

Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Оренбургский государственный университет»

Кафедра прикладной информатики в экономике и управлении

Н.Ф. Панова

ОСНОВЫ РАБОТЫ В ОПЕРАЦИОННОЙ СИСТЕМЕ LINUX

Рекомендовано к изданию Редакционно-издательским советом федерального государственного бюджетного образовательного учреждения высшего образования «Оренбургский государственный университет» в качестве методических указаний для студентов, обучающихся по программам высшего образования по направлению подготовки 09.03.03 Прикладная информатика

Оренбург
2017

УДК 004.451(076.5)
ББК 32.973.26-018.2я7
П16

Рецензент - кандидат технических наук А.Л. Коннов

Панова, Н.Ф.

П16 Основы работы в операционной системе Linux: методические указания / Н.Ф. Панова; Оренбургский гос. ун-т.- Оренбург : ОГУ, 2017. - 37 с.

Основное содержание: история создания операционной системы Linux; описание основных ее подсистем; команды для работы в терминальной режиме; обзор и функциональные возможности графических оболочек; задания на лабораторные работы; контрольные вопросы.

Методические указания «Основы работы в операционной системе Linux» предназначены для студентов очной и заочной форм, обучающихся по программам высшего образования по направлению подготовки 09.03.03 Прикладная информатика.

УДК 004.451(076.5)
ББК 32.973.26-018.2я7

© Панова Н.Ф., 2017
© ОГУ, 2017

Содержание

Введение.....	4
1 История создания операционной системы Linux.....	5
2 Файловая система.....	7
2.1 Структура каталогов.....	7
2.2 Размещение на диске и защита.....	8
2.3 Команды для работы с файлами и каталогами.....	10
2.4 Связи между файлами.....	13
2.5 Связи между файлами.....	14
2.6 Перенаправление ввода-вывода.....	15
2.7 Лабораторная работа №1.....	17
3. Управление процессами.....	18
3.1 Характеристики процессов.....	18
3.2 Лабораторная работа №2.....	20
4 Графические оболочки Linux.....	21
4.1 Обзор графических оболочек.....	21
4.2 Графическая оболочка KDE.....	27
4.3 Лабораторная работа №3.....	35
Список использованных источников.....	37

Введение

Методические указания предназначены для проведения практических и лабораторных работ по курсу «Операционные системы» для студентов вторых курсов направления 09.03.03 Прикладная информатика. Методические указания содержат теоретический материал по истории создания операционной системы, краткую характеристику основных подсистем, таких как файловая подсистема и подсистема управления процессами. В методических указаниях также описаны самые часто используемые простейшие команды, основные команды для работы с файлами и каталогами, а также процессами. Указания содержат информацию о графических оболочках, доступных для пользователей Linux, описание функциональных частей оболочки KDE.

Методические указания способствуют формированию обще-профессиональных и профессиональных компетенций, таких как ОПК-3: способность использовать основные законы естественнонаучных дисциплин и современные информационно-коммуникационные технологии в профессиональной деятельности. Методические указания могут быть полезны студентам и преподавателям других направлений, поскольку операционная система Linux является свободным программным продуктом и используется достаточно широко в различных сферах.

1 История создания операционной системы Linux

Человеком, положившим начало Linux, является Л. Торвальдс. Будущий гений родился 28 декабря 1968 года в Финляндии. Будучи ребенком, он активно интересовался точными науками, также будущий программист частенько играл с калькулятором своего деда, являвшегося на тот момент профессором статистики в Хельсинкском университете.

В 10-летнем возрасте Торвальдс занимался написанием программ, работая на компьютере дедушки. В 1989 году Линус начинает получать высшее образование. В это же время на конференции ассоциации Usenix AT&T было объявлено о выпуске новой и дорогой системе UNIX System V. В ответ профессор Таненбаум занялся написанием усеченной версии Unix – Minix. Она могла работать на персональных компьютерах. Торвальдс меньше, чем за месяц изучил Minix и стал изучать материалы технических конференций. Новая операционная система была вместо учебника в мире UNIX, по этой причине в ней было довольно много ограничений, которые не могли исправить ни патчи, ни отдельно написанные программы. Линусу многое не нравилось в Minix, особенно, эмуляция терминала, и он решил вплотную заняться собственной разработкой терминала.

Когда эмулятор был почти готов, Торвальдс решил добавить новые возможности в виде функций upload и download. В свою очередь, они требовали новой файловой системы и драйвера для дисководов. Программист выбрал довольно трудный путь и все силы отдавал для достижения заветной цели. У Линуса Торвальдса была прекрасная черта: он не мог бросить работу недоделанной, что и привело его к успеху.

Спустя время, новая система скорее напоминала готовую операционную систему, а не самодельную терминальную программу, в чем прямая заслуга Линуса.

В сентябре 1989 года оболочка будущей операционной системы успешно заработала. Торвальдс называл ее Linux, но официальным названием было Freax.

Название Linux понравилось Ари Лемке (преподаватель, согласившийся выделить место на институтском компьютере для системы), и скоро появилась

первая версия системы - 0.01. Эту версию мало кто использовал – она была еще очень «сырой», и чтобы заставить ее работать, нужно было потратить не один вечер.

В октябре вышла версия Linux 0.02, в ноябре – 0.03. В конце ноября Linux стала полностью автономной, что вывело ее на новый уровень. К тому времени, как вышла версия Linux 1.0, система стала известной во всем мире и заинтересовала предпринимателей. Благодаря своей гибкости и потенциалу, Linux получила широкое распространение, в том числе и как серверная ОС.

Linux стала поддерживаться многими хакерами со всего мира, в результате чего появилось огромное количество улучшений в виде дистрибутивов.

Между Таненбаумом и Торвальдсом в 1992 г. развернулась дискуссия о недостатках Linux. Профессор считал Linux устаревшей системой из-за отказа Линуса от микроядра в пользу моноядра. Линус был согласен с тем, что микроядро — хорошее решение, но настаивал на идее моноядра, объясняя свой выбор хорошей эффективностью и отсутствием проблем с переносимостью.

В 1997 г. Линус Торвальдс с семьей переехал в Америку, где основал собственную компанию Transmeta, которая специализировалась на разработке процессоров. Спустя некоторое время создатель Linux решил отойти от производства микрочипов и занялся доработками главного детища всей своей жизни.

Ему удалось совершенствовать Linux. Популярность этой операционной системы постоянно и неуклонно растет. У системы Linux существуют весомые плюсы и минусы.

К плюсам можно отнести разнообразие дистрибутивов для пользователя и гибкость, возможность настройки «под себя» системы в целом, а к минусам – достаточную сложность освоения и малое количество доступных приложений, игр, поскольку доля данной ОС мала для заинтересованности разработчиков.

Сейчас эту ОС вполне реально использовать с достаточным опытом обращения, а через некоторое время она сможет тягаться с Windows на равных.

2 Файловая система

2.1 Структура каталогов

Файловая система Linux организована в виде древовидной структуры каталогов, внутри которых находятся файлы и подкаталоги. Обычно файловая система имеет вид, показанный на рисунке 1. Физические, а также логические устройства в файловой системе не упоминаются. При загрузке системы происходит выбор, какое из запоминающих устройств окажется в качестве корневого каталога. Другие устройства внешней памяти подключаются в процессе работы как каталоги внутри корневой файловой системы.

В Linux выделяют 3 класса файлов:

- обычный дисковый файл – файлы любого формата с пользовательской информацией и исполняемые, их наполнение меняет пользователь;
- каталог – папки, содержащие файлы и информацию о них;
- специальный файл – это файл символических связей, отображения ВУ (внешних устройств) и другие.

Корневой каталог обозначается в виде косой черты (/). Таким образом, имя корневого каталога не содержит имени физического устройства.

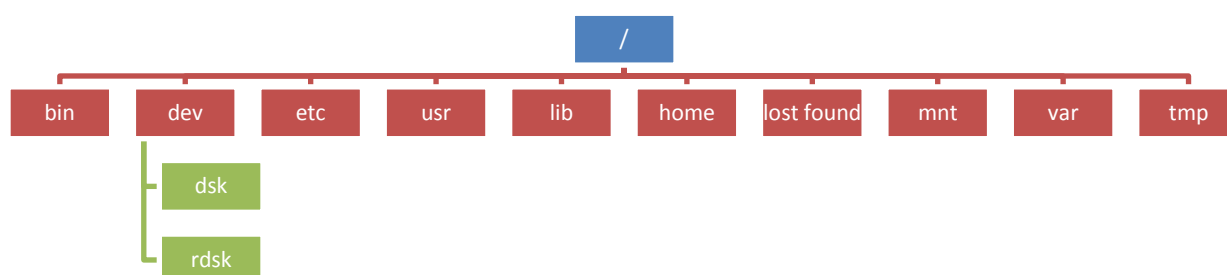


Рисунок 1 - Схема корневой системы каталогов.

Каталог bin содержит основные программы общего назначения, такие как команды и утилиты самой системы;

Каталог boot содержит файлы, используемые при загрузке системы;

Каталог `dev` содержит имена внешних устройств, с которыми можно работать, как с обычными файлами;

Каталог `etc` содержит команды администратора, различные конфигурационные файлы;

В каталоге `home` хранятся домашние каталоги пользователей;

Каталог `lib` содержит служебные файлы для работы программ – библиотеки;

Каталог `mnt` или `media` содержит каталоги для подключения устройств, файловых систем;

Каталог `sbin` хранит программы специального назначения;

Каталог `tmp` предназначен для хранения временных файлов;

В каталоге `usr` расположены прикладные программы и их данные;

Каталог `var` используется самой операционной системой, пользователи не имеют доступа к нему.

Максимальный размер имени файла составляет 65535 символов. Регистр написания также имеет значение. Имя файла может содержать любые символы, кроме кривой черты. Если имя файла содержит специальные символы, тогда их надо обособлять обратной кривой чертой или заключать в кавычки. Файл, имя которого начинается с точки, считается скрытым и не отображается обычными командами просмотра каталогов.

Linux не выделяет специально расширение в имени файла. Вместо термина "расширение" в Linux используется термин "суффикс". Использование или неиспользование суффиксов остаётся на усмотрение пользователя. При необходимости определить тип документа и связанное с ним приложение можно не по суффиксу, а по содержимому файла (по нескольким первым байтам, если это возможно).

2.2 Размещение на диске и защита

Информация на дисках размещается поблочнo, по 512 байт в каждом блоке, блок равен сектору. Диск разбивается на следующие области:

– неиспользуемый блок;

- управляющий блок или суперблок, в котором содержится размер диска и границы других областей;
- i – список, состоящий из описаний файлов, называемых i –узлами или индексными дескрипторами:
- область для хранения содержимого файлов.

Каждый индексный дескриптор содержит:

- идентификационный номер владельца;
- идентификационный номер группы владельцев;
- права доступа;
- физические адреса на диске, где находится содержимое файла;
- размер файла;
- время создания файла;
- время последней модификации файла;
- время последнего изменения атрибутов;
- число ссылок на файл;
- тип файла: каталог, обычный файл или специальный файл.

Следом за i –списком идут блоки, предназначенные для хранения файлов. Пространство на диске, оставшееся свободным от файлов, образует связанный список свободных блоков.

Защита файлов осуществляется при помощи идентификатора пользователя и десяти битов защиты – прав доступа. Права доступа подразделяются на три типа:

- чтение (read);
- запись (write);
- исполнение (execute).

Эти права доступа могут быть предоставлены трем классам пользователей:

- владельцу файла;
- группе, в которую входит владелец;
- всем прочим пользователям.

Атрибуты доступа определяют, что разрешено делать с данным файлом данной категории пользователей.

При создании файла модифицируется не сам файл, а каталог, в котором появляются новые ссылки на узлы. Удаление файла заключается в удалении ссылки. Право на создание и удаление файла – это право на запись в каталог.

Право на выполнение каталога интерпретируется как право на поиск в нем, прохождение через него. Оно позволяет обратиться к файлу по пути, содержащему данный каталог, даже тогда, когда каталог не разрешено читать и список всех его файлов недоступен.

2.3 Команды для работы с файлами и каталогами

Сегодня работа в Linux осуществляется без помощи командной строки. Многие основные действия можно выполнять, применяя графические приложения. Тем не менее, необходимо знать базовые команды и утилиты.

Многие действия с помощью командной строки можно выполнять гораздо быстрее, чем применяя графическую оболочку, стоит только знать нужные команды. Кроме того, утилиты командной строки зачастую более функциональны или не имеют графического подобия вовсе. При помощи командной строки можно делать многое, быстро и действенно, тут не придется разбираться в устройстве графического интерфейса или «где находится эта кнопка?».

Linux многопользовательская система, поэтому существует супер-пользователь - системный администратор, который имеет право регистрировать в системе других пользователей. Процесс регистрации подразумевает задание логина и пароля пользователя.

Команда Linux пишется в строке, которая начинается приглашением системы. Как правило, это символ '\$'.

В строке разрешается вводить несколько команд, разделив запятыми.

К простейшим командам относятся:

\$ date – вывод текущей даты;

\$ pwd – вывод текущего каталога, т. е. каталога, в котором находится пользователь;

\$ passwd – изменение пароля;

\$ who – вывод списка активных пользователей;

\$ man [i]– вывод справочного файла, где i – номер раздела справочного файла.

По умолчанию вывод осуществляется по разделу 1, который содержит описание команд.

Каждая команда может содержать несколько опций (параметров) в виде:

\$ <команды> [- <опции>]

Существует ряд команд для работы с файлами и каталогами:

\$ cat <файл-1>, <файл-2>, ... – показ содержимого файлов без задержки экрана;

\$ pr <файл-1>, <файл-2>, ... – вывод содержимого файлов на экран с задержкой экрана и возможностью листания экранов;

\$ more <файл-1>, <файл-2>, ... – вывод содержимого файлов на экран с задержкой экрана;

\$ ls <каталог> | <файл> – вывод на экран содержимого каталога или информации о заданных файлах. Команда может использовать опции:

–a – добавить в список «скрытые» файлы;

–s – сортировка файлов по размеру;

–t – сортировка файлов по дате изменения;

–l – вывод подробной информации по каждому файлу.

\$ cd <каталог> – смена текущего каталога;

\$ cd – возврат в домашний каталог пользователя;

\$ cp <файл-1>, <файл-2> – копирование файла файла;

\$ mv <файл-1>, <файл-2> – перемещение файла;

\$ mkdir <каталог> – создание нового каталога с заданным именем;

\$ ls — вывод имен файлов текущего каталога.

В команде могут использоваться следующие ключи:

–a – добавить в список «скрытые» файлы;

–s – сортировка файлов по размеру;

–t – сортировка файлов по дате изменения;

-l – вывод подробной информации по каждому файлу.

Например, приведенная ниже команда выводит простой список файлов в текущем каталоге:

```
$ ls
```

Экран вывода может иметь вид:

```
fonts.tar.gz linux.jpg report.odt template ubuntu.jpg
```

Добавим опции -a, -s, -l и в результате получим список всех файлов, включая скрытые, кроме того, они будут отсортированы:

```
$ ls -asl
```

```
fonts.tar.gz
```

```
.gtkrc-2.0
```

```
linux.jpg
```

```
report.odt
```

```
ubuntu.jpg
```

```
template
```

```
.filter
```

Команда ls с опцией -l позволяет увидеть полную информацию по каждому файлу.

```
$ ls -l
```

```
-rw-r--r-- 1 admin admin 217589 2009-02-12 13:35 fonts.tar.gz
```

```
-rw-r--r-- 1 admin admin 162348 2009-02-03 20:50 linux.jpg
```

```
-rw-r--r-- 1 admin admin 155847 2009-02-15 12:54 report.odt
```

```
-rw-r--r-- 1 root root 120736 2009-02-12 12:55 template
```

```
-rw-r--r-- 1 admin admin 134404 2009-02-12 12:24 ubuntu.jpg
```

Команда mv осуществляет переименование или перемещение указанных файлов.

```
$ mv <имя файла> <новое имя файла> – для переименования файла;
```

```
$ mv <имя файла> <каталог> – для перемещения
```

В команде mv можно также использовать параметр -i, чтобы защитить файл от перезаписи.

\$ rm — удаление указанных файлов. Используется для удаления непустых каталогов. В команде можно использовать ключи:

-i — запрашивать разрешение на удаление файла;

-r — рекурсивное удаление файлов и каталогов. Именно этот ключ позволяет удалять непустые каталоги.

-f — ключ подавляет вывод на экран запроса на подтверждение выполнения операции удаления. Приведенная ниже команда может удалить все файлы на диске:

```
$ rm -rf *.*
```

Существует ряд команд для работы с текстовыми файлами:

\$ tail – печатает конец файла, по умолчанию выводится последние 20 строк

\$ file <файл> — определяет тип файла

\$ file /bin / ls — определит тип как executable, т.е. исполняемый

@ sort – сортирует файл в алфавитно- цифровом порядке.

\$ grep <подстрока> <файл> — отыскивает заданную подстроку в файле.

2.4 Классы и права доступа к файлам

У каждого файла в имеется 2 владельца: владелец-пользователь и владелец-группа. Владелец-пользователь – пользователь, создавший файл. Порядок назначения владельца зависит от реализации операционной системы. Различают 3 класса доступа к файлам:

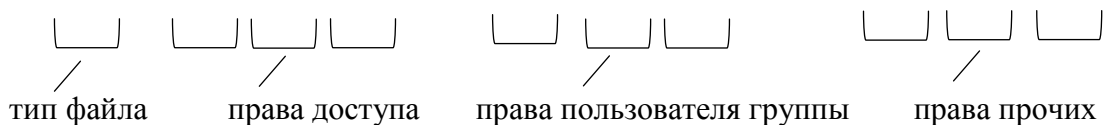
User access – класс доступа владельца – пользователя;

Group access – класс доступа владельца – группы;

Other access – класс доступа прочих пользователей.

Права доступа : право на чтение -r (read) , право на запись -w(write), право на выполнение-x(execute).

Право доступа к файлу можно увидеть, выполнив команду \$ ls с ключом -l. Вывод будет иметь следующий вид:

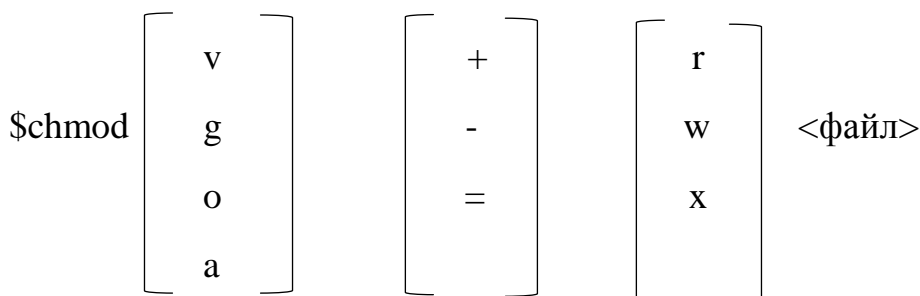


$\boxed{}$ - обычный файл

Владельца – пользователя можно пере назначить командой \$chown

\$chown <владелец> <файл>

Права доступа к файлу можно изменить командой \$chmod



В этой команде можно использовать числовые обозначения прав доступа, при этом наличие права трактуется как двоичная 1, а отсутствие права как двоичный 0.

r w x r _ _ r _ _

$\underbrace{111}$ $\underbrace{100}$ $\underbrace{100}$

7 4 4

\$chmod 744 referat

2.5 Связи между файлами

Поскольку в каталоге хранится только имя файла и указатель на метаданные (номер индексного дескриптора), то один и тот же файл может иметь несколько имен. Для этого необходимо установить между файлами связи. Связи бывают

жесткие и символические. Жесткие связи указывают на один и тот же дескриптор и устанавливаются командой `ln`. Предположим, что файл `pr.f` существует и `Ivanov` - текущий каталог.

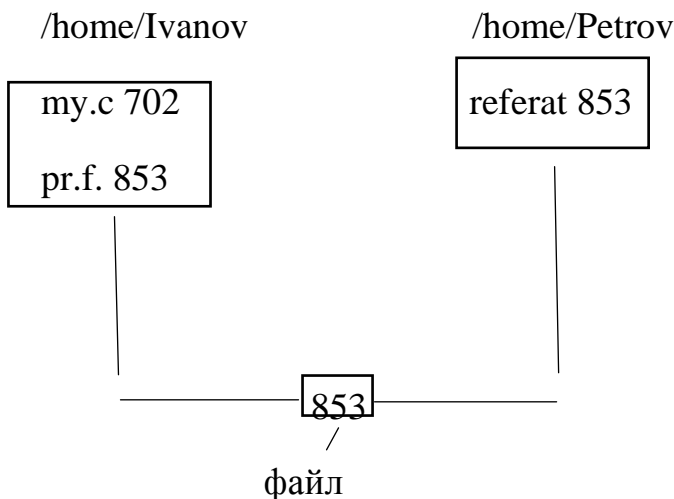
```
$ pwd
```

```
/home/Ivanov
```

```
$ ln pr.f /home/Petrov/referat
```

В результате будет создан файл `/home/Petrov/referat`, обладающий точно таким содержимым и такими же атрибутами, что и файл `/home/Ivanov/pr.f`

Жесткие связи равноправны. При удалении одного из файлов не произойдет удаления самого файла. В то же время изменения, внесенные в один файл, затронут и другой.



В отличие от жесткой связи, символическая связь адресует файл, который, в свою очередь, ссылается на другой файл. В результате, последний файл адресуется символической связью косвенно. Данные файла, являющегося символической связью, содержат только имя целевого файла.

Создается командой `$ ln -s`

2.6 Перенаправление ввода и вывода

Каждая команда использует два открытых потока ввода/вывода:

- стандартный ввод;
- стандартный вывод.

По умолчанию все эти потоки ассоциированы с терминалом. То есть любая команда будет ожидать ввода с клавиатуры терминала, весь вывод команды, включая сообщения об ошибках, будет происходить на экран терминала. Большое число команд используют только стандартные потоки. Для таких команд можно перенаправлять потоки ввода/вывода. Например, можно установить ввод или вывод из файла и даже передать вывод одной программы на ввод другой.

Для перенаправления вывода используется команды:

>

>>

Последняя команда осуществляет перенаправление вывода с дозаписью в файл.

Так в команде

```
$ ls -l /bin
```

содержимое каталога /bin будет выведено по умолчанию на экран.

Команда

```
$ ls -l /bin > f1
```

осуществляет вывод содержимого каталога /bin в файл f1 текущего каталога.

Команда

```
$ pwd >> f1
```

запишет имя текущего каталога в конец файл f1.

Перенаправление стандартного файла вывода часто используется для создания файла с клавиатуры:

```
$ cat > my.txt
```

Эта команда в текущем каталоге создаст файл my.txt, который будет содержать информацию, введенную с клавиатуры. Признаком конца ввода служит комбинация клавиш Ctrl+D.

Перенаправление стандартного ввода < используется реже . Можно использовать одновременно и перенаправление ввода, и перенаправление вывода.

```
$ cat<f1>f2
```

Команда копирует содержимое файла f1 в файл f2.

2.7 Лабораторная работа №1

Цель: Освоение пользовательского интерфейса ОС Linux , приобретение навыков работы в терминальном режиме.

Вопросы для изучения

1. Простейшие команды Linux.
2. Файловая система. Права и классы доступа к файлу. Права доступа к каталогу. Связи между файлами.
3. Стандартный ввод-вывод. Перенаправление стандартного ввода- вывода.
4. Работа с текстовыми файлами.

Задание

1. Выполнить простейшие команды (вывод даты, активных пользователей и т.д.).
2. Создать домашний каталог. В этом каталоге создать файл с клавиатуры.
3. Вывести содержимое файла и каталога, изменяя ключи соответствующих команд.
4. Изменить права доступа к файлу. Вывести изменившуюся информацию.
5. Создать жесткие и символические связи файла.
6. Выполнить команды копирования и перемещения файла.
7. Выполнить команды удаления пустых и непустых каталогов.
8. Выполнить сортировку текстового файла различными способами.
9. Создать «темный каталог». Показать, что доступ к нему закрыт.

Контрольные вопросы

1. Верно ли, что физический обмен с дисками осуществляется байтами?
2. Верно ли, что файлы располагаются на диске последовательно, один за другим?
3. Какие типы файлов существуют в LINUX?
4. Какая информация хранится в каталоге системы LINUX?
5. Какие виды устройств существуют в LINUX?
6. Как записывается путь к файлу в LINUX и чем отличие от MS DOS?
7. Каким образом файл в LINUX может иметь несколько имен?
8. Когда происходит физическое удаление файла в LINUX?

9. Сколько владельцев имеет файл в ОС LINUX?
10. Какие классы доступа к файлу существуют в ОС LINUX?
11. Какие права доступа к файлу существуют в ОС LINUX?
12. Какие операции позволяют выполнить право на чтение каталога?
13. Верно ли, что для перехода в другой каталог необходимо иметь право на запись в каталог?

3 Управление процессами

3.1 Характеристики процессов

Процесс – некоторая деятельность, связанная с выполнением программы аппаратными средствами компьютера. Процесс требует определенного окружения, прежде всего это такие ресурсы, как оперативная память, внешние устройства, время центрального процессора. Основным отличием процесса от программы является то, что программа – пассивна, а процесс – активен. Другими словами, программа представляет собой файл, который хранится на диске и никаких действий не вызывает. Этот файл может быть текстовым и содержать исходный текст программы на том или ином алгоритмическом языке или двоичным файлом, состоящим из машинных команд. Процесс всегда активен и характеризует некоторую совокупность набора исполняющихся команд и ассоциированных с ним ресурсов. Процесс управляется операционной системой.

Процесс может находиться в одном из следующих состояний:

- порождение, т.е. подготовка условий для исполнения процесса;
- готовность. Означает, что процессу выделены все необходимые ресурсы. Он ждёт, когда ему будет выделен квант времени центрального процессора;
- активность или счёт. Процесс обладает всеми необходимыми ресурсами и непосредственно выполняется процессором;
- ожидание. Это пассивное состояние процесса, процесс заблокирован, он не может выполняться по своим внутренним причинам, он ждет осуществления некоторого

события, например, завершения операции ввода-вывода, получения сообщения от другого процесса, освобождения какого-либо необходимого ему ресурса;

– окончание. Это благополучное или аварийное завершение процесса, при котором освобождаются все ресурсы, выделенные ранее процессу.

В Linux различают несколько видов процессов:

– системные процессы. Такие процессы запускаются при инициализации системы, не имеют соответствующих исполняемых файлов и располагаются в адресном пространстве ядра. Каждый системный процесс Linux имеет уникальное имя, например, процесс `Shed`. Прародителем всех процессов считается процесс `Init`. Он относится к системным процессам и это исключение из правила, так как он запускается с помощью соответствующего файла, расположенного в каталоге `/init/etc`;

– процессы-демоны. Это процессы, которые запускаются обычным образом, путем загрузки в ОП соответствующих исполняемых файлов. При этом демоны выполняются в фоновом режиме, не осуществляют вывод на экран, не ведут диалогов с пользователем. Большую часть времени демоны находятся в состоянии ожидания, когда тот или иной процесс запросит некоторую услугу. К процессам-демонам можно отнести процесс `Init`, процессы, порожденные подсистемами печати, сетевого доступа. Демоны не управляются пользователями;

– процессы-зомби – это процессы, выполнение которых завершилось, но относящиеся к ним структуры ядра по каким-то причинам не освобождены. Такие процессы можно назвать «мертвыми». Администратор должен их уничтожать. Причиной появления процессов-зомби могут быть ситуации, когда родительский процесс завершился раньше дочернего или родительский процесс не смог принять сигнал о завершении дочернего;

– прикладные процессы. К ним относятся все остальные процессы, которые порождаются во время сеанса пользователя.

Процессы имеют ряд характеристик (атрибутов):

– уникальный номер PID. Каждому новому порожденному процессу операционная система присваивает PID, увеличивая предыдущее значение на единицу. Достигнув

максимального значения, PID начинает формироваться вновь с нуля. При завершении процесса его PID освобождается;

– PPID (Parent Process Ident) – идентификатор родительского процесса;

– Nice number – заказанный приоритет процесса. Каждый процесс имеет два приоритета - текущий и заказанный. Значение текущего приоритета зависит от заказанного и может изменяться;

– TTY - терминальная линия, ассоциированная с процессом.

Заказанный приоритет можно изменить командой \$nice. Информацию об активных процессах и их атрибутах можно вывести командой \$ PS PID

Ключ -l этой команды позволяет выводить состояние процесса:

O – активное;

S – ожидание;

R – готовность;

I – порождение.

Z – зомби

Команда \$ PS выводит так же атрибут TTY.

3.2 Лабораторная работа №2

Цель: Освоение пользовательского интерфейса ОС Linux, приобретение навыков работы в терминальном режиме.

Вопросы для изучения

1. Понятие процесса.
2. Управление процессами.
3. Команды для работы с процессами.
4. Работа с текстовыми файлами

Задание

1. Вывести информацию о всех активных процессах.
2. Изменить приоритет процесса.
3. Принудительно завершить процесс.

4. Запустить команду сортировки текстового файла в фоновом режиме. Вывести характеристики соответствующего процесса.

Контрольные вопросы

1. Какие процессы называются процессами-демонами и процессами-зомби?
2. Какие 2 уровня планирования процессов существуют в ОС и какие действия выполняются на каждом из них?
3. Какие программы могут выполняться в фоновом режиме?
4. Какие дисциплины диспетчеризации используются в операционных системах?
5. Какими атрибутами обладает процесс?
6. Как рассчитывается текущий приоритет? Зачем он нужен?
7. Какие существуют команды для работы с текстовыми файлами?

4 Графические оболочки Linux

4.1 Обзор графических оболочек

Графическая оболочка Linux состоит из нескольких частей:

- X Window System — часть графического пользовательского интерфейса, позволяющая использовать видеоадаптер в графическом режиме. Она выводит на экран различную текстовую и графическую информацию, отображает действия клавиатуры и мыши;
- менеджер окон (Window manager) — часть интерфейса, которая может менять размер и положение окон, сворачивать и разворачивать их, а также менять внешний вид;
- элементы управления (toolkit, widget set) — набор стандартных элементов пользовательского интерфейса (кнопки, списки).

Графические оболочки также содержат средства для решения наиболее часто встречающихся задач - редактирование файлов, изменение списка пользователей, просмотр каталога.

Все перечисленные части оболочки не являются частью ОС. Поэтому можно выбрать любую из имеющихся реализаций любой из ее составных частей. Часто одна или несколько графических оболочек входят в состав наиболее распространенных дистрибутивов и могут быть добавлены автоматически при установке.

На данный момент наиболее распространены следующие оболочки: KDE (K Desktop Environment) и GNOME (GNU Network Object Model Environment).

Имеются и другие графические оболочки, однако они не настолько широко известны.

К достоинствам KDE можно отнести следующее:

- прекрасный современный рабочий стол;
- рабочий стол, полностью прозрачный для работы в сети;
- интегрированная система помощи, обеспечивающая удобный и согласованный доступ к системе помощи.
- общий внешний вид и управление во всех приложениях KDE;
- стандартизированные меню и панели инструментов, схемы.
- в KDE поддерживается более 40 языков;
- централизованное единообразное конфигурирование рабочего стола в диалоговом режиме;
- большое количество полезных приложений для KDE.

Gnome в настоящее время является официальным Рабочим Столом фирм IBM, HP, Sun, также используется в государственных учреждениях США, Испании и Китая. Эта оболочка поддерживает 114 языков, проста и легка в использовании.

Помимо этого, Gnome поддерживает дополнительные опции для людей с ограниченными возможностями.

На рисунке 2- 3 представлены виды экрана графической оболочки KDE.

Виды экрана оболочки Gnome представлены на рисунке 4 – 5.

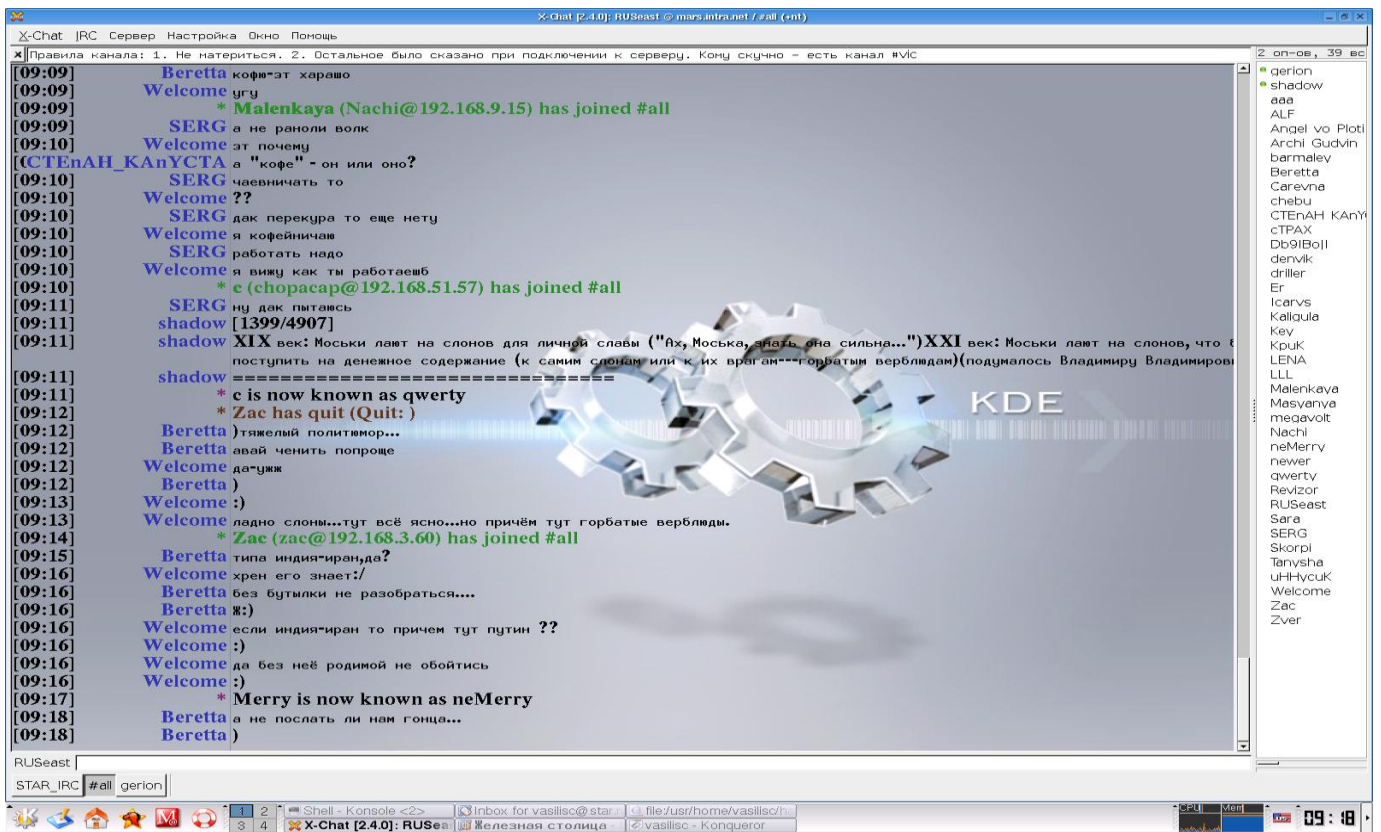


Рисунок 2 - Вид экрана оболочки KDE

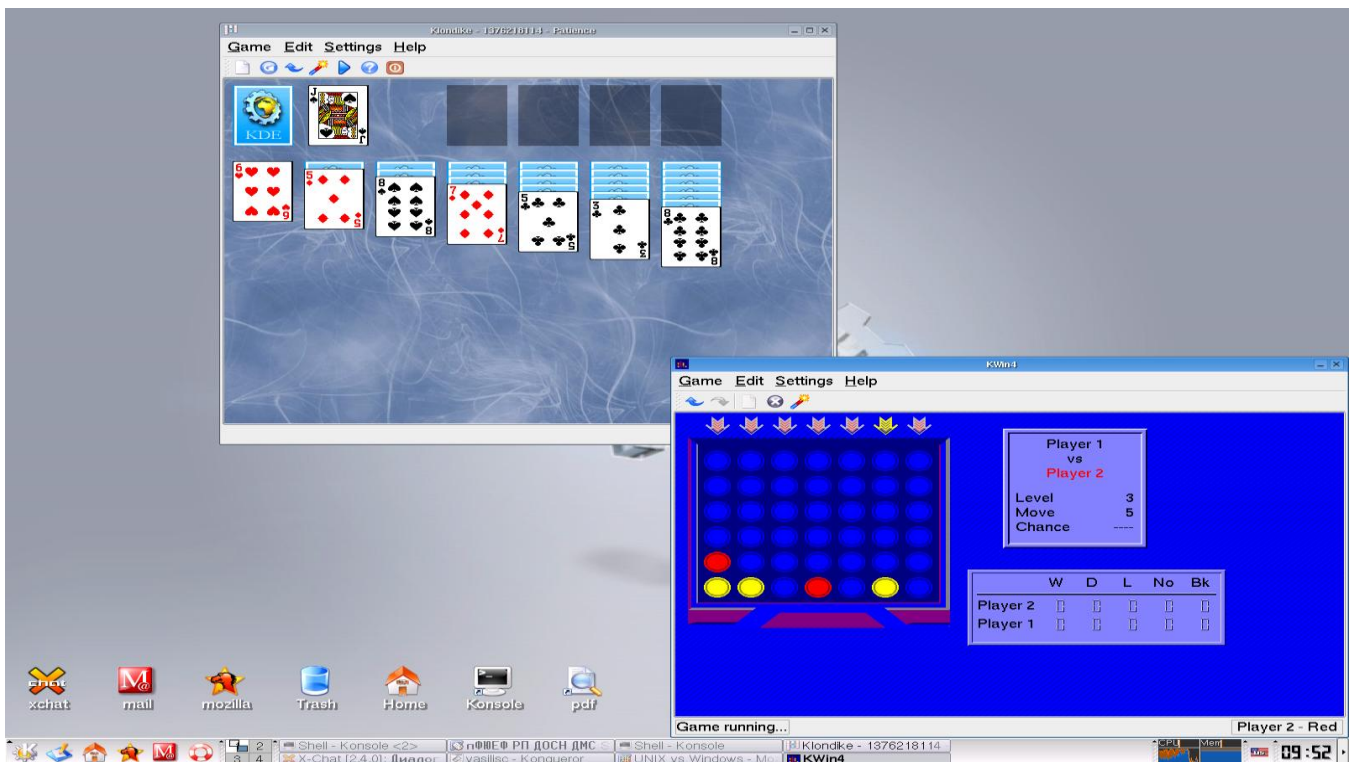


Рисунок 3 - Вид экрана оболочки KDE

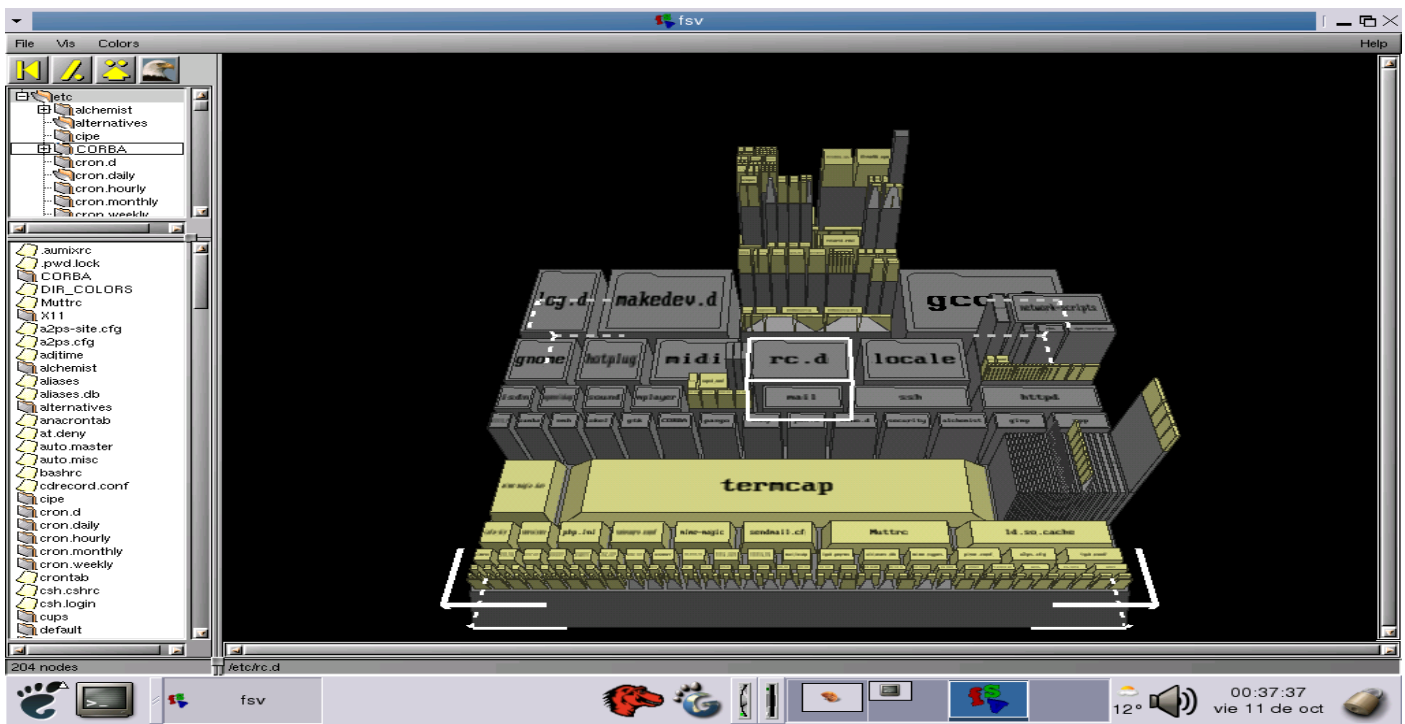


Рисунок 4 - Вид экрана оболочки Gnome

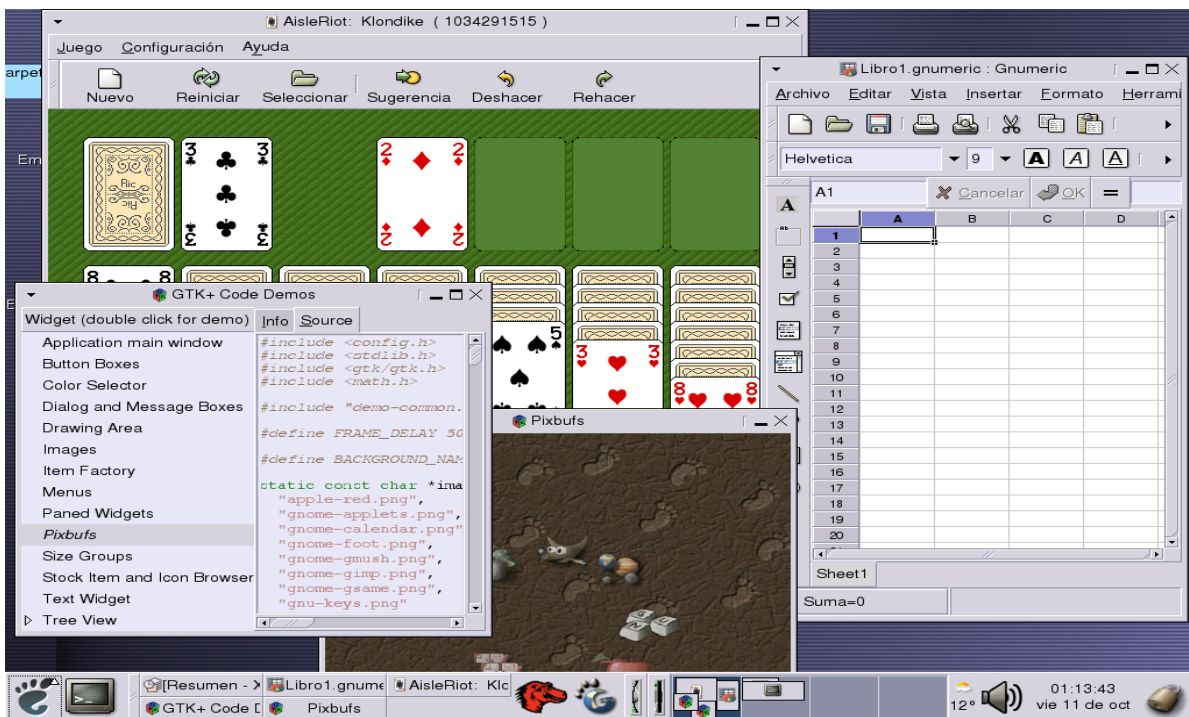


Рисунок 5 - Вид экрана оболочки Gnome

Unity — свободная оболочка для среды GNOME, изготовленная компанией Canonical для Ubuntu. Данная оболочка оптимальна для устройств с маленькими экранами. Оконный менеджер - Metacity, который был заменен на Compiz, в настоящее время используется Mutter. Уже в версии Ubuntu 11.04, оболочка

GNOME была заменена на Unity, но все стандартные приложения были из GNOME. На рисунке 6 представлен вид экрана оболочки Unity.

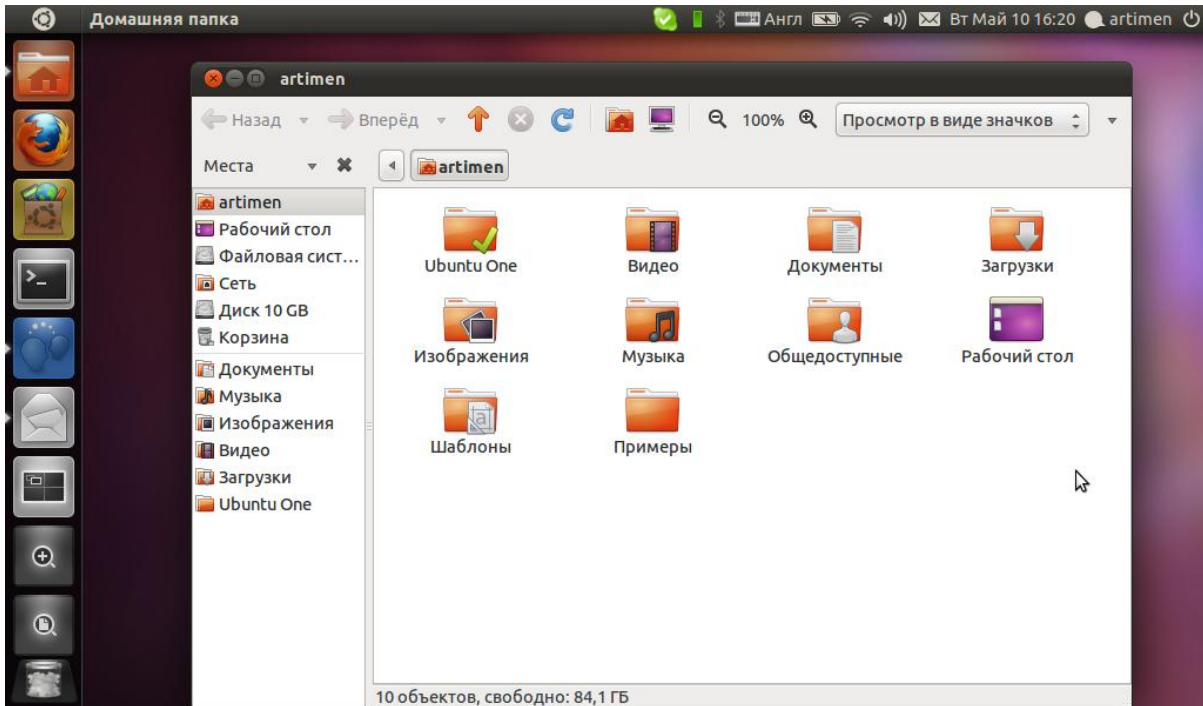


Рисунок 6 - Вид экрана оболочки Unity

Equinox Desktop Environment (EDE) — свободная среда рабочего стола для UNIX-систем, отличается своей компактностью, быстротой, лёгкостью и простотой. Вид рабочего стола этой среды представлен на рисунке 7.

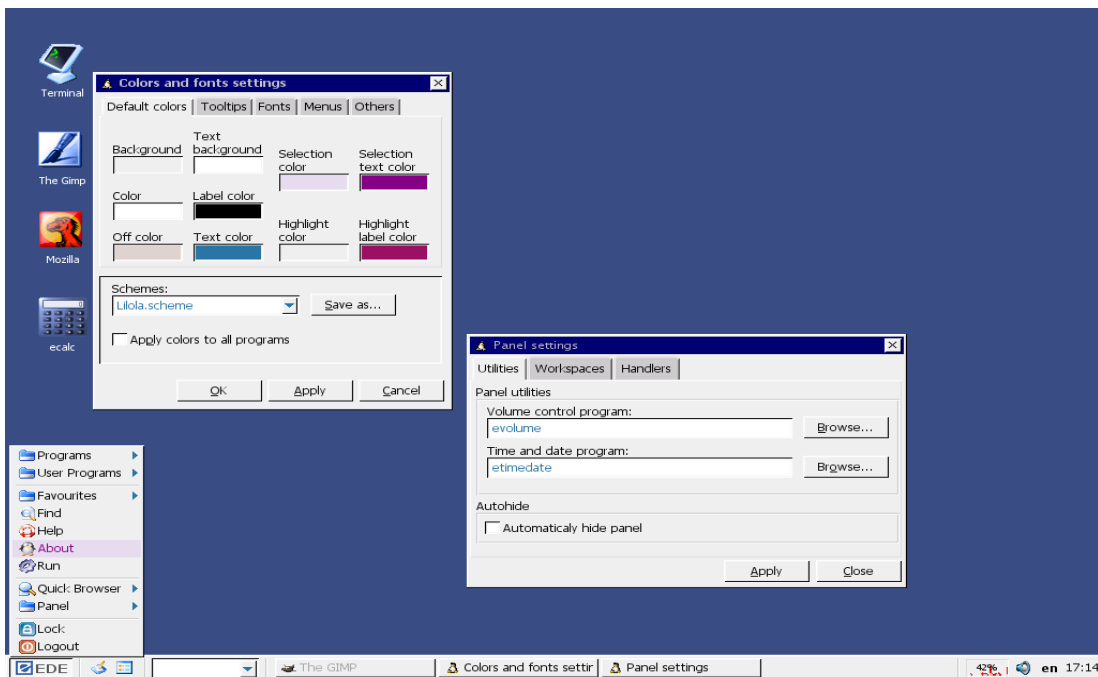


Рисунок 7 - Вид рабочего стола Equinox Desktop Environment (EDE)

OpenWindows — оконная среда для рабочих станций Sun Microsystems.

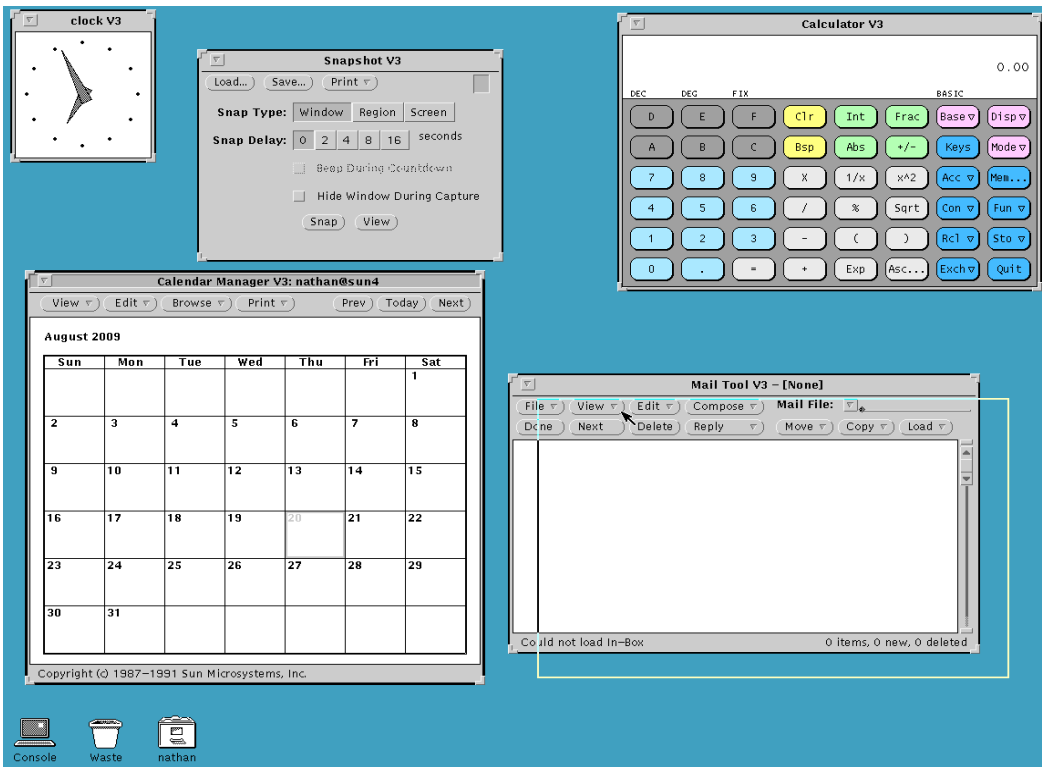


Рисунок 8 - Пример оконной среды OpenWindows

Оконная среда OpenWindows также отвечала спецификации OPEN LOOK. Пример оконной среды приведен на рисунке 8.

XPde (XP Desktop Environment) - свободная среда рабочего стола, напоминающая по виду Windows XP, написана на Kylix. На рисунках 9-10 представлены виды рабочего стола оболочки.

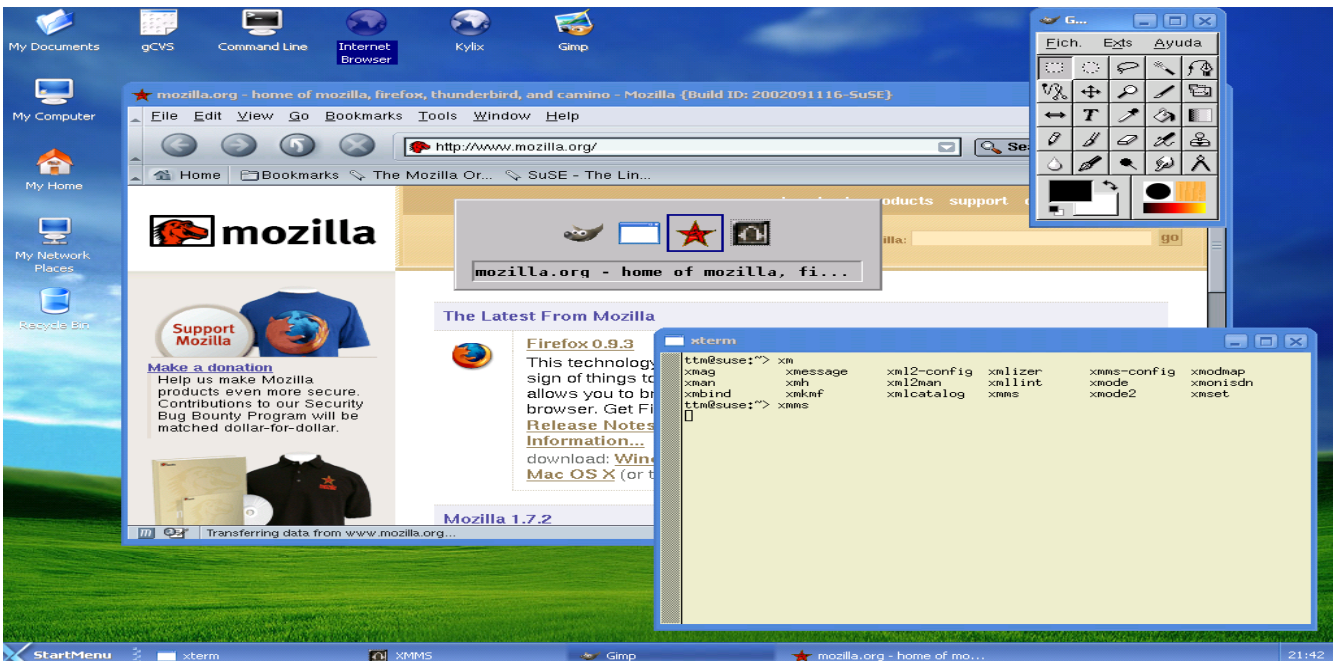


Рисунок 9 - Вид рабочего стола оболочки XPde

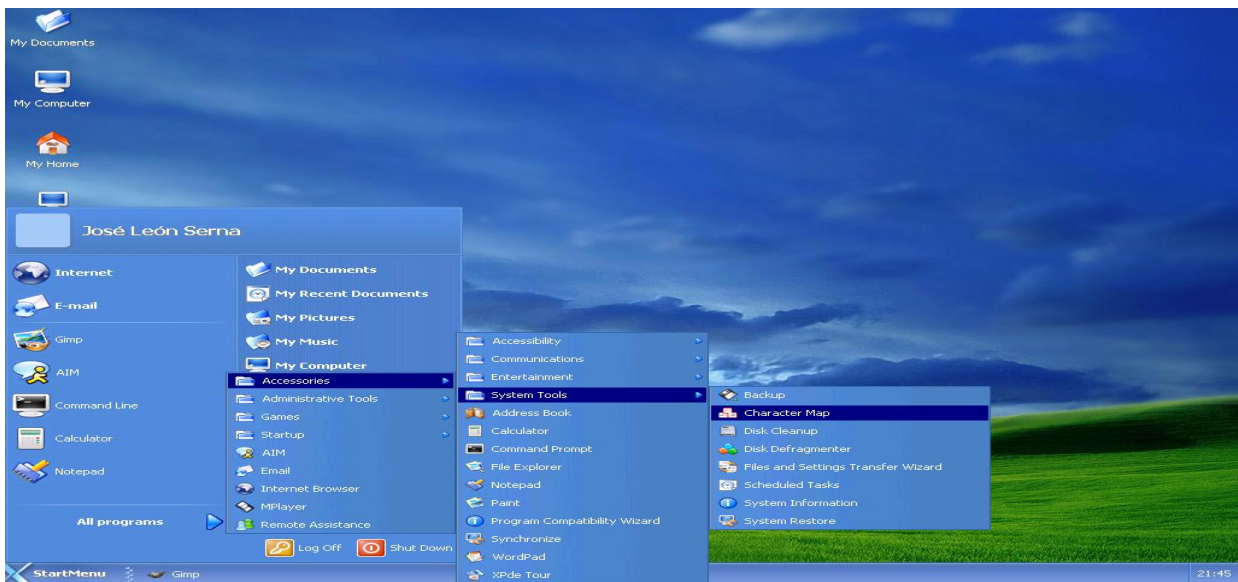


Рисунок 10 - Вид рабочего стола оболочки XPde

По замыслу разработчиков, их оконный менеджер должен облегчить пользователям Windows переход на Linux.

4.2 Графическая оболочка KDE

KDE предназначена для поддержания тех же функциональных возможностей графического интерфейса, какие предоставляют популярные операционные системы MacOS и Windows. Кроме выполнения стандартных функций, KDE обладает рядом характеристик расширения графических возможностей. Для Linux разработано несколько диспетчеров окон, таких, как olwm, fvwm, afterstep, но их возможности никак нельзя сравнить с KDE.

KDE представляет собой интегрированную среду, имеющую базовые средства для решения различных стандартных задач. При помощи KDE можно выполнять действия:

- размещение на рабочем столе ярлыков гибких дисков для работы с ними;
- отображение в графическом виде файловой структуры и перемещение по ней;
- сопоставление приложений с файлами определенных типов, если кликнуть по файлу – сразу запустится установленное для него приложение;

– создание на рабочем столе ярлыков принтеров, просто перетащить файл к ярлыку и он будет распечатан.

KDE содержит набор приложений и утилит для работы с рабочим столом. В стандартном дистрибутиве содержится больше 100 программ — от игр и утилит до пакетов офисных приложений. Также приложения KDE могут взаимодействовать друг с другом для упрощения выполнения всевозможных операций.

Рабочий стол KDE разделен на три основные части – «лицо» рабочего стола, панель и линейку задач. Рабочий стол - основная область среды KDE, фон, на котором отображаются все компоненты. На рабочем столе можно размещать ярлыки часто используемых программ, документов и устройств, что позволяет легко получать доступ к объектам для работы. Помимо области, отображаемой на экране, KDE еще предоставляет виртуальное рабочее пространство для исполнения программ (по умолчанию четыре виртуальных рабочих стола). Виртуальный рабочий стол - альтернативный экран, на который можно переключиться для того, чтобы запустить приложение или выполнить работу. Между рабочими столами программы и окна перемещаются весьма легко. Дополнительные возможности от применения рабочих столов могут быть использованы самыми разными программами, при этом не надо постоянно сворачивать-разворачивать приложения. Можно просто отложить выполняемое приложение в таком виде, как есть, а затем вернуться к нему по завершении выполнения.

В нижней части экрана располагается панель. На панели размещаются кнопки, позволяющие выполнять основные процедуры KDE, ярлыки частых программ. Одним из особо важных элементов на панели является кнопка Application Starter (Запуск Приложений), которая расположена в левой части панели по умолчанию. Это кнопка с литерой "K" над изображением зубчатого колеса. С ее помощью можно открыть меню, в котором представлены все приложения, установленные на данную систему. Кроме того, это же меню может быть использовано для доступа к некоторым другим разделам KDE, таким, как диалоговая справка и Панель Управления (Control Panel).

На панели размещен переключатель виртуальных рабочих столов Пейджер, Панель Задач (Taskbar) и Часы (Clock). Панель задач отображает открытые на

текущем рабочем столе окна. Чтобы получить немедленный доступ к программе, нужно просто щелкнуть в соответствующем месте на панели задач.

Перечисленные ниже способы позволяют запустить программу:

- щелкнуть кнопкой на панели (некоторые программы представлены по умолчанию на панели в виде ярлыков или кнопок);
- щелкнуть на элементе рабочего стола. По умолчанию на рабочем столе размещается только два объекта – корзина и ярлык рабочего каталога.
- выбрать программу из меню запуска приложений - кликнуть на литере "К" и выбрать необходимое приложение для запуска в меню.
- использовать диспетчер файлов – выбрать файл и кликнуть мышью
- в командной строке окна терминала программу можно запустить путем ввода названия программы. Можно также нажатием клавиш <Alt+F2> вызвать окно запуска программ и ввести название.

Существуют полезные программы в помощь пользователю. В первую очередь, это программа эмуляции терминала *konsole*, позволяющая открывать окна и получать доступ к стандартной командной строке. На панели имеется кнопка с изображением маленького монитора и ракушки.

Справку в диалоговом режиме можно получить, если щелкнуть на кнопке панели с изображением спасательного круга. Справка включает в себя разные темы, как, например, программа-гид для начинающих пользователей и система контекстного поиска для используемых в KDE приложений.

Просмотреть файловую систему или получить доступ к сети Интернет можно при помощи окна диспетчера файлов. Чтобы отобразить в диспетчере все содержимое рабочего каталога, нужно кликнуть на папке панели с изображением домика.

Клик по кнопке ► удаляет панель с экрана. Эта кнопка остается при этом на экране, так что можно вернуть панель обратно. Это свойство действует только на открытый в данный момент рабочий стол; другие рабочие столы сохраняют вид мини или главной панели.

Список задач - кнопка, расположенная справа от меню приложений (обозначена пиктограммой монитора), несет меню, содержащее все активные на

данный момент окна, отсортированные по имени. Список задач позволяет легко и быстро найти нужное окно и уменьшить засоренность стола при одновременном открытии нескольких окон.

Центр управления (Control Center) (кнопка с изображением гаечного ключа) составляет основу всей системы настроек KDE. В ее состав входят панели для различных компонентов рабочей среды и даже некоторых приложений KDE.

В центре управления используется деление на группы, щелкнув на знаке "плюс" в углу группы, можно увидеть список входящих в группу компонентов. Щелкнув на знаке "минус" в том же углу группы, этот список можно свернуть. Доступ к любому диалогу с раскрытым деревом меню можно получить при помощи Preferences (Предпочтения) и из меню запуска программ Start Application.

Преобладающая часть диалоговых окон владеет кнопками для вызова справки. Самый элементарный случай - это контекстная справка. Чтобы получить контекстную справку, требуется щелкнуть мышью на знаке вопроса, находящемся на рамке окна. После этого курсор мыши обновит свой вид на стрелку с большим знаком вопроса. Если щелкнуть на том элементе диалогового окна, где появились сложности, появится прямоугольник желтого цвета с текстом справки. Для того чтобы получить более детальную справку, необходимо воспользоваться опцией Help (Справка) на левой панели. В случае появления проблем с поиском необходимого диалогового окна, на этой же панели достаточно выбрать опцию Search (Поиск). После этого вводим ключевое слово, по которому и будет осуществляться поиск.

KDE обеспечивает обширные возможности по преобразованию внешнего вида окон и рабочей области, в том числе и отображение фона, ярлыков, шрифтов и тому подобное. Не составляет ни малейшего труда и управление работой отдельных компонентов, вроде того же рабочего стола или окна. К примеру, можно управлять реакцией элемента на щелчок мышью, процессом загрузки и отображения выбранных окон, также выбирать хранитель экрана. Группа Control Center может предоставить все эти и многие другие возможности.

Для того чтобы настроить параметры работы рабочего стола и окон, нужно выбрать опцию необходимого диалога настройки под названием Desktop на дереве центра управления.

Диалоговое окно выбора цвета Colors (Цвета) назначено для изменения применяемой цветовой схемы для окон KDE и графических программ.

Цветовая схема охватывает 18 пунктов выбора цвета для различных элементов окна программы и установки контрастов. В области первоначального просмотра воспроизводятся все элементы окна, которые реагируют на изменение цветовой схемы. Если пользователь поменяет параметры или установки, то в области просмотра воспроизводятся записанные изменения. Также допускается выбрать уже готовую цветовую схему из списка Color Scheme (Схема Цветов).

Для того чтобы изменить любую конкретную установку, необходимо выбрать соответствующий элемент из выпадающего меню области цветов Widget Color (Декорация) или щелкнуть в нужной части окна первоначального просмотра. После того, как мы выбрали элемент, можно изменить его цвет. Для изменения цвета достаточно будет щелкнуть на кнопке и выбрать любой цвет из появившегося диалогового окна выбора цвета.

Контраст изменяется при помощи позиционирования специального рычажка контраста. Он находится в диапазоне от Low, что означает Низкий и до High – Высокий. Эти установки используются при отображении трехмерных рамок вокруг элементов интерфейса приложений KDE.

Для того чтобы подтвердить выбор, необходимо щелкнуть на кнопке Apply. Для многократного изменения цветowych установок, выгодно будет внести изменения в список цветowych схем. Для этого делаем следующее: щелкаем на кнопку SaveScheme и задаем название для своей схемы. Чтобы удалить схему из списка, выделяем ее и щелкаем на кнопку Remove.

Для того чтобы изменить цвет фона или фоновый узор рабочего стола, необходимо на дереве опций центра управления последовательно выбрать Control Center, затем Appearance&Themes и потом Background. После этих действий должно появиться диалоговое окно, которое имеет три основные области. Это список

виртуальных рабочих столов, окно предварительного просмотра и окно настройки параметров.

Собственные настройки фона находятся на виртуальном столе KDE. В таких настройках можно выбрать фон с одноцветной либо двухцветной схемой, а также фоновый узор. Задать способ отображения фонового узора возможно при условии, что он используется. При этом можно выбрать несколько узоров и автоматически переключаться между ними. Возможны и более усовершенствованные опции, которые позволяют сочетать цвета и узоры, а также поддерживать активные настройки фона.

После внесения изменений в установки, они будут отображаться в окне первоначального просмотра.

Для создания виртуальных рабочих столов, в диалоговом окне Control Center нажимаем Desktop, а затем MultipleDesktops, эти действия проделываем для того, чтобы произвести настройку параметров виртуальных рабочих столов в KDE. Число доступных виртуальных рабочих машин показывает указатель Number of Desktops. Это число изменяется в диапазоне от одного до шестнадцати. Здесь также можно задать название для рабочего стола. Это название будет отображаться в списке окон (WindowList), либо использовано в настройках панели.

Appearance&Themes - Screensaver позволяет выбрать хранитель экрана и выполнить настройку его параметров. Опции настройки бывают глобальные и индивидуальные. Под глобальными подразумевается, например, опции установки времени запуска хранителя экрана. А под индивидуальными — установки для каждого отдельного хранителя. Диалоговое окно выбора хранителя экрана имеет три основные секции. Это окно предварительного просмотра, список программ — хранителей экрана и опции настройки.

Нужно выбрать название необходимой программы из предложенного списка. Для того чтобы настроить параметры, нужно щелкнуть на кнопку Setup. В появившемся диалоговом окне производим установку необходимых характеристик.

В поле опции Settings вводим величину интервала в минутах — это установка интервала времени. Хранитель экрана запускается через этот интервал времени.

Распределение процессором времени на работу хранителя экрана определяет параметр Priority (приоритет). Именно так в Linux организовывается многозадачность. Для качественного вывода анимации необходимо, чтобы у хранителя был наивысший приоритет, для этого передвигаем рычажок в позицию High. Если нужно обеспечить высокий приоритет других процессов, то передвигаем рычажок в позицию Low.

Кнопка Test проверяет выполненные установки. Для подтверждения сделанного выбора нажмите кнопку ОК или Apply.

С помощью опций Control Center Appearance&Themes - Desktop/Window behavior центра управления, возможно установить поведение диспетчера окон. Такие настройки определяют способ отображения окон в случае их перемещения и изменения размера. Еще возможно управление процессом разворачивания, размещения и выделения окон при работе с диспетчером окон. Опции, которые позволяют выполнить настройку параметров, находятся в верхней части диалогового окна. Эти опции задают режим перемещения окна, изменение его размеров и определяют функциональность команды Maximize (Развернуть). Для того чтобы окно отображалось вместе со всем содержимым или в виде прозрачной рамки, нужно задать режим отображения окна при перемещении или изменения его размера. При выборе режима отображения всего содержимого окна, процесс перемещения или изменения размеров окна будет требовать дополнительного времени для обновления отображаемых на экране элементов.

Установками Resize следует воспользоваться, если используются окна с изменяемыми размерами, возможен выбор режима обновления содержимого окна при каждом изменении его размера. Специальным рычажком пользуемся для выбора частоты обновления. Каждый раз при изменении размеров окна, его содержимое будет обновляться, если сделанный выбор отличен от None. Благодаря этому возможно отслеживать процесс заполнения окна программой и позволяет выбирать оптимальные размеры последнего.

Меню установок размещения окна на экране Placement (Расположение) - это Window behavior/Moving.

Window behavior/Moving — позволяет определить место на экране, где будет отображаться окно.

Smart — минимизируется перекрытие между окнами.

Cascade — первое окно отображается в левом верхнем углу. Следующее окно отображается сдвинутым немного вправо и вниз, так что окна практически полностью перекрываются.

Random — окна располагаются на экране в произвольном порядке.

Метод получения фокуса, что означает выделение отдельных окон или элементов, является индивидуальным методом настроек KDE. Какое именно из открытых окон активно, какие действия следует выполнить при активизации окна, все это определяет этот метод. Рассмотрим более детально:

Click to focus - окно получает фокус или, другими словами, становится активным, при щелчке на нем мышью. Окно автоматически выводится на первый план по отношению к другим окнам. Этот метод используется по умолчанию.

Focus follows mouse - окно становится активным при обращении к нему напрямую. Это возможно сделать с помощью указателя мыши, при этом используя комбинацию клавиш <Alt+Tab>. Окно может подниматься поверх других окон, а может и не подниматься. Перемещение указателя мыши на рабочую область за пределы окна не значит потерю последним фокуса. Если выбрать опцию Auto Raise, окно будет всплывать на экране при перемещении в его область курсора в течение нескольких миллисекунд, число которых устанавливается с помощью рычажка Delay. А если выбрать опцию Click Raise, окно будет подниматься поверх других окон при щелчке мыши в любой части окна. Иначе такая реакция окна будет наблюдаться лишь при щелчке на его заголовке. Метод получения фокуса весьма полезный метод передачи фокуса. Потому что допускает печатать текст в одном окне и одновременно читать содержимое другого окна, который располагается частично поверх указанного.

Focus Under Mouse - окно становится активным при любом перемещении на него указателя мыши, при этом комбинация клавиш <Alt+Tab> может и не помочь.

Focus Strictly Under Mouse - окно становится активным, если только указатель мыши находится внутри окна, в противном случае ни одно из окон не получит фокус.

Открытое окно может содержать следующие элементы: Window menu – Меню управления окном, Maximize – максимально увеличить окно, Minimize – сделать окно невидимым, Move – передвинуть окно мышью, Size – увеличить или уменьшить окно, Shade – свернуть окно до заголовка, To desktop... – перенос окна на другой рабочий стол, Close – закрыть окно, Always on Top — режим окна «поверх всех».

Window menu - в левом верхнем углу каждого окна находится пиктограмма манипулирования окном, кликнув по ней, появится меню, которое содержит команды манипуляции данным окном.

To desktop...- для начала выберите стол, на который вы хотите переместить это окно, само окно потом исчезнет. Для того чтобы увидеть его снова, необходимо выбрать имя на Линейке задач или щелкнуть на соответствующую кнопку рабочего стола на панели KDE. Ниже панели меню находятся пиктограммы инструментов, которые позволяют выполнять другие команды. Панель инструментов возможно передвигать в разные стороны.

4.3 Лабораторная работа №3

Цель работы: изучение работы с основными функциональными частями графической оболочки KDE, получение навыков по настройке KDE и созданию простейших текстовых и графических документов в KWord и Paint.

Порядок выполнения работы

1. Открыть Центр управлений.
2. Изменить фон сначала на моноцветный, а затем установить изображение фона.
3. Установить любой хранитель экрана и режим ожидания, который равен минуте.
4. Сделайте так, чтобы окна передвигались вместе со всем содержимым.
5. Установить звуковое подтверждение нажатием любой клавиши.

6. Изменить ширину линейки панели.
 7. Запустить диспетчер приложений, а затем запустить программу текстового процессора KWord.
 8. В другом рабочем столе открыть программу растрового редактора Paint.
 9. Открыть KWord и набрать произвольный текст, используя при этом два разных стиля.
- Сохранить этот файл в домашнем каталоге пользователя, затем закрыть KWord.
10. Открыть ваш домашний каталог пользователя при помощи Konqueror, создать в нем каталог и скопировать ваш текстовый файл в этот каталог.
 11. Просмотреть содержание домашнего каталога и скопировать с источника памяти файлы.
 12. Получить справку об интересующем вас объекте.
 13. Создать любой рисунок с помощью Paint, используя не менее четырех цветов.
 14. Сохранить файл с рисунком в домашнем каталоге и закрыть Paint.
 15. Скопировать файл с рисунком в тот же каталог, который вы создали.
 16. Изменить атрибуты доступа к ранее созданным файлам.

Контрольные вопросы

1. Перечислите стандартные функции KDE.
2. Назовите компоненты рабочего стола KDE.
3. Перечислите основные функции панели рабочего стола.
4. Как получить справку в диалоговом режиме?
5. Какие функции реализует центр управления KDE?

Список использованных источников

- 1 Курячий, Г.В. Операционная система Linux: курс лекций / Г.В. Курячий, К.А. Маслинский. –М: ДМК Пресс, 2010.-348 с. - ISBN 978-5-94074-591-4.
- 2 Шоттс, У. Командная строка Linux. Полное руководство / У. Шоттс. – Санкт-Петербург: ИД Питер, 2016.-480 с.- ISBN 978-5-496-02303-0.
- 3 Граннеман, С. Unix. Карманный справочник / С. Граннеман. М: И.Д. Вильямс, 2007. – 260 с. - ISBN 5-8459-1118-4 978-5-8459-1118-6
- 4 Робачевский, А.М. Операционная система UNIX: учеб. пособие / А. М. Робачевский, С.А. Немнюгин, О.Л. Стесик. - Л. : BHV-Санкт-Петербург, 2010. – 656 с. - ISBN 978-5-94157-538-1.