of wavelengths M , wavelengths $\lambda(i)$.

Procedure: access spectrum data and uncertainty, interpolate to desired wavelengths.

Output: ST magnitudes $ST(i)$ and uncertainties $\sigma_{ST}(i)$ interpolated to the specified wavelengths.

NEWSPEC: Install a new spectrum in the database.

Input: spectrum database, name of new spectrum, wavelengths, ST magnitudes, uncertainties, and other relevant information.

Procedure: abort if proposed name matches existing component, install the new spectrum, record of activity in database history log.

Output: new member in the database, record of activity in database history log.

EDITSPEC: Update (create a new version of) an existing spectrum.

Input: spectrum database, name of target, wavelengths, ST magnitudes, uncertainties.

Output: new version of member in the database, record of activity in database history log.

PLOTSPEC: Plot spectrum data and uncertainty against wavelength.

Input: spectrum database, list of targets.

Output: plot of $ST(\lambda)$ and $ST(\lambda) \pm \sigma_{ST}(\lambda)$ for each specified target.

16.2.4 Calibration Observations and the Comparison of Observed and Predicted Responses

The calibration observations database will provide access to fully-reduced observations of the calibration targets. This database will reference ground-based measurements and measurements by the IUE as well as measurements made with the HST. It will include both spectrophotometric measurements, with high spectral resolution, and broadband photometric measurements, with low spectral resolution. The main purpose of the database is to provide a uniform method of accessing these diverse observations.

Each member of the observations database corresponds to an observation of one of the calibration targets, and provides access in a uniform way to at least the following information:

unique calibration observation number

date and time of entry into the database

TARGET, the name of the calibration target

MODE, character string with keywords that specify the observing mode

other parameters that may affect the throughput

date and time at midpoint of the exposure (heliocentric ephemeris time)

t , the exposure time or dwell (sec)

N , the number of spectral pixels (1 for broadband measurements)

$\lambda^-(N)$, $\lambda^+(N)$, the wavelength limits (\AA) for each pixel (can be *e.g.*, 0.999999 for a broadband mode).

$C(N)$, the observed responses (*counts s₋₁pixel⁻¹*) in each pixel, fully corrected for instrumental effects (*e.g.*, background, airmass, flat field, nonlinearity) and integrated in a standard way over spatial dimensions of the detector.

$\sigma_C(N)$, the estimated uncertainty of the observed count rates (1- σ)

Expert input will be required to populate the database with fully-reduced calibration target observations.

Software required to access, update and verify the calibration observations database:

PREPOBS: Prepare a fully-reduced observation (from RSDP, SDAS or other source) for insertion into the database.

Input: a reduced observation (from RSDP, SDAS, or other source) fully corrected for instrumental effects (*e.g.*, background, airmass, flat field, nonlinearity) and integrated in a standard way over spatial dimensions of the detector.

Output: all attributes needed to form an entry in the observations database (target name, mode specification, observation date and time, dwell, number of pixels, total count and uncertainty for each pixel).

NEWOBS: Install the new observation in the database.

Input: components database, observatory graph database, spectrum database, calibration observations database, all attributes of the new observation (except installation date and time).

Procedure: abort if dwell is not positive,
abort if proposed calibration observation number is already in use,
abort if target is not in spectrum database,
abort if mode keywords fail to generate passband function
abort if predicted count rate is not positive in every pixel
install new observation in the database,
record of activity in database history log.

Output: new member in the database, record of activity in database history log.

EDITOBS: Correct (create a revised version) of an observation in the database.

Input: components database, observatory graph database, spectrum database, calibration observations database, all attributes of the revised observation (except installation date and time).

Procedure: abort if dwell is not positive,
abort if requested calibration observation number is not present in the database,
abort if target is not in spectrum database,
abort if mode keywords fail to generate passband function
abort if predicted count rate is not positive in every pixel
install revised observation in the database,
record of activity in database history log.

Output: revised version of member in the database, record of activity in database history log.

SELECTOBS: Select subsets of the calibration observations.

Input: calibration observations database, criteria for selecting observations (ranges, lists, or wildcards giving required or excluded values for attributes of the observations, e.g., range of epochs, list of targets names, specific mode keywords)

Output: merged collection of all calibration observations satisfying the selection criteria.

THRUPUTCAL Generates plots and tabular reports that compare observed and predicted responses to the calibration targets. These will be used to monitor the throughput calibration of the HST and to generate updates for the components and observatory graph databases.

Input: components database, observatory graph, spectrum database, observations database, criteria for selecting observations (ranges, lists, or wildcards giving required or excluded values for attributes of the observations, *e.g.*, range of epochs, list of targets names, specific mode keywords)

Procedure: Select requested observations. Generate corresponding passbands and retrieve the target spectra. Compute for each pixel of each observation the sensitivity, observed and predicted count rate, observed and predicted flux density, pivot wavelength, effective wavelength, FWHM bandwidth. Sort observations as required. Create plots and tabular reports of results.

Output: Optional plots and tabular reports comparing observed and predicted responses.

Plot giving the target spectrum f_λ (or ST magnitude) vs λ and overlain with points giving for each pixel of each observation the observed flux density (or ST magnitude) \pm uncertainty vs $\lambda_{\text{eff}} \pm \frac{1}{2}\lambda_{\text{FWHM}}$.

A plot with uncertainty bars of the ratio of observed to predicted response vs effective wavelength, vs time, vs specified spectral index of source, or vs other specified or calculated attributes of the observations.

Tabular report summarizing the selected observations by giving for each observation the observation number, date and time, mode, target name, dwell, and for each pixel the pixel number, λ_p , λ_{FWHM} , predicted ST magnitude and uncertainty, observed ST magnitude and uncertainty, the ratio observed/predicted and uncertainty.