

Основы макроязыка EASI (PCI Geomatica) Часть 2.

Анализ данных

[Обсудить в форуме](#) Комментариев — 0

Эта страница опубликована в основном списке статей сайта по адресу <http://gis-lab.info/qa/easi2.html>

Вторая часть серии статей о языке EASI, основные команды языка для анализа данных.

Содержание

- [1 Арифметические операции](#)
- [2 Логические операции](#)
- [3 Специальные команды для выполнения математических операций](#)
 - [3.1 ARI – арифметические вычисления](#)
 - [3.2 SCALE – применение линейных и нелинейных функций к значениям раstra](#)
 - [3.3 CLR – Очистка канала изображения](#)
- [4 Операции над растровыми масками](#)
 - [4.1 BLO – Логические операции над растровыми масками](#)
 - [4.2 MAP – Копирование растровой маски в канал изображения](#)
 - [4.3 THR – создание растровой маски по интервалу пороговых значений](#)
 - [4.4 RANDBIT – простая команда для создания случайной выборки пикселей](#)
- [5 Команда MODEL и язык моделей](#)
- [6 Фильтры](#)
 - [6.1 FAV – Вычисление среднего в плавающем окне](#)
 - [6.2 FME – Вычисление медианы в плавающем окне](#)
 - [6.3 FED – Фильтр для обнаружения границ \(edge detection\)](#)
 - [6.4 FMO – Вычисление моды](#)
- [7 Гистограммы значений](#)
 - [7.1 HIS – Графический вывод гистограммы значений по каналу](#)
 - [7.2 HISDUMP – Вывод гистограммы значений в виде таблицы](#)
- [8 Работа с векторными данными](#)
 - [8.1 VECPRO – Изменение проекции векторных сегментов](#)
 - [8.2 GRDPOL – Растеризация полигональных векторных данных](#)
 - [8.3 KRIGING – Интерполяция точечных векторных данных](#)
 - [8.4 RTV – Векторизация растровых данных](#)
- [9 Краткий перечень специальных команд анализа изображений](#)
- [10 Ссылки по теме](#)

Арифметические операции

Язык EASI отличается чрезвычайной простотой выполнения арифметических и логических операций над растровыми данными. Для выполнения расчетов с использованием нескольких растров, необходимо, что бы изображения были собраны в один файл PCIDSK. Для проведения расчетов дополнительные каналы данных из других файлов можно временно привязать к рабочему файлу с помощью команды PCIADD2 с опцией exist (см. [предыдущую статью](#)).

Перед началом выполнения операций необходимо задать рабочий файл с помощью параметра file:

file=" (имя файла)

При создании арифметических выражений ссылка на каналы данных обозначается «%»: %1 – первый канал.

Ссылка на растровые маски (сегменты) выглядит так: %%2 – сегмент номер 2.

Примеры записи простых выражений:

- $\%2 = \%1 + 10$ – к значениям пикселей канала 1 прибавить константу (10) и записать в канал 2.
- $\%1 = 0$ – обнулить значения канала 1.
- $\%7 = (\%1 - \%2) / (\%1 + \%2)$ – Вычисление NDVI и запись его в 7 канал.
- $\%16 = 100 * ((\%3 - \%2) / (\%3 + \%2) + 1)$ – Вычисление NDVI и масштабирование результата из диапазона -1 – 1 в диапазон 0 – 200.

Возможна запись результата в исходный канал:

- $\%10 = \%10 + 5$ – увеличиваем все значения пикселей на константу.

Существует ряд дополнительных команд, которые можно использовать в арифметических выражениях:

- min, max – минимальное и максимальное значение пикселя по заданным каналам,
- int, abs – округление до целого и модуль,
- sin, cos и т.д. – тригонометрические функции,
- ln, exp, log10, exp10 – логарифмические функции.

Пример использования функции:

- $\%10 = \max(\%1, \%2, \%3, \%4, \%5, \%6, \%7, \%8, \%9)$ – вычисление максимального значения из 9 каналов.

В случае, когда результат нужен только на время вычисления, можно использовать переменные. Переменные декларируются перед началом операции:

```
local int i
i = max(%1,%2,%3)
```

Если однотипную операцию надо провести над множеством каналов, можно использовать циклы:

```
local int i
for i = 10 to 15
  %{i} = %{i-9} * 100
endfor
```

Каналы 10-15 получат значения каналов 1-6, умноженных на 100.

Впрочем, использование циклов не ускоряет расчеты, поэтому для наглядности обычно удобнее использовать простые команды.

Для обозначения пространственных координат пикселя существует набор встроенных переменных, которые также бывает полезно использовать в расчетах:

- @x, @y – положение пикселя по отношению к верхнему левому углу, в пикселях
- @geox, @geoy – метрические координаты пикселя в проекции растра, и другие

Дополнительно об использовании встроенных переменных смотрите в справочной системе EASI.

Логические операции

Описание простого условия в EASI выглядит так:

- if (%1>100) %2=3 – если значения канала 1 больше 100, канал 2 принимает значение 3.

Условие с записью результата в растровую маску (сегмент):

```
if (%1>100) %%2=1
```

Помните, что пиксели растровой маски принимают только значения 0 и 1.

Выполнение действия только в пределах растровой маски:

```
if (%%2=1) %2=%1+10
```

Двойное условие:

```
if (%1>100) and (%3=1) %2=3
if (%1>100) or (%3=1) %2=3
```

Сложное условие:

```
if (%1>100) or (%2<=1) then
%3 = 5
%4 = 20
else
%3 = 0
%4 = 10
endif
```

Специальные команды для выполнения математических операций

ARI – арифметические вычисления

Использование:

- file=" (имя файла)
- oper="add/sub/mul/div/and/or (тип операции)
- cnst= (значение константы – если есть)
- mask= (маска для проведения операции)
- dbic= (список входных каналов)
- dboc= (канал для записи результата)
- auto= (автоматическое масштабирование результатов)
- rval= (значения минимального и максимального порогов для входных и выходных значений при масштабировании: rval=IN_min,IN_max,OUT_min,OUT_max)
- zerodiv= (значение, которое будет выдаваться при получении не действительного результата – например, при делении на ноль)

```
r ari
```

Использование команды ARI в цикле:

```
local int i

for i = 1 to 5
cnst=90
mask=
oper="add
dboc=i

r ari

endfor
```

К значениям каналов 1-5 будет прибавлена константа 90.

SCALE – применение линейных и нелинейных функций к значениям растра

Пример простого использования – изменение диапазона значений 1-100 на 1-255

- `file=` (имя входного файла)
- `filo=` (имя выходного файла)
- `dbic=` (список входных каналов)
- `dboc=` (список выходных каналов)
- `inrange=1,100`
- `outrange=1,255`

`r scale`

Дополнительный параметр `sfunct` позволяет задать тип функции для масштабирования - подробнее смотрите в справочной системе EASI.

CLR – Очистка канала изображения

- `file=` (имя файла)
- `dboc=` (список каналов)
- `valu=` (значение, которое будет записано в канал)
- `dbow=` (опционально – возможность задать прямоугольную область для очистки)

`r clr`

Операции над растровыми масками

BLO – Логические операции над растровыми масками

- `file=` (имя файла)
- `logfunc="and/or/xor/sub/not` (тип операции)
- `dbsn="` (имя выходной маски – обязательный параметр)
- `dbib=` (входные маски a,b)
- `dbob=` (выходная маска. Если пустой – будет создана новая)

`r blo`

Типы операций для масок a и b

- `and` — a и b;
- `or<tt>` — a или b;
- `<tt>xor` – что-то одно, если обе маски нулевые или =1, то выходное значение =0;
- `sub` — a и не b;
- `not` – «негатив» маски (в `dbib` должно быть только одно значение).

MAP – Копирование растровой маски в канал изображения

- `file=` (имя файла)
- `dbib=` (маска)
- `dboc=` (каналы изображения, которые будут изменены)
- `valu=` (значение, которое записывается в канал изображения под маской)

`r map`

THR – создание растровой маски по интервалу пороговых значений

- `file=` (имя файла)
- `dbic=` (канал)
- `dbob=` (растровая маска. Если пустой – будет создана новая)
- `tval=` (минимальное и максимальное пороговые значения. Например, 1,10 – маска получит значения 1 только если значения канала равны или больше 1 и при этом равны или меньше 10)
- `dbsn="` (имя нового сегмента – обязательное поле)

`r thr`

RANDBIT – простая команда для создания случайной выборки пикселей

Создаёт выборку на основе существующей растровой маски. Подробнее смотрите в справочной системе EASI.

Команда MODEL и язык моделей

Помимо прямого использования арифметических и логических операторов, вводимых из командной строки или записанных в виде скрипта, в EASI используется специальный язык моделей, сходный по синтаксису с самим EASI. Модели для выполнения записываются в файлы с расширением .mod и вызываются на выполнение командой MODEL:

- file=" (имя файла)
- source=" (имя файла модели *.mod)

```
r model
```

В файле модели имя файла не указывается. Таким образом, одна и та же модель может быть применена к множеству файлов с одинаковым набором каналов.

Основное отличие языка моделей от языка EASI – необходимость разделения строк точкой с запятой:

```
%1=%2+10;
```

Запись условия:

```
if (%1>100) then  
%2=3;  
%3=4;  
endif;
```

Встроенные функции работают как в языке EASI:

```
%10= max(%1,%2,%3);
```

Декларировать переменные в моделях не нужно. Скалярные переменные обозначаются знаком «#»:

- #a=100; — переменная равна константе
- #a=%1; — переменная равна значению канала 1
- if (#a>0) and (#b>0) then %2=3; endif; — использование переменных в условии
- #test=max(%1,%2,%3); — создание временной переменной и запись в нее результата вычисления

Текстовые переменные (если они нужны) обозначают знаком «\$»:

```
$a="test"
```

Другой вариант использования модели: ввод модели из командной строки или из скрипта. Для этого перед началом записи модели введем выражение:

```
model on " (имя файла)
```

Далее поместим текст модели и завершим модель:

```
endmodel
```

ОБРАТИТЕ ВНИМАНИЕ: если модель вводится таким образом из командной строки или в скрипте, использование точки с запятой в конце строки не допускается!

Фильтры

Создание фильтров на языке EASI основано на использовании специальных переменных, определяющих координаты пикселя в единицах раstra: @x, @y (см. выше). Запись фильтра, вычисляющего среднее значение канала 4 в плавающем окне 3x3 пикселя выглядит так:

```

model on "(имя файла)

%7 = (%4@x-1,@y-1 + %4@x,@y-1 + %4@x+1,@y-1 + \
%4@x-1,@y + %4@x,@y + %4@x+1,@y
+ \
%4@x-1,@y+1 + %4@x,@y+1 + %4@x+1,@y+1 ) / 9

endmodel

```

Фильтры, заданные таким образом, корректно обрабатывают края изображения, дублируя значения из области раstra за пределы изображения.

Существует набор специальных команд для выполнения стандартных операций фильтрации.

FAV – Вычисление среднего в плавающем окне

- file=" (имя файла)
- dbic= (входной канал)
- dboc= (выходной канал)
- flsz= (размер плавающего окна. Например: 3,3)
- mask= (опционно: маска для выполнения операции)

```
r fav
```

FME – Вычисление медианы в плавающем окне

- file=" (имя файла)
- dbic= (входной канал)
- dboc= (выходной канал)
- flsz= (размер плавающего окна. Например: 3,3)
- mask= (опционно: маска для выполнения операции)

```
r fme
```

FED – Фильтр для обнаружения границ (edge detection)

Вычисляет среднее модулей разности значений пикселей окна с центральным пикселем.

- file=" (имя файла)
- dbic= (входной канал)
- dboc= (выходной канал)
- flsz= (размер плавающего окна. Например: 3,3)
- mask= (опционно: маска для выполнения операции)

```
r fed
```

FMO – Вычисление моды

Этот фильтр аналогичен majority filter в ERDAS Imagine.

- file=" (имя файла)
- dbic= (входной канал)
- dboc= (выходной канал)
- flsz= (размер плавающего окна. Например: 3,3)
- mask= (опционно: маска для выполнения операции)

```
r fmo
```

Существуют два дополнительных полезных параметра этой команды:

- thinline="on/off

При использовании фильтра 3x3 значение «on» (по умолчанию) означает, что вычисление моды будет производиться только тогда, когда число пикселей класса, которому принадлежит центральный пиксель окна, меньше 3. Это позволяет при фильтрации тематических карт сохранять линейные объекты.

- keepvalu=

Значения классов, фильтрация которых не производится.

Дополнительные команды фильтрации (описание команд смотрите в справочной системе EASI):

- TEX – вычисление текстурных функций
- FSHARP – Увеличение контраста изображения
- FGA – Гауссовское сглаживание

Гистограммы значений

HIS – Графический вывод гистограммы значений по каналу

- file= (имя файла)
- dbic= (номер канала)
- mask= (опционально: маска для выполнения операции)
- report= (вывод на экран или имя файла для вывода в файл)

```
r his
```

Существует несколько дополнительных параметров, настраивающих вид гистограммы. Описание дополнительных параметров команды смотрите в справочной системе EASI.

Помимо графического или табличного представления гистограммы значений, будет выведено среднее, медиана, мода, максимум и минимум значений, а так же число пикселей, использованных для расчета. Поскольку формат вывода стандартный, результаты вывода в файл могут быть обработаны простыми программами для создания таблиц значений. Пример обработки файлов с использованием программ на языке PERL приведен в [4-ой статье](#).

HISDUMP – Вывод гистограммы значений в виде таблицы

- file= (имя файла)
- dbic= (список каналов)
- mask= (опционально: маска для выполнения операции)
- report= (вывод на экран или имя файла для вывода в файл)
- hisform= (формат гистограммы)

```
r hisdump
```

Значения параметра hisform определяют формат таблицы:

- raw – одна колонка для каждого канала, показывающая частоту встречаемости каждого класса
- center – то же плюс дополнительная колонка, показывающая значение, встречаемость которого вычисляется
- hitcenter – то же, значения с нулевой встречаемостью опущены

Описание дополнительных параметров команды смотрите в справочной системе EASI.

Работа с векторными данными

Векторные данные обычно хранятся внутри рабочих файлов PCIDSK в виде сегментов. Для этого проекция вектора должна совпадать с проекцией растровой базы данных. Если Вы импортировали векторный файл в PCIDSK из формата shp, скажем, в географических координатах, то для переноса векторного сегмента в рабочий растровый файл его надо спроецировать командой VECPRO.

VECPRO – Изменение проекции векторных сегментов

Краткий вариант использования команды

- fili= (имя файла с векторным сегментом)
- filo= (имя файла с растровыми данными, в которые переносится сегмент)
- dbvs= (номер сегмента. Если векторный файл импортирован с помощью команды FIMPORT, номер сегмента = 2)

r vecpro

Программа произведет трансформацию на основе сегментов привязки файлов.

Дополнительные параметры команды (смотрите в справочной системе EASI) позволяют производить трансформацию по выбранным сегментам привязки.

GRDPOL – Растеризация полигональных векторных данных

- filv=" (имя файла с векторным сегментом. Если сегмент находится в рабочем файле, параметр не заполняем)
- dbvs= (номер векторного сегмента)
- file=" (рабочий файл)
- dboc= (канал, в который производим растеризацию)
- fldnme=" (имя поля, значения которого будет присвоено пикселям)
- algo=" (алгоритм растеризации. Для полигональных объектов лучше использовать polygon)

r grdpol

Обратите внимание, что, если номер выходного канала не задан, будет создано новый растровый сегмент и произведена растризация в него. При этом значения поля учитываться не будут.

KRIGING – Интерполяция точечных векторных данных

Подробнее смотрите в справочной системе EASI.

RTV – Векторизация растровых данных

- file=" (рабочий файл)
- dbic= (векторизуемый канал)
- dbsn=" (имя сегмента – обязательное поле)
- polyinfo=" (создание линий или меток. Обычно используют оба – both)
- border=" (отрисовка линий границ растра. Обычно «on»)

r rtv

Дополнительные команды (описание команд смотрите в справочной системе EASI):

- VECBUF – Создание растровых масок в виде буфера линейных и точечных объектов векторного слоя
- CONTOUR – Создание векторного контура на основе растровых данных (обычно применяется для ЦМР)
- VECMERG – Объединение векторных данных из разных сегментов
- VIMAGE – Расчет зональной статистики по растру в пределах векторных объектов и запись статистики в базу данных векторного сегмента

Краткий перечень специальных команд анализа изображений

Подробное описание команд смотрите в справочной системе EASI.

Преобразование изображений:

- LUT – Масштабирование значений по каналам на основе данных сегмента таблицы радиометрической коррекции (Look up Table)
- MATCH – «Подгонка» гистограмм изображений
- RADNORM – радиометрическая нормализация перекрывающихся снимков
- PCA – Вычисление главных компонент
- TASSEL – Преобразование tasseled cap для Landsat MSS, TM и Landsat 7
- PANSHARP – Увеличение контрастности многоканального изображения с использованием панхроматического канала высокого разрешения

Работа с ЦМР (DEM):

- ASP – Расчет аспекта
- SLP – Расчет склона

- ARE – Расчет площади в пределах растровой маски с использованием ЦМР

Пространственная статистика:

- IPG – Создание базы растровых полигонов. Используется правило объединения 8-соседей
- POG – Расчет площади и периметра растровых полигонов
- SIEVE – Объединение полигонов по заданному критерию минимальной площади

Кластеризация:

- KCLUS — Кластеризация методом K-means
- FUZCLUS – Кластеризация методом Fuzzy K-means
- ISOCLUS — Кластеризация методом ISODATA
- NGCLUS – Непараметрическая кластеризация

Классификация:

- MLC – Непараметрическая (параллелепипед) или параметрическая (максимальное подобие) классификация с обучением
- MINDIS – Классификация по методу наименьшего расстояния
- KNN – Классификация методом k-ближайших соседей (K-Nearest Neighbors method)

Ссылки по теме

- [Основы макроязыка EASI \(PCI Geomatica\). Часть 1. Управление данными](#)
- [Основы макроязыка EASI \(PCI Geomatica\). Часть 3. Географическая привязка и проекции](#)
- [Основы макроязыка EASI \(PCI Geomatica\). Часть 4. Примеры автоматизации процессов](#)

[Обсудить в форуме](#) Комментариев — 0

Последнее обновление: 2014-05-15 00:08

Дата создания: 15.08.2007

Автор(ы): Пётр Потапов