NICMOS Data Formats in OPUS and STSDAS

John W. MacKenty, Howard Bushouse, Chris Skinner
July 25, 1995

ABSTRACT
In this ISR the contents and structure of the NICMOS science data files generated by the STScI ground system OPUS pipeline are described. NICMOS data differ from the data from prior HST science instruments in two ways. First, for most observing modes, the data are partially processed within NICMOS (using its autonomous computer) before the data are sent to the Science Data Formatter, stored onboard HST, and downlinked to the ground. Second, the STScI ground system structures the data into a single format which is used for all observing modes and which remains the same at each step along the data calibration process. Therefore, while the user of NICMOS data should understand the consequences of the different onboard processing options, all data are represented and stored in a common format and software can access them in a simple and consistent manner.

1. Introduction
Each HST Science Instrument has had an specific format for its datasets. The original complement of SIs and the WFPC2 used a hybrid of GEIS format within the STScI ground (PODPS and OPUS) and STSDAS systems and FITS for archiving and distribution. With the introduction of two new, and complex, SIs as part of the Second Servicing Mission, a single data format using FITS Image Extensions is being adopted. This ISR defines the realization of the NICMOS data format.

NICMOS data are stored in FITS format datasets. There is one dataset (or file) for each exposure obtained with NICMOS. By “exposure” we mean a single SI activity that results in a dataset with a unique IPPPSSOOT. An exposure may contain multiple “readouts” of the detector arrays in the case of MULTI-ACCUM mode exposures.

In this ISR we define the elements of the NICMOS standard data format, consider the nature of the elements for each observing mode, and provide size estimates of the resulting datasets.
2. Standard Data Format

Structure
NICMOS data are represented by five arrays for each “readout”. These arrays contain

- Science Image
- Error Array
- Quality Flags Array
- Samples Array
- Integration Time Array

Each downlinked readout is always represented by these five arrays. Multiple readouts in the same dataset (e.g. MULTI-ACCUM data) are represented by repeated sets of these five arrays. It is expected that compact FITS representations will be available to store arrays in which all elements have the same value.

Individual Arrays

Science Image
The Science Image contains the data from the focal plane array (FPA) detectors. These may be represented as counts (i.e. data numbers) or as count rates (i.e. data numbers per second). Generally the later is desirable since it is easier to interpret in mosaiced datasets and corresponds closely to flux.

Error Array
The Error Array contains an estimate of the statistics error at each pixel. It is expressed as a real number of standard deviations. This is a calculated quantity based on a model of the instrument and its environment.

Quality Flags Array
The Quality Flags Array provides 16 independent flags for each pixel. Each flag has a true (set) or false (unset) state and is encoded as a bit in a 16 bit (short int) word. Users are advised that this word should NOT be interpreted as an integer. The provisional list of NICMOS data quality flags is given in Table 1.

Samples Array
The Samples Array is used for either of two purposes as appropriate for the situation. For data where multiple samples of the array were obtained during the integration, the Samples Array denotes the number of samples for each pixel in the corresponding Science
Image. When multiple integrations are combined to produce a single image, the Samples Array will contain the number of samples retained at each pixel. Note this implies that the original number of samples information is not propagated forward into combined images.

**Integration Time Array**

The Integration Time Array contains the total integration time at each pixel. While a simple parameter initially in some observing modes, the combination of datasets into composite mosaics and the utilization of the information obtained by multiple non-destructive readouts during an integration requires us to keep track of the actual “exposure” time for each pixel. This array is useful for simple conversions between counts and count rates.

**Table 1. Provision List of NICMOS Data Quality Flags**

<table>
<thead>
<tr>
<th>Flag</th>
<th>Quality Indicated</th>
<th>Flag</th>
<th>Quality Indicated</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reed-Solomon Error</td>
<td>9</td>
<td>Bad Pixel</td>
</tr>
<tr>
<td>2</td>
<td>Poor Linearity Correction</td>
<td>10</td>
<td>Pixel contains Cosmic Ray</td>
</tr>
<tr>
<td>3</td>
<td>Poor Dark Correction</td>
<td>11</td>
<td>Pixel contains Source</td>
</tr>
<tr>
<td>4</td>
<td>Poor Flat Field Correction</td>
<td>12</td>
<td>Unassigned</td>
</tr>
<tr>
<td>5</td>
<td>Poor Background Correction</td>
<td>13</td>
<td>User Flag</td>
</tr>
<tr>
<td>6</td>
<td>Defective Pixel</td>
<td>14</td>
<td>User Flag</td>
</tr>
<tr>
<td>7</td>
<td>Saturated Pixel</td>
<td>15</td>
<td>Reserved</td>
</tr>
<tr>
<td>8</td>
<td>Missing Data</td>
<td>16</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

**IRAF Access**

This data structure will be implemented so that the default array is the most meaningful Science Image array. For example, if the user requests IRAF to display a dataset, the first Science Image array would be displayed. Access to the other arrays within a dataset will be via element numbers (1-5; mod 5 for datasets containing multiple readouts) or via a standardized naming convention. It is a critical requirement that existing IRAF tasks be able to access any array in a NICMOS dataset.

**3. Data Format Details for Each Observing Mode**

**Uncalibrated Datasets (post Generic Conversion)**

The data returned from the NICMOS to the STScI ground system will contain a single image array except for the MULTI-ACCUM datasets which will contain up to 25 image arrays. In Table 2 arrays where all pixels will contain the same value after generic conver-
sion are indicated with an “*”. The initial error arrays are indicated with a zero because all of these values are set to INDEF initially. Unlike the sample and integration arrays, they do not have a meaningful initial value.

Table 2. Uncalibrated Dataset Contents

<table>
<thead>
<tr>
<th>SI Mode</th>
<th>Image</th>
<th>Error</th>
<th>Quality</th>
<th>Samples</th>
<th>Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCUM</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1*</td>
<td>1*</td>
</tr>
<tr>
<td>MULTI-ACCUM</td>
<td>N</td>
<td>0</td>
<td>N</td>
<td>N*</td>
<td>N*</td>
</tr>
<tr>
<td>RAMP</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1*</td>
</tr>
<tr>
<td>BRIGHTOBJ</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1*</td>
<td>1*</td>
</tr>
<tr>
<td>ACQ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Partial Calibrated Datasets (post single dataset processing)

After processing the individual datasets (the step currently known as CALNICA), the structure of the files does not change but a considerable amount of information is added to each dataset as shown in Table 3.

Table 3. Partially Calibrated Dataset Contents

<table>
<thead>
<tr>
<th>SI Mode</th>
<th>Image</th>
<th>Error</th>
<th>Quality</th>
<th>Samples</th>
<th>Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCUM</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1*</td>
<td>1*</td>
</tr>
<tr>
<td>MULTI-ACCUM</td>
<td>N</td>
<td>1</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>RAMP</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>BRIGHTOBJ</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1*</td>
<td>1*</td>
</tr>
<tr>
<td>ACQ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fully Calibrated Datasets (associated dataset processing)

Finally, after combining the members of an association of individual datasets (the step currently known as CALNICB), the structure of the files does not change but additional information is added to each dataset (see Table 4). At this point the individual character of each observing mode is basically gone. For example, an association might contain both RAMP and ACCUM mode observations which are intended to be mosaiced together (*n.b. this is not supported by the SI commanding and thus not available within OPUS but might be done off-line by an observer*). At this stage the full dimensionality of the data format is utilized yet at any stage in the processing all elements of the data structure are both identical and meaningful to software accessing them.
Table 4. Fully Calibrated Dataset Contents

<table>
<thead>
<tr>
<th>SI Mode</th>
<th>Image</th>
<th>Error</th>
<th>Quality</th>
<th>Samples</th>
<th>Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANY</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**Notes on Each Observing Mode**

**ACCUM**

Each observation yields a single image. The error array is initially undefined and must be populated during the first stage of calibration processing. The quality array after generic conversion contains flags for Reed-Solomon and Missing Data. About 50% of the flags are potentially set after the first stage of calibration processing. The samples array values are always one for ACCUM mode (unless this array is subsequently updated to reflect “useless” pixels for which the number of samples would become zero -- TBD). The integration array contains the same exposure time for each pixel.

**RAMP**

Each observations yields a single image accompanied by an error and samples array. The quality array after generic conversion contains flags for Reed-Solomon and Missing Data. About 50% of the flags are potentially set after the first stage of calibration processing. The integration time array will be calculated from an understanding of the parameters used by the onboard ramp mode software.

**MULTI-ACCUM**

The MULTI-ACCUM format has the identical structure as the other modes’ formats but is repeated for each readout. For each readout (N < 25) an image array is produced and downlinked. It is ordered so that the default image extension (group) contains the longest integration time. Hence the image extensions are ordered in the opposite sense from which they are obtained.

After generic conversion a quality array containing flags for Reed-Solomon and Missing Data exists for each readout (*i.e.* N quality arrays). About 50% of these flags are potentially set after the first stage of calibration processing. For uncalibrated data the sample number for each readout is that readout’s number (starting with zero for the readout immediately following the FPA reset). The integration time is the cumulative integration time up to the time of that readout. After first stage calibration the error array will be populated and the quality, samples, and integration arrays updated (*i.e.* these arrays will no longer contain constant values).
**BRIGHTOBJ**

Bright Object mode data are the same as ACCUM mode data.

**ACQ**

The treatment of acquisition mode data is TBD.

### 4. Size Estimates of Data Products

The NICMOS arrays are 256 x 256 pixels in size. In Table 5 we calculate the sizes of the datasets (in units of $10^6$ bytes) for each stage of the processing and for each observing mode. We have ignored the overhead associated with the image headers. We have assumed that the FITS representation for arrays of constant value will be essentially of zero size. We also assume that the Image, Error, and Integration require 4 byte words and that the Quality and Samples array require 2 byte words.

For Multi-Accum observations the sizes are provided with the appropriate factor of N included. The worst case NICMOS Multi-Accum dataset (25 readouts) would require 10 MB after generic conversion and 27 MB after First Stage calibration.

**Table 5. Size of NICMOS Datasets**

<table>
<thead>
<tr>
<th>Mode</th>
<th>Uncalibrated</th>
<th>First Stage</th>
<th>Second Stage</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCUM</td>
<td>0.39</td>
<td>0.66</td>
<td>1.05</td>
</tr>
<tr>
<td>MULTI-ACCUM</td>
<td>0.39 * (N+1)</td>
<td>1.05 * (N+1)</td>
<td>1.05</td>
</tr>
<tr>
<td>RAMP</td>
<td>0.79</td>
<td>1.05</td>
<td>1.05</td>
</tr>
<tr>
<td>BRIGHTOBJ</td>
<td>0.39</td>
<td>0.66</td>
<td>1.05</td>
</tr>
<tr>
<td>ACQ</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>