

# Стандартная система редукции астрономических данных MIDAS

## Лекция IV. Работа с различными структурами данных

А.Ю. Князев

Специальная астрофизическая обсерватория РАН, п.Нижний Архыз, 357147, Россия

*Но приготовься, чтобы не поразила тебя слишком эта премудрость, ибо от нее легко потерять рассудок — настолько она поразительна, ослепительна и необъятна.*

*Леонид Соловьев. "Повесть о Ходже Насреддине"*

Я думаю, что уже на основе разбора примеров из предыдущей лекции Вы почувствовали, что вдумчивая работа с системой редукции типа MIDAS требует от пользователя определенной логики поведения, сильно отличающейся от логики программирования, скажем, на языках типа FORTRAN или С. Я бы назвал этот стиль UNIX-подобным. Это, конечно, не чистое сравнение — трудно напрямую сравнивать UNIX, скажем, с MIDAS. Но, скажем, если перед продвинутым UNIX-пользователем встает проблема отсутствия какого-то средства для решения небольшой проблемы, то он быстренько-быстренько напишет новый командный файл, используя множество из 600 дополнительных команд этой операционной системы и НИКОГДА не будет писать новой программы на С для выполнения задуманного.

**Замечание :** Следствие этой аксиомы — если Вы увидели, что человек программирует на С в UNIX, то это либо "зеленый" пользователь, понятия не имеющий, что в UNIX есть команды **grep** и **find**, либо "крутой" системщик, у которого, скажем, монитор упал на бок, а он решил заставить текст на экране повернуться на 90 градусов и для этого переделывает системный драйвер... (Конечно, это шутка...)

То же самое относится, скажем, и к MIDAS. Если пользователь тут же начинает встраивать в него свои новые С или FORTRAN программы, то он либо действительно ограничен возможностями MIDAS, либо, с большей степенью вероятности, еще просто не понимает, КАК, используя 2–20 MIDAS-команд в командном файле, совершить то, что ему необходимо. Это понимание приходит только с опытом работы.

Однако вернемся к лекциям. Следующую лекцию я хотел бы посвятить рассмотрению двух вопросов:

1. Более детальное ознакомление с различными структурами данных, поддерживаемыми MIDAS.

**Замечание :** При оперировании изображениями, таблицами и так далее очень важно не только представлять, какое количество разнообразных операций MIDAS позволяет производить над этими структурами (это достаточно просто узнать, посмотрев на HELP по группам команд, имеющим общий квалификатор), но и знать, как обращаться к элементам этих структур.

2. Краткому обзору команд MIDAS по группам и назначениям.

## 1. Более подробно о работе с различными структурами данных

Я не раз видел, как, казалось бы, умудренный опытом пользователь впадает в прострацию, пытаясь изменить значение какого-нибудь единственного элемента спектра (одномерного изображения) или понять, как при выполнении командного файла вписать одно значение в таблицу. А ведь это делается действительно элементарно — достаточно просто знать, как обращаться к элементам различных структур.

### 1.1. Изображения. Элементы в изображениях

Поскольку чаще всего в MIDAS астрономы-наблюдатели имеют дело с изображениями (одномерными, двумерными или трехмерными), то я начну свой краткий обзор именно с них.

Файл изображения состоит из шапки, содержащей дескрипторы и тела изображения (массив данных размерности 1-3). Как и в случае FITS-файла, дескрипторы полностью описывают данные: тип, размерность и так далее (более подробно об этом смотрите ниже). Изображение состоит из элементов, или ячеек, или пикселей. Имеется очень большой набор команд, работающих с изображениями, и подавляющее большинство этих команд имеют квалификатор `.../IMAGE: READ/IMAGE, WRITE/IMAGE, COMPUTE/IMAGE, COPY/II ...`.

Кроме того, MIDAS предоставляет возможность обратиться не ко всему изображению в целом, а к некой прямоугольной его области (частный случай — один пиксел). Формат этого стандартного обращения можно записать в виде:

`frame[x1,y1:x2,y2]`, где

`x` — номер колонки,

`y` — номер строки,

`x1,y1` — координата левого нижнего угла выделяемой области,

`x2,y2` — координата правого верхнего угла выделяемой области.

В случае трехмерного массива вид обращения: `frame[x1,y1,z1:x2,y2,z2]`.

**ВНИМАНИЕ :** Помните о двух символах, обозначающих начало изображения "`<`" и конец изображения "`>`".

При таком стандартном обращении монитор, выполняя Вашу команду, должен из-

влечь эту область из полного изображения и подать на выполнение MIDAS-команде.

**Пример :**

COMPUTE/IMAGE qubodub = 0 — присвоение всем пикселям одномерного изображения значения 0.

COMPUTE/IMAGE qubodub[<:>] = 0 — то же самое.

COMPUTE/IMAGE qubodub[1000:>] = 0 — присвоение пикселям одномерного изображения значения 0. Номера пикселей: от 1000-го до последнего.

COMPUTE/IMAGE qubodub[1000,>:1000,>] = 0 — присвоение всей двумерной области (правый верхний угол) двумерного изображения значения 0.

PLOT/ROW Fabri[<,<,5:>,>,5] 100 — нарисовать 100-ую строку из 5-ой XY-плоскости трехмерного массива данных.

Полезно знать, что есть и короткая форма записи. К сожалению, она применима только для ряда частных случаев: строка в двумерном изображении или XY-плоскость трехмерного массива данных.

**Пример :**

PLOT/ROW Fabri@5 100 — нарисовать 100-ую строку из 5-ой XY-плоскости трехмерного массива данных.

Конечно, существуют различного рода исключения из этого правила. Это касается команд, у которых уже есть отдельный параметр для задания области работы (например, STAT/IMAGE). В этом случае все зависит от версии MIDAS, в которой Вы работаете.

**Замечание :** Например, у меня в версии 96NOV команда

COMP/IMA qq[1:50] = qq[51:100] не работает, а команда

COMP/IMA qq[<:50] = qq[51:100] работает, но в других версиях результат может быть иным.

Следующее, что необходимо знать каждому при работе с изображениями — это наличие двух систем координат: мировой и пиксельной — одновременно. Поскольку начало и шаг для элемента изображения (они описаны в дескрипторах изображения) могут быть любыми числами, то адресация к элементу изображения может производиться как по номеру пикселей (пиксельные координаты), так и по номерам абсолютным, вычисляемым с помощью дескрипторов START и STEP (мировые координаты). При этом задание пиксельных координат производится при помощи символа "@". Так [@30:@60] или [30:60] — это разные области одномерного изображения, хотя очень часто мировые и пиксельные координаты совпадают.

**ВНИМАНИЕ :** Это происходит, когда дескрипторы изображения START и STEP равны 1!

**Пример :**

Есть одномерный спектр **duboqub.bdf** (изображение), содержащий 1024 числа. Дескриптор START равен 3700.0 (конечно, ангстрем — спектр-то линейаризован, т.е. приведен к равномерному шагу!), а дескриптор STEP равен 3.0 (конечно, ангстрем). Тогда:

COMPUTE/IMAGE duboqub[3700:3709] = 0 — присвоение трем первым пикселям значения 0. Задание в мировых координатах.

COMPUTE/IMAGE duboqub[@1:@3] = 0 — то же самое. Задание в пиксельных координатах.

COMPUTE/IMAGE duboqub[<:@3] = 0 — то же самое.

**ВНИМАНИЕ** : Символы "<" и ">" можно использовать в обеих формах записи (начало/конец - оно и в Африке начало/конец).

**Пример :**

Есть двумерное изображение **duboqub.bdf** размером 512×512 чисел (пикселей). Дескриптор START равен 5.0,10.0 а дескриптор STEP равен 1.0,-1.0 — вот такой дурной случай. Тогда:

EXTRACT/IMAGE line = cube[<,@1:>,@1] — извлечение первой строки. Задание в пиксельных координатах.

EXTRACT/IMAGE line = cube[<,10:>,10] — извлечение первой строки. Задание в мировых координатах.

**Замечание** : Очевидно, что обе системы координат — линейные. Постепенно происходит переход к истинно мировой системе координат (так называемой WCS). На сегодняшний день в MIDAS уже есть команда — GET/CURSOR, работающая в этой системе координат. Это значит, что если Вы, например, возьмете любое изображение, полученное с оцифрованного Паломарского обзора, то эта команда будет выдавать астрометрические координаты.

**ВНИМАНИЕ** : Помните, что команда COMPUTE/IMAGE работает в мировых координатах. Если же Вам необходимо проделать арифметические операции с несколькими изображениями, забыв про мировые координаты, пользуйтесь командой COMPUTE/PIXEL.

**Замечание** : Команда COMPUTE/IMAGE также полезна, поскольку позволяет не только производить математические операции с изображениями, но и использовать ее как калькулятор (содержит очень большое число дополнительных функций).

## 1.2. Описатели (дескрипторы)

Не успели мы с Вами вспомнить о дескрипторах, как они — тут как тут. Напомню, что дескрипторы — это служебные переменные, описывающие информацию, заключенную в image-table-fit файлах MIDAS. Они содержатся в "шапках" этих структур и

полностью их описывают. Также они могут содержать некоторую служебную информацию: например, кем и когда получены эти данные, статистические характеристики, историю преобразований, которым они подверглись.

Из этого следует, что дескрипторы можно разделить на две группы — обязательные (есть всегда, ибо именно они описывают структуру изображений и таблиц) и остальные. Так, к стандартным дескрипторам для изображений (смотрите `HELP [Descr]`) относятся:

NAXIS	—	размерность
NPIX	—	число точек
START	—	начало на каждой из осей
STEP	—	шаг на каждой из осей
IDENT	—	имя
CUNIT	—	величина $[(NAXIS+1)*16]$
LHCUTS	—	min, max, vis_min, vis_max

В общем случае дескриптор — это вектор (набор) чисел или строк с максимальным количеством элементов 32767. Стандартные дескрипторы имеют длину имени, не превышающую 8 символов. Все остальные могут иметь длину имени до 72 символов. Дескрипторы бывают четырех типов: **integer(i)**, **real(r)**, **double(d)**, **character(c)**. Смешанные типы дескрипторов не поддерживаются. Общая форма обращения к дескриптору: **Имя/Тип/начало/сколько\_элементовN элемент1,...элементN**.

Пример :

`IDENT/C/1/60` — с 1-го по 60-ый элемент `character` дескриптора `IDENT`.

`STATIST/r/1/5 5.,6.,2.4,0.,1.2` — с 1-го по 5-ый элемент `real` дескриптора `STATIST`.

`STATIST/r/5/6 5.,6.,2.4,0.,1.2` — с 5-го по 11-ый элемент `real` дескриптора `STATIST`.

Для полного определения дескриптора требуется еще имя изображения или таблицы, в котором он находится.

Пример :

`WRITE/DES testima ident/c/1/60 "Mkn 996"` — записать в дескриптор `IDENT` изображения **testima.bdf** имя галактики.

`WRITE/DES testima statist/r/1/5 5.,6.,2.4,0.,1.2` — записать в `real` дескриптор `STATIST` изображения **testima.bdf** ряд чисел.

**ВНИМАНИЕ** : При чтении и записи дескрипторов для табличных и *fit*-файлов необходимо указывать соответствующие расширения `".tbl"` и `".fit"` в имени файла. Для изображений этого не требуется, "по умолчанию" берется изображение с расширением `".bdf"`.

**ВНИМАНИЕ** : Самый "заумный" тип дескриптора — это массив строк. Определить его можно так:

`WRITE/DESCR testima smore/c*5/1/10 abcde all` — определение дескриптора `smore` как массива из 10 строк по 5 символов в каждой и присвоение

значения "abcde" им всем.

Это аналог, например, "CHARACTER STORE(10)\*5" в FORTRAN или "char store[10][5]" в С. Очень удобен при создании собственного пакета.

Создается дескриптор командой WRITE/DES. Если он существует, то его можно дописать (скажем, состоял из 5-ти элементов, а увеличился до 10-ти).

Все команды работы с дескрипторами имеют квалификатор .../DESCR: READ/DESCR, SHOW/DESCR, WRITE/DESCR, DELETE/DESCR ....

При преобразовании MIDAS-файла данных (изображение, таблица) в FITS-формат значения дескрипторов заносятся в ASCII шапку FITS-файла.

### 1.3. Ключевые слова

Ключевые слова - это глобальные переменные в MIDAS. Часть из них зарезервирована для хранения системных параметров MIDAS. Остальные используются для временного хранения данных, результатов выполнения программ, передачи данных между программами.

Работа с ключевыми словами очень сильно похожа на работу с дескрипторами. В общем случае ключевое слово — это вектор (набор) чисел или строк с максимальным количеством элементов 32767. Ключевые слова могут иметь длину имени до 8 символов. Ключевые слова бывают четырех типов: **integer(i)**, **real(r)**, **double(d)**, **character(c)**. Смешанные типы ключевых слов не поддерживаются. Общая форма обращения к ключевому слову:

**Имя/Тип/начало/сколько\_элементовN элемент1,...элементN.**

Пример :

WRITE/KEY ident/c/1/60 "Mkn 996" — записать ключевое слово.

WRITE/KEY statist/r/1/5 5.,6.,2.4,0.,1.2

**Замечание :** "Заумный" тип ключевого слова — массив строк — также существует. Определить его можно так:

WRITE/KEY store/c\*5/1/10 abcde all — определение ключевого слова store как массива из 10 строк по 5 символов в каждой и присвоение значения "abcde" им всем.

Никогда за все годы моей работы с MIDAS я их не использовал. И Вам не советую. Но мало ли что...

В отличие от дескрипторов, ключевые слова не добавляются. Это значит, что если Вы уже определили размер ключевого слова, то для увеличения размера надо ключевое слово удалить и создать новое, с тем же именем, но большего размера.

Все команды работы с ключевыми словами имеют квалификатор .../KEYWORD: READ/KEY, SHOW/KEY, WRITE/KEY, DELETE/KEY, COMPUTE/KEY, COPY/KEY ....

Существуют стандартные ключевые слова, которые создаются, когда MIDAS стартует. Их полезно знать. Для ознакомления с ними смотрите команду HELP [Key].

**Замечание :** Очень полезна команда COMPUTE/KEY, которая позволяет производить математические операции с ключевыми словами, а также содержит достаточно большое число дополнительных функций.

## 1.4. Элементы в таблице

Наиболее часто таблицы используются при анализе полученных материалов, поскольку позволяют хранить неоднородные данные (каждая колонка может содержать свой собственный формат).

Файл таблицы состоит из шапки, содержащей дескрипторы и тела таблицы (если считать, что обыкновенная таблица — это двумерный массив данных, то MIDAS позволяет работать с трехмерными таблицами, и в этом случае каждая ячейка таблицы является, в свою очередь, одномерным массивом данных). Имеется очень большой набор команд, работающих с таблицами, и подавляющее большинство этих команд имеют квалификатор `.../TABLE: READ/TAB`, `SHOW/TAB`, `WRITE/TAB`, `DELETE/TAB`, `COMPUTE/TAB`, `COPY/TAB` .... Таблица состоит из элементов или ячеек. Доступ к отдельному элементу осуществляется по имени таблицы, имени (номеру) колонки и номеру строки, в которых этот элемент находится. Адресация к колонке производится по имени (`:COLUMN_NAME`) или номеру (`#COLUMN_NUMBER`). Адресация к строке производится или по номеру (`@ROW_NUMBER`) или с использованием референс-колонки, определяемой командой `SET/REFCOLUMN`.

**Пример :**

`READ/TABLE mytab #3 @10..20` — прочитать содержимое с 10 по 20 элемент колонки номер 3 таблицы `mytab.tbl`.

`READ/TABLE mytab :MAGNITUDE @10..20,100..200` — прочитать содержимое с 10 по 20 элемент, а также с 100 по 200 элемент колонки с именем `MAGNITUDE` таблицы `mytab.tbl`. // `WRITE/TABLE mytab :MAGNITUDE @10 13.5` — записать значение.

**Замечание :** *Внимательно изучите команду `COMPUTE/TABLE`. Это одна из основных команд для работы с таким типом данных. Содержит очень большое число дополнительных функций, также позволяющих преобразовывать колонки одного типа в другой.*

## 1.5. Работа с каталогами

Каталоги используются обычно для работы с группами файлов одинаковой структуры: изображениями или таблицами и т. д. Для работы с различными группами используются различные квалификаторы:

`.../ICAT` — для работы с изображениями;

`.../TCAT` — для работы с таблицами;

`.../FCAT` — для работы с Fit-файлами;

`.../ACAT` — для работы с ASCII-файлами.

Для операций с каталогами существует достаточно ограниченное число команд: `CREATE/xCAT`, `SET/xCAT`, `CLEAR/xCAT`, `READ/xCAT`, `ADD/xCAT`, `SUBTRACT/xCAT`.

Однако необходимо помнить о команде `EXECUTE/CATALOG`, которая и позволяет, собственно, после создания каталога выполнить определенную операцию для всех файлов из каталога (пример работы с этой командой смотрите в Лекции 2).

## 1.6. Еще быстрее...

При использовании прямого обращения к элементам изображений, таблиц и т.д. в командных файлах помните, что при выполнении команд типа `READ/...` `WRITE/...` монитор вызывает командный файл, который, в свою очередь, вызывает исполняемую программу. Это замедляет работу, особенно если Вы исполняете подобные команды часто. Для ускорения работы используйте нижеприведенный формат обращения к данным структурам. При таком обращении команда будет выполняться самим MIDAS-монитором:

**frame,descr = value** — запись ОДНОГО значения в дескриптор изображения или таблицы;

**frame[x,y,z = value]** — запись значения ОДНОГО пикселя в изображение;

**table,column,row = value** — запись ОДНОГО табличного элемента.

Пример :  
`WRITE/DES testima statist/r/6/1 5.` или:  
`testima,statist(6) = 5.`  
`WRITE/IMAGE testima [@1,@1:@1,@1] 1.0` или:  
`testima[@1,@1] = 1.0`  
`WRITE/TABLE mytable :RADVEL @20 13.5` или:  
`mytab,:RADVEL,20 = 13.5`

**ВНИМАНИЕ** : Пробелы при написании этих команд обязательны!

## 2. Обзор MIDAS-команд по группам и назначениям

Весь огромный набор команд MIDAS можно примерно разделить на следующие группы (или категории):

- Primitive commands — команды нижнего уровня. Они в свою очередь разделяются на команды ядра (команды, исполняющиеся самим MIDAS-монитором). Например, `SET/MIDAS`, `DELETE/IMAGE` и базовые команды, исполняющиеся из командных файлов (например, `PLOT/TABLE`).
- Application commands — команды всех дополнительных пакетов. В свою очередь эти пакеты разделяются на три категории:
  - `Applic` — команды из дополнительных пакетов, расширяющие стандартный набор команд (например, `COMPUTE/FIT`);
  - `Stdred` — команды из пакетов, созданных для проведения первичной редукции данных, полученных на различных приборах (например, `EXTRACT/ECHELLE`);
  - `Contrib` — команды из пакетов, созданных для проведения дальнейшего анализа данных (например, `SEARCH/INVENTORY`).



- Procedure control commands — командный язык монитора, используемый при написании командных файлов (например, DEFINE/PARAMETER).

**Замечание** : Знать (хотя бы приблизительно), какие команды к какой категории относятся, чрезвычайно полезно, особенно при написании командных файлов — это может ускорить время выполнения Вашей программы в несколько раз. А найти эту информацию можно в тех трех томах MIDAS документации, о которых я уже упоминал ранее.

Теперь я попытаюсь объединить Primitive commands в группы по назначению. При этом, я совсем не буду упоминать группы команд, необходимых для ввода-вывода и работы с image и графическими дисплеями, поскольку я рассматривал их в предыдущих лекциях.

Итак, группы по назначению:

**Работа с координатами изображения** — набор программ для считывания координат изображения или определения центра:

```
CENTER/GAUSS, CENTER/MOMENT
GET/CURSOR,   GET/GCURSOR
```

**Координатные преобразования изображений** — различные программы преобразований: вращения, извлечение части и помещение новой части в изображение, переход к новому шагу и т.д.

```
ALIGN/IMAGE, EXTRACT/IMAGE, FLIP/IMAGE, TRANSPOSE/IMAGE
GROW/IMAGE,  INSERT/IMAGE,
REBIN/II (liner, polinom, exp, ..., user define) - только 1D изображения
REBIN/LINEAR (start, step)                       - и 2D тоже
REBIN/ROTATE (CENTER/..., ALIGN/IMA)
REBIN/SPLINE,
REBIN/IT (liner, polinom, exp, ..., user define) - только 1D изображения
ROTATE/CLOCK, ROTATE/COUNTER_CLOCK, ROTATE/1DIM
XCORR/IMAGE
```

**Арифметика** — программы выполнения арифметических операций над изображениями. Сюда я также отношу и усреднение нескольких изображений в одно.

```
AVERAGE/AVERA all pixels in the subframe
AVER/COLUMN, AVER/ROW
AVERAGE/IMAGES (aver,min,max,med)
AVER/KAPPA, AVER/WEIGHT (COMP/WEIGHT)
COMP/ROW, COMP/COLUMN
COMP/IMAGE, COMP/PIXEL
Функции: SQRT(a), EXP(a), EXP10(a), LN(a), LOG10(a), SIN(a) ASIN(a), COS(a),
ACOS(a), TAN(a), ATAN(a), INT(a), ABS(a), ATAN2(a1,a2), MAX(a1,a2),
MIN(a1,a2) и MOD(a1,a2)
```

## **ВНИМАНИЕ** : *Помните про отличие COMP/IMAGE от COMP/PIXEL!*

**Фильтрация** — программы проведения фильтрации различного рода по всему изображению или по его части.

```
CONVOLVE/IMA, DECONVOLVE/IMA (with psf)
CREATE/FILTER
FFT/IMAGE, FFT/INVERSE, FFT/POWER, FFT/FREQ
FILTER/COSMIC, FILTER/DIGITAL, FILTER/GAUSS,
FILTER/MAX, FILTER/MIN, FILTER/MEDIAN,
FILTER/SMOOTH, FILTER/ADAPTIV
```

**Image creation and extraction** — программы для создания новых изображений различными способами, в том числе и путем извлечения различных частей из уже существующих изображений.

```
CREATE/IMAGE, CREATE/RANDOM (gauss, exp, pol, poisson ...)
EXTRACT/IMA, EXTRA/CURSOR, EXTRACT/SLIT
EXTRACT/STRACE, EXTRACT/RTRACE, EXTRACT/LINE (диагональ),
EXTRACT/TRACE, EXTRACT/REFERENCE_IMAGE, EXTRACT/ROTATED_IMAGE
```

**Преобразование пиксельных значений** — программы модификации пиксельных значений. Модификация производится либо в областях, отмеченных курсором, либо на основе выбранных критериев.

```
FIT/FLAT_SKY, ITF/IMAGE,
MODIFY/CURSOR, MODIFY/GCURSOR
MODIFY/PIXEL, MODIFY/AREA
REPLACE/IMAGE
REPLACE/POLYGON - работает только для полигонов.
```

**ВНИМАНИЕ** : *Хотелось бы специально выделить программу REPLACE/IMAGE. Разберитесь с ее использованием: это должно облегчить Вашу жизнь при проведении обработки или анализе. Данная программа предназначена для создания изображений-масок. Имеет тысячу других приложений.*

<pre>Пример : REPL/IMA input out &lt;,100=0. REPL/IMA input out &lt;,100=ref REPL/IMA input out &lt;,100=ref+2.5,d</pre>
--

**Numerical values of image pixels** — программы для получения различных характеристик анализируемых изображений.

```
FIND/MINMAX, INTEGR/APERTURE, INTEGR/LINE
MAGNITUDE/CIRCLE, MAGNITUDE/RECTANGLE
STATIST/IMAGE
```

**Fitting** — программы приближения имеющих распределений различными функциями.

```
COMPUTE/FIT, COMPUTE/FUNCTION,
EDIT/FIT, FIT/IMAGE, FIT/TABLE, READ/FIT, SET/FIT
SHOW/FIT, SELECT/FIT
REGRESSION/POLYNOMIAL, SAVE/REGRESSION
```

**Работа с таблицами** Я хотел бы еще более подробно остановиться на работе с таблицами. С моей точки зрения, для эффективной работы с ними Вам необходимо знать следующие вещи:

1. Каждая таблица содержит две дополнительных колонки — первая — это просто номер строки (называется **:SEQUENCE** или **:SEQ**), а вторая называется **SELECT** или **SEL** и равна или 0 или 1 для каждой строки, которая удовлетворяет/не удовлетворяет набору условий соответственно. Обратите внимание, что хоть формально это и колонка, но пишется без ":". Знание этого момента, а также команд **SELECT/TAB** и **COMPUTE/TAB** позволяет творить с таблицами чудеса.

**ВНИМАНИЕ** : Учтите, что команда **COMPUTE/TABLE** в конце удаляет все выделения (практически работает как команда **SELECT/TAB Table All**).

Пример :

**SELECT/TAB CFA :MAG.lt.15.and.:VEL.gt.1000** — выделяем в таблице, содержащей каталог галактик, все объекты ярче  $15^m$  со скоростями больше 1000 км/сек.

**COMP/TAB CFA :Z = min(:MAG,:R) + (1-SEL)\*:VEL** — создаем новую колонку, в которую записываем для каждой строки минимальное значение из колонок **:MAG** и **:R** и добавляем к нему значение скорости для выделенных предыдущей командой строк (в записанном мной виде выделенные значения будут умножены на 1).

**COMP/TAB CFA :SURF\_BRIGHT = 2.5 \* log10(10\*\*((0.4\*:MAG)/(:D\*\*2)))** — считаем для всех галактик среднюю поверхностную яркость.

**COMP/TAB mytab :name = concat(:star,tochar(:SEQ))** — делаем колонку: имя звезды+номер строки таблицы, в которой она есть.

2. Если в поле таблицы нет значения, это называется NULL-величина. Обозначается как "\*". На эту же величину будут заменяться все результаты некорректных операций (LOG от -1, например).

3. Есть набор команд для копирования колонок из таблицы в таблицу, а также преобразования структуры "таблица" в структуру "изображение" и наоборот: **COPY/KT**, **COPY/TK**, **COPY/TT**, **COPY/TI**, **COPY/IT**...

**ВНИМАНИЕ** : Разберитесь с командой SET/REFCOLUMN в этом месте. Она позволяет то, чего не позволяет делать просто знание о колонке :SEQ.

4. Таблицы в MIDAS можно интерактивно редактировать. Редактор таблиц вызывается командой EDIT/TAB. В редакторе надо знать волшебные последовательности:

<Ctrl/Z> — выход в командный режим

<Ctrl/Z>q<Enter> — выход без записи

<Ctrl/Z>e<Enter> — выход с записью

<Ctrl/Z>help<F1><F1> — выдача списка функций

После выхода в командный режим можно пользоваться набором встроенных функций. Вот их список:

```

keypad  line    screen  word    _last_column  change_format  advance
bottom_of_table  backup  top_of_table  create_column  delete_column  page
command section  right_page  left_page    _delete_word   _undel_word
_ delete_character  _undel_character  _next_line    row    gold
next    find    _delete_line  _undel_line    _arrow  _bad_key    tabular
help    show    sort    status  Function

```

5. И еще набор команд, существующих для таблиц:

```

COMPUTE/HISTOGRAM, COMPUTE/REGRESSION
REGRESSION/LINEAR, REGRESSION/POLY,
SAVE/REGRESSION, COMPUTE/TAB, CONVERT/TAB,
INTERPOLATE/TT, INTERPOLATE/TI, INTERPOLATE/IT,
REBIN/TT, REBIN/TI, REBIN/IT          (1D)
MERGE/TABLE, JOIN/TABLE, NAME/COLUMN,
SELECT/TAB, SORT/TABLE, STATIST/TABLE

```

## 2.1. Не забывайте делать HELP/APPL

Помните про то, что всегда существует некий дополнительный набор командных файлов (так называемые applications), которые написаны другими пользователями и постоянно пополняются. Эти командные файлы иногда бывают очень полезны и вызываются как @а Имя\_программы. Их список можно получить, используя HELP/APPL, а полный набор параметров для каждой, — используя HELP/APPL Имя\_программы. Ниже я привожу список этих программ, существующий в версии 98NOV с кратким описанием того, что данная команда выполняет.

**asload** — назначение дисплейного окна и визуализация в нем изображения;

**areatable** — построение таблицы, содержащей определение областей заданного размера, равномерно покрывающих заданное изображение;

- autocuts** — определение "хороших" уровней для визуализации изображения;
- buildcube** — построение куба данных из заданного набора двумерных изображений;
- catalsync** — проверка набора дескрипторов всех изображений/таблиц данного каталога этих файлов;
- checker** — создание изображения хитрого содержания;
- checkname** — проверка имени MIDAS-файла на наличие некорректных символов;
- convolfft** — свертка или корреляция изображения с изображением с использованием быстрого преобразования Фурье;
- copyfile** — копирование ASCII файлов по некоторым правилам;
- diffdsc** — сравнение заголовков двух изображений;
- diffima** — сравнение пикселей в двух изображениях;
- dscedit** — редактирование дескриптора;
- edges** — нахождение краев с использованием фильтров Sobel или Prewitt;
- expand** — увеличение размера изображения;
- fastcomp** — быстрое вычисление функций  $\ln$ ,  $\log_{10}$  и  $\sqrt{\phantom{x}}$  от изображения;
- func2d** — создание изображений, соответствующих любой двумерной функции;
- fftflt** — фильтрация изображений в Фурье плоскости;
- gammacorr** — создание "хитрой" ГТТ-таблицы;
- getcoord** — пересчет пиксельных координат в мировые (и наоборот) для заданного изображения;
- histogram** — построение изображения по гистограмме от изображения;
- matconst** — определение часто используемых математических констант;
- normalize** — нормализация входного изображения (вписывание значений в диапазон [0.,1.]);
- outima** — преобразование MIDAS-изображений в другие форматы (требует настройки при инсталляции MIDAS);
- packdsc** — убирание "пустого" места из заголовка изображения;
- perspec** — построение и визуализация 3-х мерной перспективы от куба данных;
- polyrep** — отметить полигон на визуализованном изображении и заменить его на константу;

- plottab** — нарисовать содержание LUT или ITT-таблиц;
- replace** — расширенный вариант команды `REPLACE/IMAGE`;
- resetback** — восстановить соединение с фоновым MIDAS;
- sancheck** — проверить изображение на наличие NaN или Inf (медленная процедура);
- scale** — промасштабировать изображение в данной области величин;
- scaler** — промасштабировать изображение по алгоритму команды `LOAD/IMAGE`, но сделать это на диске.
- scanima** — сделать из 2D-изображения 1D-изображение по определенным законам (иногда очень полезна);
- sharpen** — выделение краев изображения с использованием специального маскирования;
- showfonts** — показать фонты, которые могут быть использованы в MIDAS;
- showmidas** — показать текущие MIDAS-установки;
- slicube** — извлечь 2D плоскости из 3D куба данных с определенным шагом;
- swap4** — переставить 4 квадранта изображения по диагонали;
- vertlut** — нарисовать LUT-таблицу вертикально и надписать уровни;
- zperspec** — построение и визуализация перспективной проекции двумерного изображения.

### 3. Работа с контекстами

Последнее, что хотелось бы описать в этом обзоре различных групп команд — работа с контекстами. MIDAS-контекст — это пакет программ, внешний по отношению к монитору. Контекст можно "загрузить" — выполнить команду `SET/CONTEXT имя_контекста`, после которой все команды этого пакета становятся доступными для исполнения (также становится доступным `HELP` на эти команды) и "выгрузить" — исполнить команду `CLEAR/CONTEXT имя_контекста`.

Насколько я понимаю, было две основных идеи создания контекстов:

1. Контекст — это набор программ, которые создаются любым человеком, а после этого "встраиваются" им в MIDAS.
2. Пользуясь контекстами, производится первичная редукция различного рода данных, в результате которой эти данные сводятся к основным структурам MIDAS. После чего, опять-таки пользуясь различными контекстами, можно производить анализ полученных и обработанных данных.

Ниже я очень коротко охарактеризую пакеты, доступные в настоящее время в MIDAS:

**Applications** — дополнительные пакеты, расширяющие стандартный набор команд;

**Plot** — программы, работающие с графикой, которые не относятся к базовым программам: (1) Апертурная интеграция; (2) Вычисление звездной величины; (3) Вычисление размера апертуры в координатах экрана; (4) Интерактивная идентификация линий.

**Statist** — статистические тесты таблиц: сравнение эмпирического распределения с теоретическим распределением, сравнение независимых выборок данных, измерение корреляции выборок и т.д.;

**Table** — Различные дополнительные утилиты для работы с таблицами, в частности, разбиение данных в столбцах на интервалы, усреднение колонок;

**FIT** — пакет для приближения распределений с использованием набора как стандартных функций, так и функций, определяемых пользователем. Приближаются как табличные данные, так и изображения, как одномерные, так и двумерные распределения.

**Stdred** — пакеты, созданные для проведения первичной редукции данных, полученных на различных приборах;

**CCDRED** — пакет для проведения первичной редукции CCD-изображений. При полной настройке работает в автоматическом режиме.

**CCDTEST** — предназначается для проведения регулярного тестирования CCD-приемников. Содержит следующие тесты:

- (1) тест BIAS-изображений;
- (2) вычисление горизонтальной и вертикальной эффективностей переноса;
- (3) тест DARK-изображений;
- (4) проверка эффективности затвора;
- (5) тест линейности приемников.

**DO** — организация данных. В частности, используется при организации базовых таблиц при проведении первичной CCD-редукции пакетом CCDRED;

**ESHELLE** — обработка эшелле-спектров. Пакет содержит ~40 команд.

**ESHELLEC** — старый вариант обработки эшелле-спектров. Пакет содержит команды обработки эшелле-спектров, полученных с двумерным счетчиком фотонов в качестве приемника. Программы были разработаны и встроены в MIDAS с участием автора этих лекций (Kniazev, Shergin & Lipovetsky, 1992).

**IRAC2** — обработка данных IRAC2;

**IRSPEC** — пакет для обработки спектров, получаемых со спектрографом IRSPEC (IR спектрограф);

**LONG** — стандартная обработка спектров, полученных с модой "длинная щель". Пакет содержит ~20 команд. Шаги: исправление двумерного спектра за геометрические искажения, калибровка длин волн и исправление за дисторсию вдоль щели для изучения особенностей в спектрах протяженных объектов, исправление за кривую спектральной чувствительности и т.д.;

**MOS** — обработка данных мульти-объектного спектрографа. Пакет содержит ~15 команд для калибровки, построения дисперсионных кривых, линеаризации данных и т.д.;

**OPTOPUS** — обработка данных, полученных на фиберном спектрографе;

**PISCO** — полная обработка поляриметрических данных, полученных с поляриметром PISCO;

**QC** — пакет контроля качества изображений в течение наблюдений: анализируется качество по времени изображений BIAS, DARK и научных;

**SPEC** — набор программ, необходимых для работы с одномерными спектрами. Эти команды обычно используется в других спектроскопических пакетах, например, LONG, ECHELLE. Пакет содержит ~10 команд для построения дисперсионной кривой, исправления за кривую спектральной чувствительности, построения кривой спектральной чувствительности, коррекции за ripple-эффект и т.д.;

**Contrib** — пакеты, созданные для проведения дальнейшего анализа данных;

**ASTROMETRY** — пакет для проведения астрометрии. MIDAS версия пакета состоит из трех основных шагов: (1) чтение измерений положений стандартных звезд и вычисление коэффициентов преобразования; (2) редактирование таблицы стандартных звезд с целью удаления плохих измерений; (3) пересчет измеренных координат с использованием коэффициентов преобразования.

**CLOUD** — моделирование особенностей межзвездного поглощения. Модель представляется в виде полиномиального континуума, на который можно наложить эмиссионные и/или абсорбционные линии;

**DAORHOT** — получение точной фотометрии и астрометрических позиций звездных объектов. Пакет содержит следующие шаги:

- (1) нахождение всех звездных объектов на изображении;
- (2) апертурная фотометрия для всех выделенных объектов;
- (3) построение точечной функции рассеяния для изображения, с использованием одной или нескольких звезд;
- (4) вычисление точных позиции и величин всех найденных звезд.

**GEOTEST** — программы для создания тестовых изображений разных видов;

**HST** — пакет для преобразования данных, полученных на космическом телескопе Хаббла во внутренний формат MIDAS.

**IMRES** — программы восстановления изображений (алгоритмы, применяемые для восстановления изображений в StScI);

**INVENTORY** — выделение объектов разного типа на изображениях и классификация этих объектов. Функционально пакет состоит из трех шагов: поиск всех объектов на изображении, анализ всех объектов на изображении и классификация всех объектов.

**IUE** — работа с данными, полученными со спутником IUE.



**LYMAN** — пакет для проведения многокомпонентного разложения межзвездных абсорбционных линий, который может использоваться для анализа спектров высокого разрешения.

**MVA** — статистический анализ многомерных выборок. Содержит:

- (1) Метод главных компонент;
- (2) Определение набора неперекрывающихся кластеров;
- (3) Иерархическую кластеризацию;
- (4) Линейный дискриминантный анализ;
- (5) Множественный дискриминантный анализ; (6) Дискриминантный анализ по K-ближайшим соседям.

**PEPSYS** — планирование фотометрических наблюдений и полная обработка фотометрических наблюдений (не только CCD-фотометрия, но также наблюдения и обработка электрофотометрических наблюдений). Учитывается информация о телескопе и инструменте, полученные величины преобразуются в стандартную систему (если это возможно), оцениваются ошибки получаемых величин;

**ROMAFOT** — получение звездной фотометрии для полей с большой плотностью объектов (например, фотометрия звезд шаровых скоплений). Незвездные объекты автоматически удаляются;

**SURFPHOT** — набор программ, полезных при проведении поверхностной фотометрии. Содержит: (1) программы проведения изофот и фона; (2) поиск позиционного угла и наклона галактики; (3) анализ азимутальных профилей; (4) коррекция изображений, искаженных дисторсией; (5) коррекция плохих пикселей; (6) нормализация изображений; (7) восстановление изображений;

**TSA** — пакет для анализа временных рядов. Содержит 15 команд;

**WAVELET** — wavelet преобразование. Данное преобразование извлекает из изображений информацию об их пространственных и частотных свойствах. Очень полезно для описаний иерархических структур. В пакет входят определенные алгоритмы (pyramidal algorithms, Mallat's algorithm with bi-orthogonal filters, algorithms using the FFT, ...), программы визуализации и две программы работы с изображениями (фильтрация и сравнение).

**SAO** — пакеты, созданные как в Специальной Астрофизической Обсерватории, так и в других обсерваториях, но встроенные в MIDAS в CAO РАН. Не являются стандартными в MIDAS, но доступны в CAO РАН.

**AIP** — пакет проведения адаптивной фильтрации и поверхностной фотометрии изображений. Содержит также топологическую работу с масками. Программы созданы G.Richter в Астрофизическом Институте Потсдама (Германия). Встроены в MIDAS автором (Князев 1997). Доступны только в CAO РАН или по согласованию с автором программ (gmrichter@aip.de).

**SCANNER** — обработка данных, полученных с одномерным счетчиком фотонов (Князев 1994; Князев 1997);

**NICE** — наблюдения с CCD (Князев, Шергин 1995). Позволяет проводить сбор **различных** типов данных (спектральных в модах “длинная щель” и “эшелле” и фотометрических) на **различных** телескопах (БТА и Цейс-1000) и параллельную сетевую буферизацию данных. Пакет построен таким образом, что позволяет:

1. Легко менять моды наблюдений (спектральные или фотометрические) в течение наблюдательной ночи ;
2. Быть настроенным на новые типы CCD;
3. Встраивать новые системы сбора;
4. Поддерживать FLEX-формат имен выходных данных (Кононов 1995), зависящий от систем сбора;
5. Производить сетевую буферизацию получаемых данных.

**ОВРР** — пакет для обработки низкодисперсионных спектров, полученных с CCD в качестве приемника. Разработан автором (Kniazev et al. 1996; Kniazev et al. 1997; Князев 1997). Содержит: программу цифрового фильтра подавления низких частот (Shergin et al. 1996), пакет поиска и выделения объектов по прямым снимкам, программы привязки спектрального изображения к прямому, программы извлечения одномерных спектров из двумерных спектральных снимков, полученных с объективной призмой, собственный пакет астрометрии и собственный пакет апертурной фотометрии.

**EXAS** — система редукции и анализа рентгеновских данных, полученных со спутником ROSAT.

И это конец четвертой сказки о MIDAS.

## Список литературы

- Князев А.Ю.: Система управления и сбора данных спектрофотометра в Несмит-1 БТА. IV. Редукция сканнерных данных в MIDAS. — *Отчет САО*, 1994, **223**, с.1–28.
- Князев А.Ю., Шергин В.С.: CCD наблюдения в среде MIDAS. Контекст NICE. Версия 1.0. — *Отчет САО*, 1995, **239**, с.1–19.
- Князев А.Ю., Кандидатская диссертация, 1997, с.1–201  
([http://precise.sao.ru/Laboratory/Dis\\_akn/index.html](http://precise.sao.ru/Laboratory/Dis_akn/index.html))
- Кононов В.К.: Унифицированный формат представления наблюдательных данных (FLEX-стандарт). — *Препринт САО*, 1995, № 111Т, с.3–21.
- Kniazev A.Y., Shergin V.S., Lipovetsky V.A. (1992) MIDAS on small computers: reduction of low order echelle spectra. 4-th ESO/ST-ECF Data Analysis Workshop: 169–176
- Kniazev A.Yu., Salzer J., Lipovetsky V.A., Boroson T., Moody J., Thuan T., Izotov Yu.I., Herrero, J., & Frattare, L.: KISS: A New Digital Survey for Emission-Line Objects. — *in Proceeding of 179 symposium of IAU*, 1996, p.500-503.
- Kniazev A.Yu., Salzer J., Lipovetsky V.A., Boroson T., Moody J., Thuan T., Izotov Yu.I., Herrero, J., & Frattare, L.: KISS: A New Digital Survey for Emission-Line Objects. — *Newsletter Working Group on Wide field imaging, IAU Comission 9*, 1997, **9**, p.5–10.
- Shergin V.S., Kniazev A.Yu., & Lipovetsky V.A.: New family of non-linear filters for background subtraction of wide-field surveys. — *Astronomische Nachrichten*, 1996, **2**, с.95-100.

MIDAS Users Guide, 1995a, **Volume A**  
MIDAS Users Guide, 1995b, **Volume B**