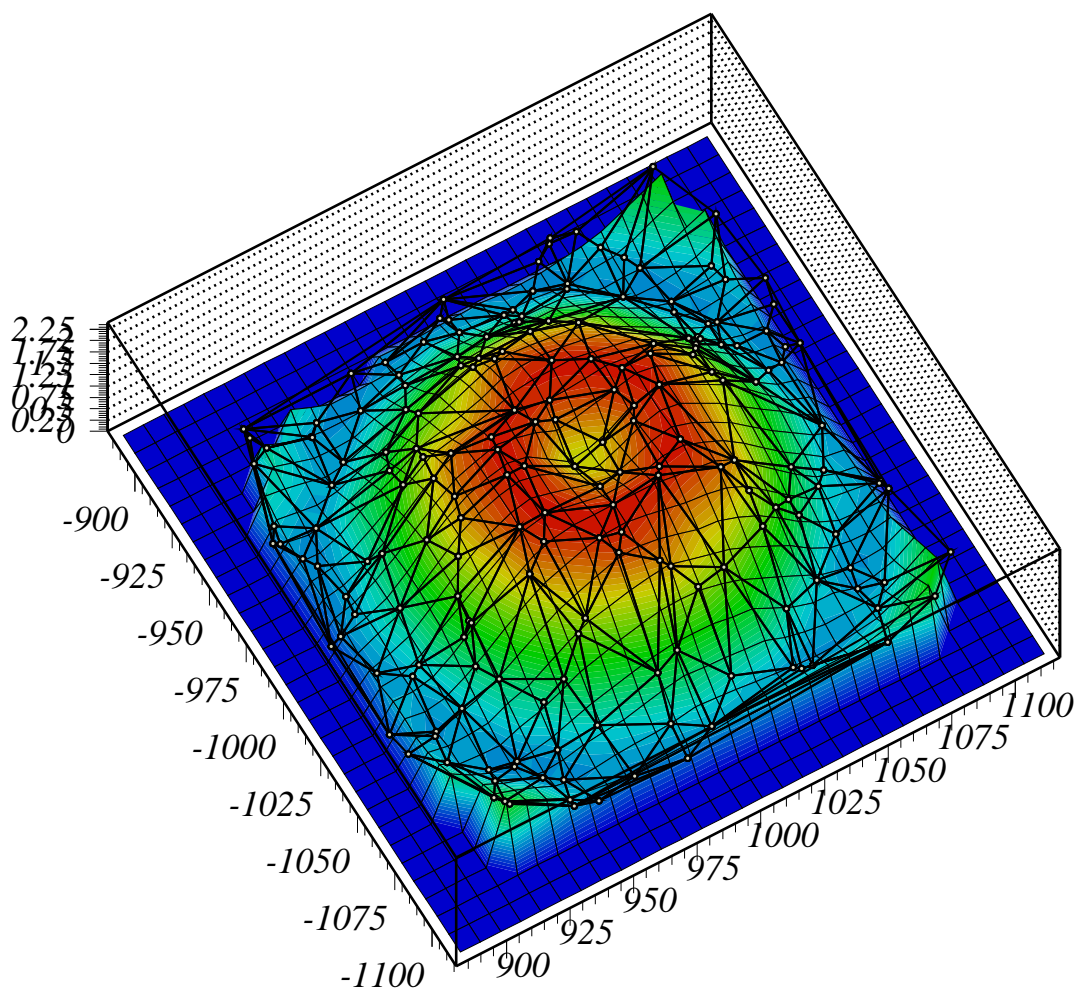


Physics Analysis Workstation (PAW)

© Е.М. Балдин*



Del Trirx ry rz 200

*e-mail: E.M.Baldin@inp.nsk.su

Скрипт, который создал картинку, взят с <http://paw.web.cern.ch/paw/contributions/>. Автор скрипта Luke Jones.

Оглавление

1	Знакомство с PAW	1
1.1	Введение	1
1.2	Немного истории	1
1.3	Запускаем PAW	2
1.4	Объекты PAW	4
1.5	Помогите или Help!!!	6
1.6	«Командная логика»	7
1.7	Интерпретатор FORTRAN (COMIS)	8
1.8	Файл инициализации	9
1.9	Проблемы	10
1.10	Литература	11
2	PAW tutorial	13
2.1	Простейший анализ	13
2.2	Ntuple	20
2.3	Гистограммы	21
2.4	Функции	24
2.5	Заключение	26

Эта статья была опубликована в 11ом (июльском) номере русскоязычного журнала Linux Format (<http://www.linuxformat.ru>) за 2006 год. Статья размещена с разрешения редакции журнала на сайте <http://www.inp.nsk.su/~baldin/> и до декабря месяца все вопросы с размещением статьи в других местах следует решать с редакцией Linux Format.

Текст, представленный здесь, не является точной копией статьи в журнале. Текущий текст в отличии от журнального варианта корректор не просматривал. Все вопросы по содержанию, а так же замечания и предложения следует задавать мне по электронной почте <mailto:E.M.Baldin@inp.nsk.su>.

Текст на текущий момент является просто *текстом*, а не книгой. Поэтому результирующая доводка в целях улучшения восприятия текста не проводилась.

1 Знакомство с PAW

1.1 Введение

PAW или Physics Analysis Workstation — это интерактивная программа анализа и графического представления результатов, с возможностью автоматизации посредством написания скриптов.

PAW позволяет работать с большими объёмами данных. Данные в основном представлены в виде списка не связанных между собой событий (ntuple). Каждой событие представляет из себя набор именованных полей. Данные можно представить в виде одномерных или двумерных гистограмм. В PAW реализована фильтрация событий по условию. Для сравнения экспериментальных данных с теорией PAW предоставляет интерфейс к чрезвычайно мощной и гибкой процедуре подгонки и оценки ошибок (MINUIT).

1.2 Немного истории

Пакет PAW был создан для целей физики элементарных частиц (ФЭЧ). История PAW берёт начало в 1986 году в европейском центре ядерных исследований (CERN¹). В процессе разработки основной упор делался на работу с очень большим объёмом² данных.

Для целей физического анализа PAW распространялась свободно с самого начала, но только в 2000 году PAW стала доступна под GPL. Начиная с 2002 года PAW является одним из пакетов дистрибутива GNU/Linux Debian, что в определённой степени гарантирует лицензионную чистоту.

По сути PAW является интерфейсом к набору библиотек CERNLIB. Всё, что можно сделать в PAW интерактивно, также можно реализовать в компилируемом программном коде. В PAW встроен интерпретатор языка программирования Fortran.

В последнее время появилась мода на отрицание Fortran. Что-то в этом, возможно, есть. Выдавить Fortran полностью, естественно, не удастся, так как он имеет ряд преимуществ: исключительная простота и очень большие объёмы наработанного надёжного кода, но для популяризации проекта приходится выбирать «более другой» язык программирования. В частности поэтому в 1994 году лидер команды PAW Рене Брун (René Brun) запустил проект «переписывания» PAW/CERNLIB на

¹Organisation Européenne pour la Recherche Nucléaire www.cern.ch — здесь же сэр Тим Бернерс-Ли работал, когда создавал WWW.

²Именно «очень большим объёмом», так как фраза «большой объём» недостаточно точно описывает случай PAW :)

C++ и назвал его ROOT. С начала 2006 года ROOT (<http://root.cern.ch/>) стал выпускаться под лицензией GNU, и, возможно, скоро попадёт в основные дистрибутивы GNU/Linux.

Чем же интересно PAW сейчас, когда есть ROOT? Во первых нет необходимости менять PAW в тех проектах, где оно уже прижилось. Для целей интерактивного анализа PAW эффективнее чем ROOT из-за излишней многословности последнего (наследие c++). Многие базовые идеи, которые были заложены в фундамент PAW достаточно интересны в плане изучения и копирования. Программы для того и делаются свободными, чтобы идеи лежащие в их основе не умирали.

1.3 Запускаем PAW

Очевидно, что перед запуском необходимо поставить этот пакет. Для пользователей дистрибутивов GNU/Linux Debian и Alt процесс установки выглядит примерно так:

```
> sudo apt-get install paw
```

Если же PAW не является составляющей вашего дистрибутива, то необходимо скачать исходники или готовые пакеты под вашу платформу, например, с официального ЦЕРНовского сайта <http://paw.web.cern.ch/paw/>. PAW идёт как составляющая пакета CERNLIB. Подробности о сборке PAW/CERNLIB можно найти на страничке <http://people.debian.org/~kmcarty/cernlib/> — CERNLIB on Debian. Её поддерживает Кэвин Макарти (Kevin McCarty).

После установки открываем X-терминал и запускаем PAW:

```
> paw
*****
*
*           W E L C O M E           t o   P A W           *
*
*           Version 2.14/04           12 January 2004           *
*
*****
Workstation type (?=HELP) <CR>=1 :
```

При запуске PAW задаёт традиционный вопрос о выборе типа окна для отображения. Раньше это было достаточно актуально. На сегодня интерес представляет обычное X-окно (появляется по умолчанию), или буквенно-цифровой терминал. В последнем случае результаты выводятся с помощью текста, то есть для представления данных достаточно окна терминала.

Во время запуска paw считывает и выполняет команды из файла `~/pawlogon.kumac`. Получив после инициализации приглашение можно приступить к работе. Сказать «Hello World» из PAW можно следующим образом:

```
PAW > message Hello World
```

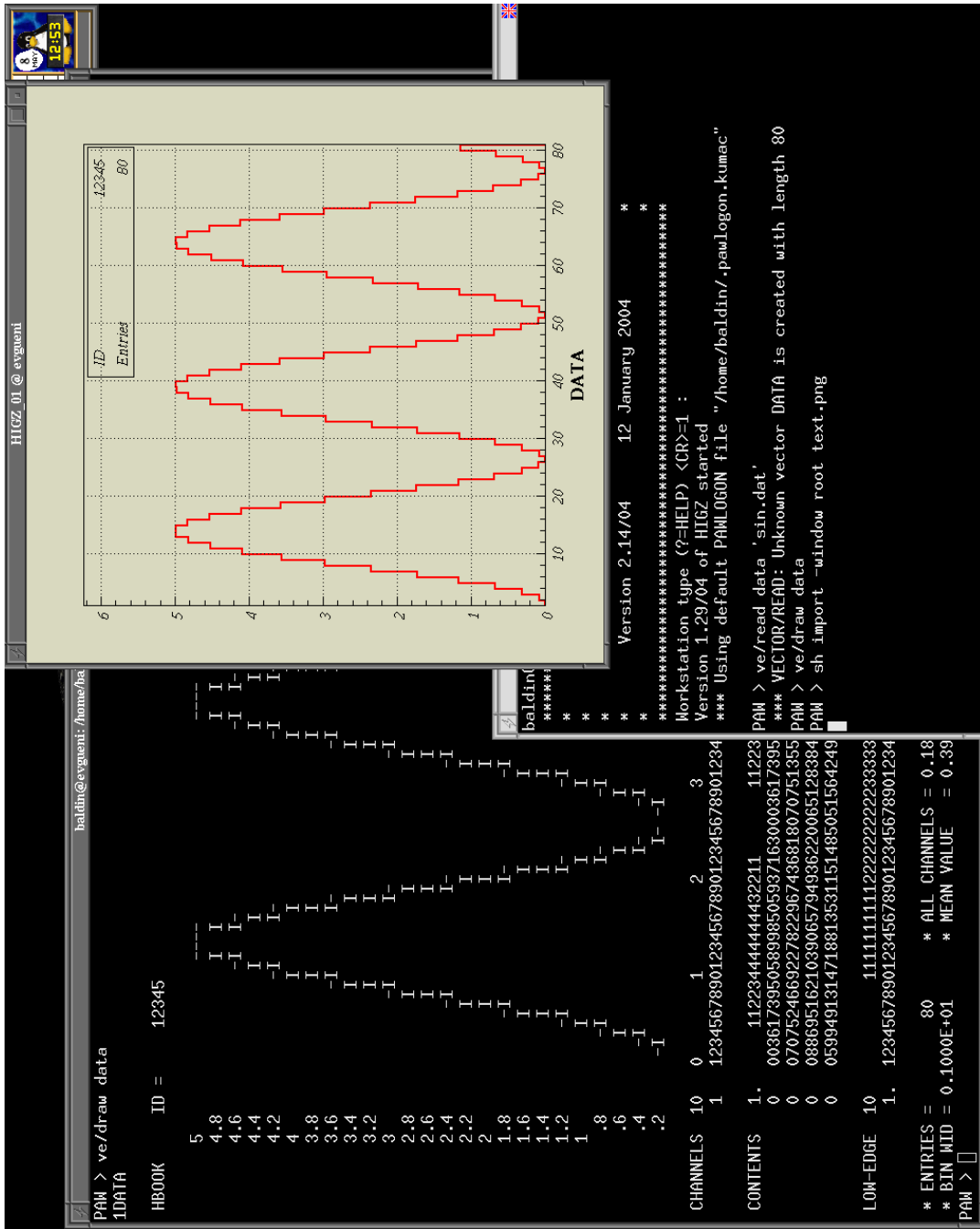


Рис. 1.1. Буквенно-цифровое и X-Window представление данных

Hello World

Чтобы окончить сеанс достаточно набрать команду `exit`. В случае, если PAW удалось тем или иным образом «заикнуть», то на помощь приходит прерывание по `^C`.

1.4 Объекты PAW

PAW оперирует векторами, гистограммами, `ntuple`'ами и `cut`'ами. Эти объекты являются для PAW базовыми.

Вектора представляют из себя одномерные или многомерные массивы чисел. Для работы с векторами предоставляется интерфейс к пакету для манипуляций с массивами SIGMA и интерпретатору Fortran COMIS. Вектора используются для анализа небольших объёмов текстовых данных.

Гистограмма — это базовый объект для анализа в PAW. Гистограмма представляет из себя столбчатую диаграмму — один из видов графического представления эмпирических распределений. Чтобы создать (`book`) гистограмму достаточно определиться с пределами гистограммы и числом элементарных бинов (`bin`). При заполнении (`fill`) гистограммы в соответствующий записываемому числу бинов добавляется единица с определяемым пользователем весом (`weight`). В PAW любой гистограмме может приписываться подгоночная функция (`fit`) с результатами подгонки.

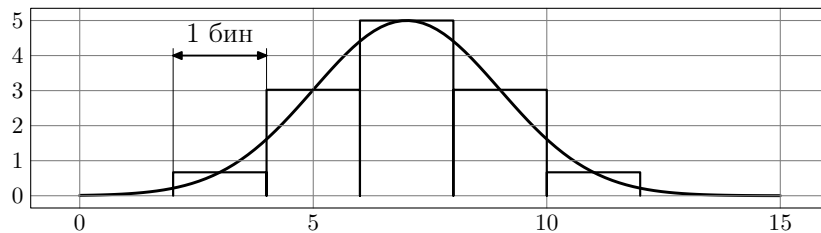


Рис. 1.2. Пример гистограммы и «родительской» функции распределения

Двумерная гистограмма является обобщением обычной одномерной гистограммы на плоскости. В отличие от одномерной гистограммы при отображении двумерной возникает серьёзная проблема выбора графического представления данных. Если есть возможность, то лучше всё сводить к одномерным гистограммам.

`Ntuple` (произносится как «энтапл») представляет из себя коллекцию независимых событий. Каждое событие представляет из себя набор именованных параметров. `Ntuple` по сути дела обычная таблица, в которой каждая строчка соответствует событию, а столбец переменной.

```
# открытие файла ee-ang.rz
PAW > hi/fil 1 ee-ang.rz
# вывод информации об ntuple с идентификатором 1
# ntuple был хранится в ee-ang.rz
```

```
PAW > nt/print 1
*****
* NTUPLE ID=      1  ENTRIES= 10673  ee angles
*****
*  Var numb  *   Name    *   Lower    *   Upper    *
*****
*      1    *  th1      * 0.473953E+00 * 0.173814E+03 *
*      2    *  th2      * 0.741767E+01 * 0.173814E+03 *
*      3    *  phi1     * 0.000000E+00 * 0.359949E+03 *
*      4    *  phi2     * 0.000000E+00 * 0.359957E+03 *
*      5    *  p1       * 0.131558E-01 * 0.579733E+04 *
*      6    *  p2       * 0.259194E-01 * 0.156410E+03 *
...
*      24   *  ntrk     * 0.000000E+00 * 0.900000E+01 *
*****
```

Миллион событий и тысячи переменных в `ntuple` это вполне рядовое явление. Физически `ntuple` представляет из себя файл с прямым доступом. Все операции оптимизированы для максимизации скорости. Для работы с `ntuple` отсутствует встроенный язык по типу SQL — так как PAW это узкоспециализированный инструмент. Объём данных имеет тенденцию к увеличению со временем во всех областях знаний, поэтому подобные методы обработки следует взять на заметку.

`Cut` (произносится как «кат») это булева функция от именованных переменных `ntuple`. Каты используются для выборки подмножества событий удовлетворяющих условию для заполнения гистограммы.

```
# открытие файла ee-ang.rz
PAW > hi/fil 1 ee-ang.rz
# создание катa по углам с идентификатором 1
PAW > nt/cut 1 th1>30.and.th1<150.and.th2>30.and.th1<150
# создание катa с требованием на тип частиц с
# идентификатором 2, 11 соответствует e-, -11 — e+
PAW > nt/cut 2 f1=11&&f2=-11
# запуск скрипта для открытия eps-файла
PAW > epsopen th1.eps
*****
File: th1.eps      Xsize: 20.0 cm      Ysize: 20.0 cm      LUN: 99
*****
# деление графического пространства на две зоны
PAW > zone 1 2
# создание гистограмм из ntuple с идентификатором 1
# фильтрация событий производится по первому катy
PAW > nt/plot 1.th1 $1
# фильтрация событий производится по обоим катам
PAW > nt/plot 1.th1 $1.and.$2
```

```
# закрытие eps-файла
```

```
PAW > epsclose
```

```
*****
```

```
File "th1.eps" closed.
```

```
*****
```

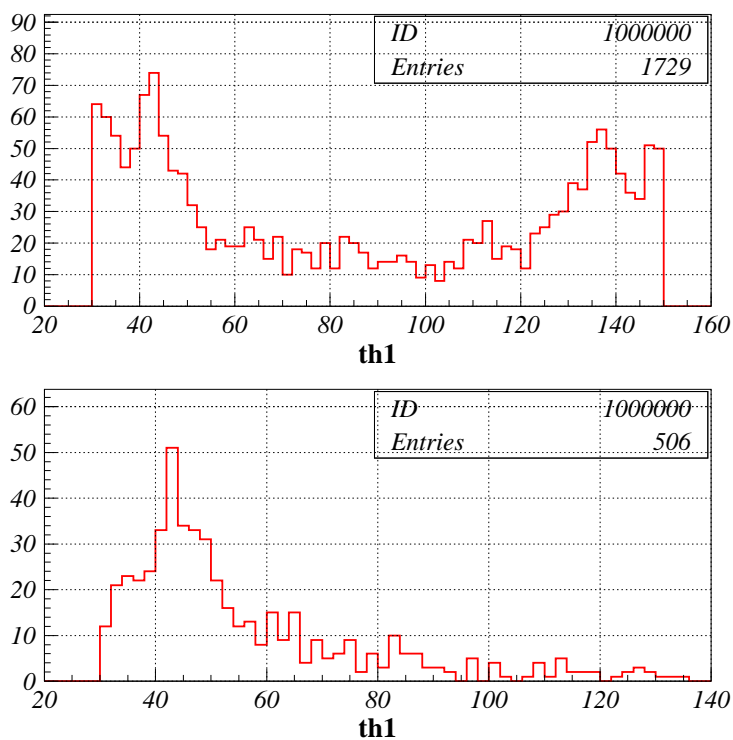


Рис. 1.3. Гистограммы по углу θ_1 без отбора и только для e^+e^- -пары. Данные взяты из файла ee-ang.rz.

1.5 Помогите или Help!!!

Сразу после запуска PAW можно вводить команды. Вероятно, одной из самых полезных команд является команда `help`:

```
PAW > help
```

```
From /...
```

- 1: KUIP Command Processor commands.
- 2: MACRO Macro Processor commands.

- 3: VECTOR Vector Processor commands.
- 4: HISTOGRAM Manipulation of histograms.
- 5: FUNCTION Operations with Functions. Creation and plotting.
- 6: NTUPLE Ntuple creation and related operations.
- 7: GRAPHICS Interface to the graphics packages HPLOT and HIGZ.
- 8: PICTURE Creation and manipulation of HIGZ pictures.
- 9: ZEBRA Interfaces to the ZEBRA RZ, FZ and DZ packages.
- 10: FORTRAN Interface to MINUIT, COMIS, SIGMA and
FORTRAN Input/Output.
- 11: NETWORK To access files on remote computers.
- 12: MLP Multi-Layer Perceptron (MLP).
- 13: OBSOLETE Obsolete commands

Enter a number ('0'=Top, '\ '=one level back, 'Q'=command mode):

Структура помощи в PAW отличается от привычных. Похожая система присутствовала в справке VAX/VMS. После ввода команды help выводится меню и предлагается выбрать интересующий вас пункт. Обычно, чтобы добраться до необходимой команды требуется пройти 2-4 уровня. Если известен путь до команды, то можно вызвать справку без выбора вариантов:

```
PAW > help hi/plot
Command "/HISTOGRAM/PLOT" :
```

Справка гарантировано работает на любых терминалах. Следует отметить, что более подробное встроенное описание вряд-ли где можно найти (разве что в VAX/VMS³). PAW/CERNLIB отличается очень подробной и качественной документацией, в смысле содержания. Там даже баги задокументированы :). Такое качество в частности было достигнуто с помощью встроенных прямо в PAW команд для редактирования описания.

1.6 «Командная логика»

Чуть выше было рассказано про иерархическое меню справки. Прежде чем добраться до описания команды необходимо пройти некий путь. Например, чтобы узнать как нарисовать двумерную гистограмму в виде LEGO, то надо выбрать сначала пункт 4 (HISTOGRAM), после пункт 11 (2D_PLOT), а затем пункт 2 (LEGO). Полная команда состоит из пути HISTOGRAM/2D_PLOT/, самой команды LEGO и опций, которые передаются с этой командой

```
# нарисовать двумерную гистограмму номер 10
PAW > HISTOGRAM/2D_PLOT/LEGO 10
```

Полное название команды набирать долго, даже не смотря на то, что есть история команд для текущего сеанса. Поэтому действует соглашение, что если команда

³В F.A.Q. fido7.ru.os.cmp вопрос номер 8 звучит так: А где это есть и круче? Ответ: В VMS!

уникальная, то можно опускать корневые пункты меню. В случае команды `LEGO`, можно писать `2D_PLOT/LEGO` или просто `LEGO`. Второе соглашение позволяет сокращать имена и пути команд до тех пор, пока они остаются уникальными, например, `HISTOGRAM/PLOT` сокращается до `hi/pl`, что сложнее запомнить, но набрать гораздо быстрее. Эти два соглашения позволяют увеличивать эффективность работы с ростом опыта.

Команды при выполнении обрабатываются командным процессором `KUIP` (`help kuip`).

1.7 Интерпретатор FORTRAN (COMIS)

Для любого сколько-нибудь серьёзного инструмента анализа необходим встроенный язык программирования. В случае PAW особого выбора не было. FORTRAN или FORmula TRANslator является традиционным языком для научных вычислений. Он очень прост и стабилен. Как следствие, существует очень большое количество устоявшихся кросс-платформенных математических библиотек. Существует мнение, что на чём бы в будущем физики не программировали и как бы этот язык не выглядел — он будет называться FORTRAN.

Вызов интерпретатора FORTRAN производится с помощью команды `COMIS`. После этого можно писать программу, например, вычисления чисел Фибоначчи:

```
PAW> comis
CS> ia=0
MND> ib=1
MND> type ia
MND> type ib
MND> do i=2,10
MND> ix=ia
MND> ia=ib
MND> ib=ix+ib
MND> type ib
MND> enddo
MND> end
*T IA = 0
*T IB = 1
*T IB = 1
*T IB = 2
*T IB = 3
*T IB = 5
*T IB = 8
*T IB = 13
*T IB = 21
*T IB = 34
*T IB = 55
```

```
PAW
CS> end
PAW >
```

Одна из основных проблем PAW заключается в различных языках для написания программ и для написания скриптов (COMIS против KUIP). И та и другая подсистема по отдельности хорошо справляется со своими проблемами, но серьёзные задачи в рамках какого-то одного подхода не решаются. Для нетривиального анализа необходимо взаимодействие и, как следствие, знание обоих языковых инструментов.

1.8 Файл инициализации

При запуске PAW, как уже упоминалось, считываются инструкции `~/pawlogon.kumac`. Файл инициализации представляет из себя скрипт для командного процессора KUIP. Скрипты PAW имеют расширение `.kumac` и выполняются с помощью команды `exec <имя файла без расширения>`.

Обычно, в файле инициализации прописываются настройки по умолчанию с помощью команды `set` или создаются сокращения команд:

```
alias/create ac alias/create
ac man help
```

Полезным является прописывание путей где лежат макросы, то есть аналог `$PATH`:

```
* запускать макросы в текущей директории и в ~/kumac
macro/default -AR '.',~/kumac'
```

При исполнении макросов из путей прописанных таким образом можно упускать команду `exec`.

Первые полезные макросы, которые следует создать и положить в `~/kumac`, это макросы открытия/закрытия eps-файла. Эти макросы использовались уже в примере с `ntuple`. Макросы созданы Димой Бондаревым:

```
*файл epsopen.kumac
*****
macro epsopen file=paw.eps xsize=20.0 ysize=20.0 lun=99
*****
glob/import PS_FILE,PS_LUN

mess *****
if ([PS_LUN].ne.'[PS_LUN]') then
  mess File "[PS_FILE]" already open!
else
  glob/create PS_FILE [file] 'Filename for output'
  glob/create PS_LUN [lun] 'Indicator of output file opening'
```

```

mess File: [file] Xsize: [xsize] cm Ysize: [ysize] cm LUN: [lun]
size [xsize] [ysize]
for/file [lun] [file]
gr/meta [lun] -113
endif
mess *****

exitm

```

```

*файл epsclose.kumac
*****
macro epsclose
*****
glob/import PS_FILE,PS_LUN

mess *****
if ([PS_LUN].ne.'[PS_LUN]') then
    close [PS_LUN]
    mess File "[PS_FILE]" closed.
    glob/del PS_FILE,PS_LUN
else
    mess No open output file!
endif
mess *****

exitm

```

1.9 Проблемы

При начале работы с PAW могут возникнуть некоторые проблемы связанные с тем, что программа писалась довольно давно и не рассчитана на текущие представления об удобстве.

Если графическое окно не сохраняет рисунок при перекрытии окна другим окном или при переключении между виртуальными экранами, то необходимо добавить в файл настройки X-Window (/etc/X11/XF86Config-4) в разделе Device добавить опцию:

```

Section "Device"
...
Option "BackingStore" "On"
...
EndSection

```

и перезапустить X-Window.

Следует учитывать, что пути до файлов не должны превышать 256 символов. Обычно это не проблема, но всякое бывает.

Приходится признать, что PAW с русским языком не дружит совсем. Чтобы добавить русские надписи, можно воспользоваться пакетом `psfrag`. В начале с помощью макросов `epsopen/epsclose` создаётся картинка в которую внедряются англоязычные метки. Например, метку `Petr` необходимо заменить на «Здесь будет город заложен...», а метку `moment` на «Я помню чудное мгновенье». Далее создаётся \LaTeX -файл вида:

```
\documentclass[12pt]{article}
\usepackage[warn]{mathtext}
\usepackage[T2A]{fontenc}
\usepackage[koi8-r]{inputenc}
\usepackage[english,russian]{babel}
\usepackage{psfrag}
\begin{document}
\pagestyle{empty}
\psfrag{Petr}{Здесь будет город заложен\ldots}
\psfrag{moment}{Я помню чудное мгновенье}
\includegraphics{«ваша картинка».eps}
\end{document}
```

Файл компилируется

```
> latex «файл».tex
> dvips -E «файл».dvi -o «файл».eps
```

и на выходе получается eps-файл с кириллическими надписями.

К преимуществам такого подхода можно отнести то, что в картинку можно внедрить абсолютно любую конструкцию, которую поддерживает \LaTeX , а к недостаткам то, что уж больно много возни. С другой стороны Вы в Linux, а следовательно написать скрипт, который будет это делать за Вас, труда не составит.

1.10 Литература

Приёмы работ с PAW в этом тексте пока не раскрыты, поэтому при желании всю информацию об этом пакете можно почерпнуть в документации. В любом случае, рано или поздно при использовании PAW её придётся прочитать. Вся стандартная документация к PAW существует в электронном виде⁴.

Отправная точка для поиска является официальная страничка в CERN, посвящённая в PAW <http://paw.web.cern.ch/paw/>. По состоянию странички можно заметить, что пакет к сожалению стареет — более двадцати лет, однако.

⁴Раньше единственный способ добыть печатную копию была посылка бандероли из CERN, благо доставка осуществлялась за счёт CERN. Возможно не одна тонна бумаги пересекла границу именно по этой причине. Если в округе кто-то пользовался PAW, то вы сможете найти твёрдую копию с тех времён.

С главной странички идёт ссылка PAW Reference manual http://paw.web.cern.ch/paw/reference_manual/. Там повторяется то, что можно получить по команде `help`.

По адресу <http://wwwasdoc.web.cern.ch/wwwasdoc/Welcome.html> можно найти всю официальную документацию. Следует учитывать, что документация создавалась для получения твёрдой копии, поэтому html версии не всегда соответствуют оригиналу. Для овладения технологией представляет интерес PAW Tutorial: <http://paw.web.cern.ch/paw/tutorial/>. В сети есть некоторое количество сайтов с информацией по PAW, но уж больно качественно была сделана официальная документация. Так что начинать надо с неё.

В дистрибутиве Debian, кроме самого `paw` можно установить пакет `paw-demos`:

```
> apt-cache search paw-demos
paw-demos – Physics Analysis Workstation examples and tests
```

Для демонстрации возможности PAW и для тестирования Вашей версии этого пакета достаточно запустить скрипт `paw-demos` (`man paw-demos`).