

# Лабораторная работа № 1

**Знакомство  
с операционными системами  
семейства \*nix  
на примере ОС ALT Linux Server**

# Рассматриваемые темы

- История развития ОС семейства Unix
- Организация файловой системы \*nix
- Стандартная структура файловой системы
- Пользователи и процессы в системе
- Права доступа к файлам
- Командный интерпретатор и его роль в системе
- Основные команды системы
- Работа с программными компонентами
- Менеджеры пакетов
- Управление работой демонов
- Возможности SSH для удалённой работы

# История развития ОС семейства Unix

- 1964 – AT&T, GE и MТИ начали разработку MULTICS
- 1969 – AT&T (Bell Labs) выходит из проекта MULTICS
- 1969 – Ken Thompson, Dennis Ritchie, Douglas McIlroy, первая версия UNIX для PDP-7
- 1971, ноябрь – версия для PDP-11 (Edition 1)
- 1969-1973 – создание C
- 1973 – Edition 4, с ядром на C
- 1975 – Edition 5, полностью на C
- 1974 – распространение по университетам
- 1978 – BSD UNIX
- 1980 – начало коммерциализации UNIX, появление многочисленных ветвей системы
- 1988 – стандартизация систем, POSIX

# Свободное программное обеспечение

1983 – Richard Stallman, манифест проекта GNU (GNU's Not Unix)

Свободы пользователей программ:

0. Свобода запускать программу в любых целях
1. Свобода изучения работы программы и адаптации её
2. Свобода распространять копии
3. Свобода улучшать программу и публиковать улучшения

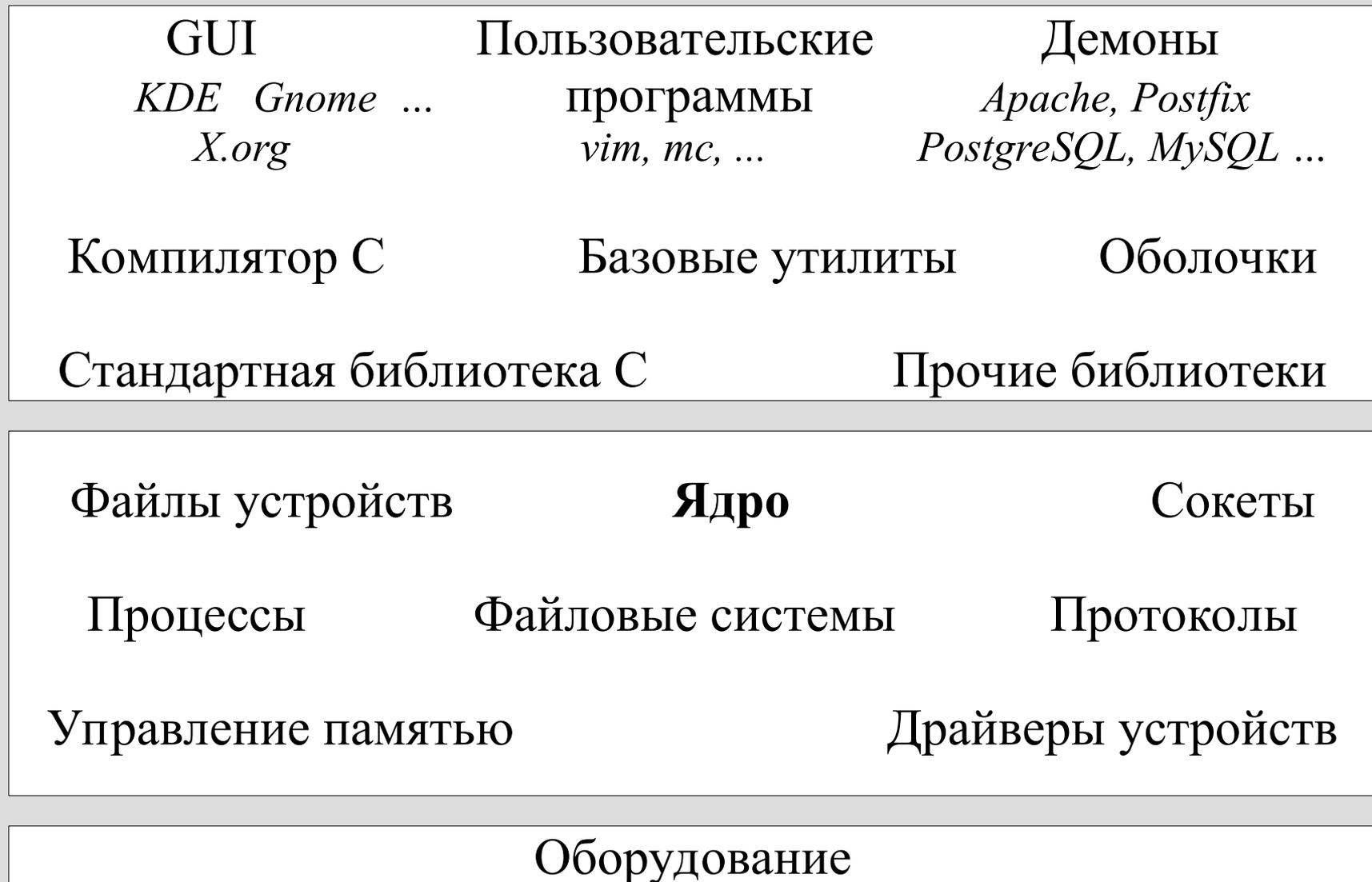
Свободные лицензии:

- GNU General Public License, v. 3 (GNU GPL v.3)
- GNU General Public License, v. 2 (GNU GPL v.2)
- GNU Free Documentation License
- BSD
- MIT
- Artistic (Perl license)
- MPL (Mozilla Public License)
- ...

# Linux-системы

- 1987 – MINIX
- 1991, сентябрь – анонс разработки Linux
- 1991, 5 октября – 0.02
- 1994, 6 марта – 1.0
  
- Состав Linux-системы:  
ядро + утилиты GNU + сторонние приложения
  
- Дистрибутивы:  
Debian, Ubuntu, RedHat, Fedora, Mandriva, Suse,  
Slackware, ALT Linux, ASP Linux

# Архитектура Unix-систем



# Типы файловых систем

## Общего назначения:

- Ext2
- Ext3
- Ext4
- XFS
- JFS
- ReiserFS
- Btrfs
- ISOFS (iso9660)
- UDF
- VFAT
- NTFS

## Псевдо-файловые системы

- procfs
- sysfs
- udevfs

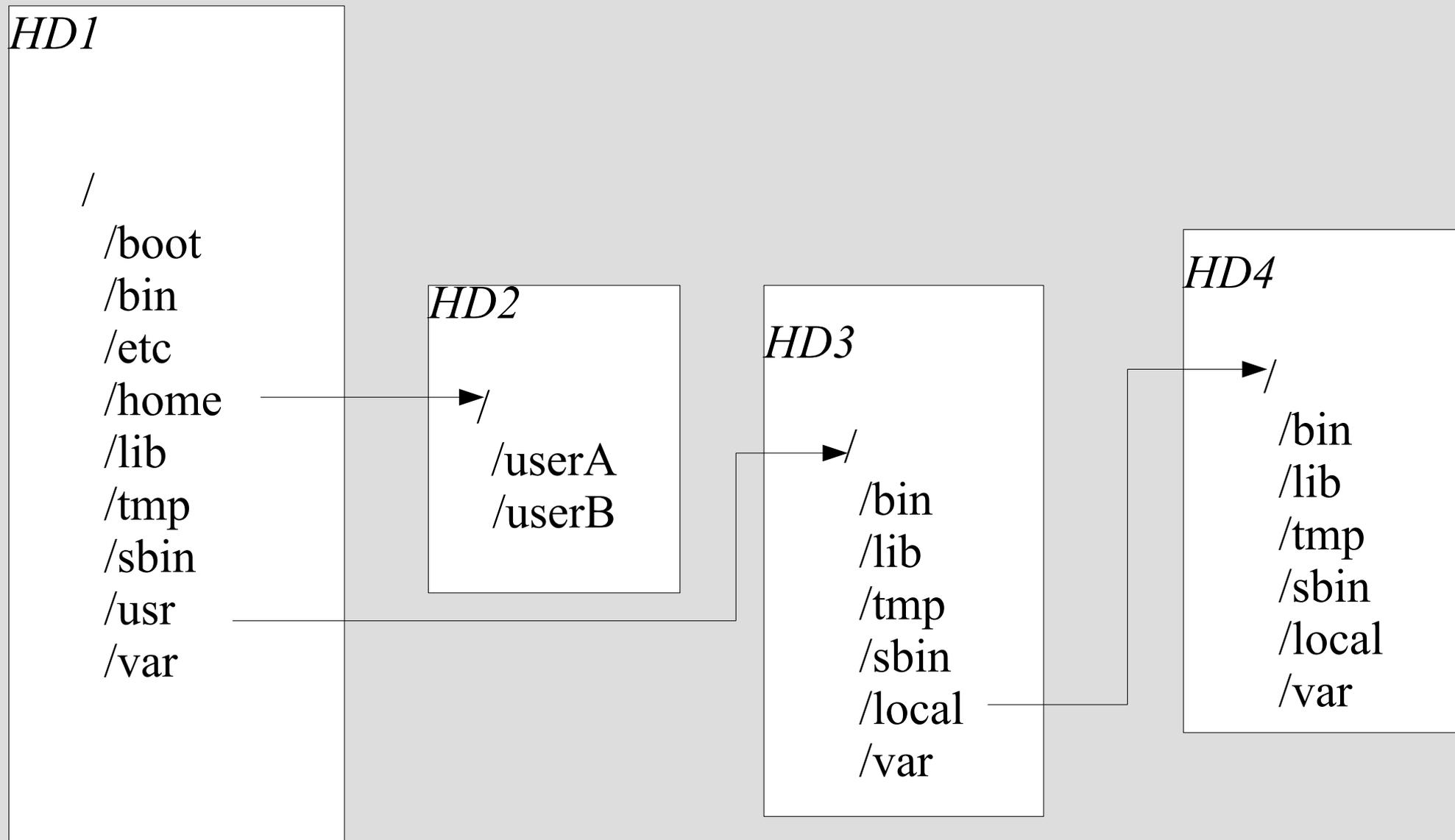
## Сетевые файловые системы

- NFS
- CIFS

## Специализированные файловые системы

- tmpfs
- jffs2
- squashfs

# Стандартная структура файловой системы



# Стандартная структура файловой системы

- структура файловой системы стандартизирована
- есть стандартные каталоги для размещения тех или иных файлов

```
$ ls -l /
```

```
bin
boot      ....
dev       proc
etc       root
home     sbin
lib       srv
lib64    sys
lost+found tmp
media    usr
mnt      var
opt
...
```

```
$ ls -l /usr/
```

```
bin
etc
games
include
lib
lib64
libexec
local
sbin
share
X11R6
```

```
$ ls -l /var
```

```
...
empty
lib
local
lock
Log
...
run
spool
tmp
...
```

# Пользователи и процессы в системе

- Каждый процесс выполняется с правами определённого пользователя
- Система различает пользователей и группы пользователей
- UID, GID – числовые идентификаторы пользователя и группы
- /etc/passwd – список пользователей в системе
- /etc/group – список групп в системе
  
- Пользователи делятся на обычных и псевдо-пользователей
- Пользователь с UID=0 – администратор системы
- Традиционное имя для пользователя с UID=0 – root
  
- Первый процесс в системе – init. Запускается ядром с UID=0, GID=0
- Есть системные вызовы для смены UID и GID
- Изменить свой UID может только процесс с UID=0

# Права доступа к файлам

Каждый файл имеет владельца, группу, и права доступа

Права доступа:

- Read – право на чтение из файла
- Write – право на запись в файл
- eXecute – право на выполнение файла

Каталоги также имеют владельца, группу, и права доступа

- Read – право на чтение списка файлов в каталоге
- Write – право на запись в каталог (создание/удаление файлов)
- eXecute – право на переход в каталог

Запись прав:

- rwx – право на чтение, запись и выполнение
- rw- – право на чтение и запись
- r-- – право на чтение и запись

# Права доступа к файлам

```
$ ls -l ~
итого 8
drwx----- 2 student student 4096 Фев 19 17:30 Documents
-rw-r--r-- 1 student student 0 Фев 20 08:03 file.txt
drwx----- 2 student student 4096 Фев 19 15:59 tmp
$ ls -l /var
итого 72
drwxr-xr-x 2 root root 4096 Апр 19 2007 adm
drwxr-xr-x 4 root root 4096 Фев 15 08:32 cache
drwxr-xr-x 2 root root 4096 Апр 19 2007 db
dr-xr-xr-x 2 root root 4096 Апр 19 2007 empty
drwxr-xr-x 11 root root 4096 Фев 9 15:29 lib
drwxr-xr-x 14 root root 4096 Фев 20 07:32 log
lrwxrwxrwx 1 root root 10 Фев 5 13:22 mail -> spool/mail
drwxr-x--- 2 root nobody 4096 Апр 19 2007 nobody
drwxrwxrwt 2 root root 4096 Апр 19 2007 tmp
drwxr-xr-x 3 root root 4096 Фев 15 09:24 www
drwx----- 2 root root 4096 Апр 19 2007 yp
$ ls -l /bin/su
-rws--x--- 1 root wheel 23712 Окт 18 2006 /bin/su
```

# Командный интерпретатор и его роль в системе

- Обеспечивает пользовательский интерфейс командной строки
- Позволяет пользователю запускать программы
- Предоставляет возможность создания и исполнения файлов с последовательностями команд – скриптов
- Имеет набор встроенных команд
- Часть системных утилит, в т.ч. управления процессами запуска / остановки системы, написаны на языке командного интерпретатора

Существует ряд командных интерпретаторов:  
sh, csh, tsh, bash, zsh ...

Общий формат вызова команды выглядит следующим образом:

```
$ command -f --flag --key=parameter argument1 argument2 ...
```

# Основные команды системы

|  |          |
|--|----------|
| Смена каталога:                        | cd       |
| Просмотр каталога:                     | ls       |
| Создание каталога:                     | mkdir    |
| Удаление каталога:                     | rmdir    |
| Удаление файла:                        | rm       |
| Изменение даты/создание пустого файла: | touch    |
|  |          |
| Просмотр текстового файла:             | cat      |
| Поэкранный просмотр файла:             | less     |
| Просмотр начала файла:                 | head     |
| Просмотр конца файла:                  | tail     |
|  |          |
| Изменение прав доступа к файлу:        | chmod    |
| Изменение владельца файла:             | chown    |
|  |          |
| Редактор:                              | vi / vim |

# Работа с программными компонентами

Основная часть программного обеспечения доступна в виде исходных кодов.

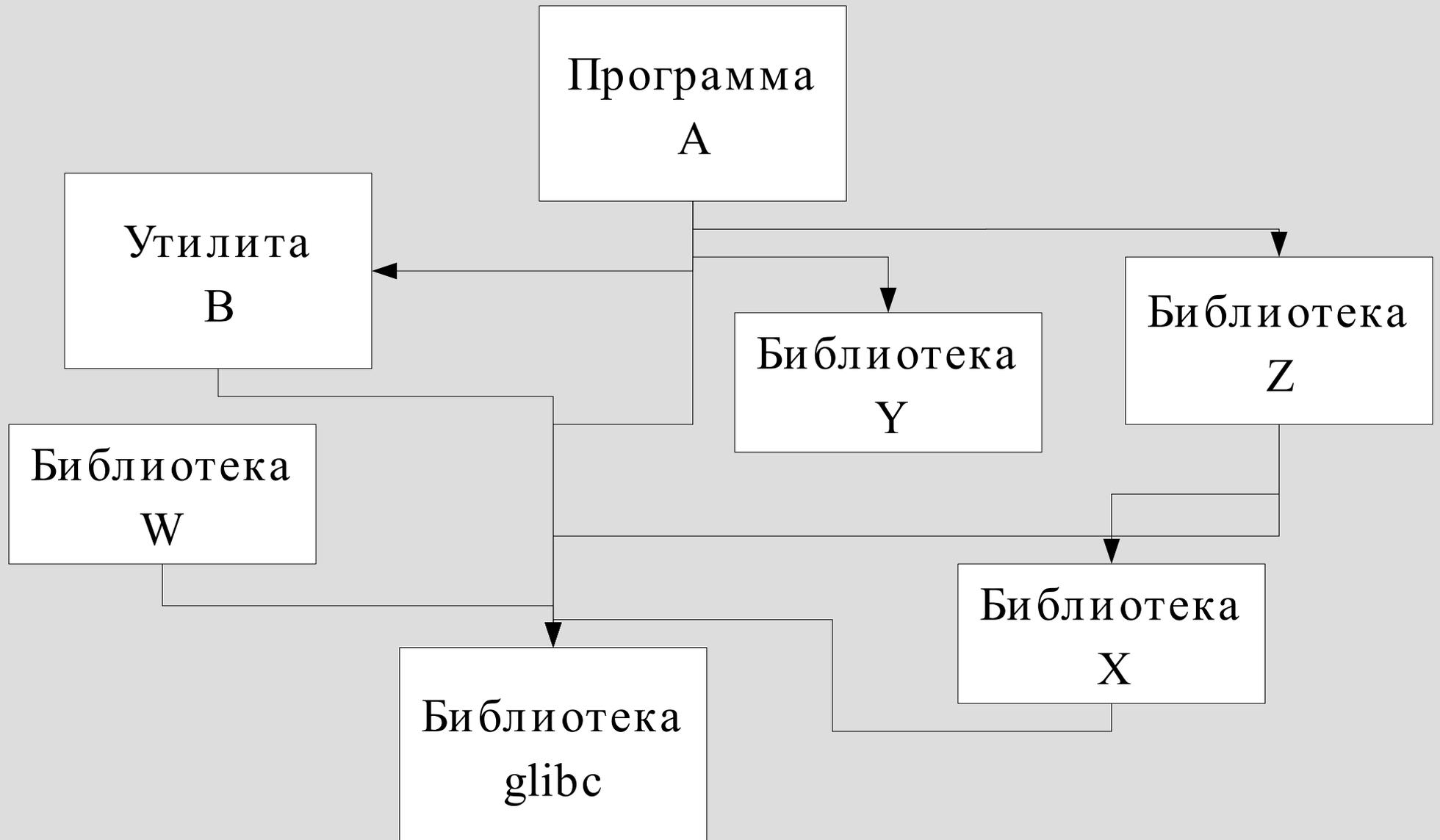
Дистрибутив – набор программных пакетов, настроенных для совместной работы в системе.

Существует большое число различных дистрибутивов:

- общего назначения: Debian, Ubuntu, RedHat, Fedora, Suse, Mandriva, ALT Linux, Astra Linux, ROSA Linux.
- специального назначения: OpenWRT, RIP, DBAN, ...

Как правило, программные пакеты используют другие программные пакеты – в виде разделяемых библиотек, внешних исполняемых файлов, и т.п.

# Работа с программными компонентами



# Менеджеры пакетов

Менеджеры пакетов:

- .deb – dpkg (Debian Package manager) - Debian, Ubuntu
- .rpm – RPM Package Manager – RedHat, Fedora, Suse, ALT Linux, ...

Основная утилита RPM – rpm

RPM отслеживает зависимости в отдельных пакетах .rpm.

Совокупность пакетов – репозиторий.

В репозитории (в идеале) зависимости между пакетами замкнуты.

В ALT Linux для организации репозитория используется APT (Advanced Packaging Tool)

# Менеджеры пакетов

Настройки АРТ: `/etc/apt/sources.list`, `/etc/apt/sources.list.d/*`

Запись о репозитории:

```
#rpm [p11] ftp://ftp.altlinux.org/pub/distributions/ALTlinux/p11/branch x86_64 classic  
#rpm [p11] ftp://ftp.altlinux.org/pub/distributions/ALTlinux/p11/branch noarch classic
```

Работа с АРТ:

```
# apt-get update  
# apt-get dist-upgrade  
# apt-get install <package>  
# apt-get remove <package>  
# apt-cache search <название>
```

# Управление работой ДЕМОНОВ

Демон – традиционное название для неинтерактивных программ.

Управлением порядком загрузки занимаются системы инициализации: `sysvinit`, `systemd`, `Upstart`, `Runit`, `Launchd`, `Initng`, ...

Уровни загрузки системы для `sysvinit`:

0 — уровень остановки системы

1 — однопользовательская система

2 — многопользовательская система без сетевой поддержки

3 — многопользовательская система

4 — предоставлено для конкретных систем

5 — многопользовательская система с поддержкой графики

6 — уровень перезагрузки системы

Переход между уровнями осуществляет `init`

# Возможности SSH для удалённой работы

SSH:

- сервер и клиент для удалённой работы с системами,
- протокол обмена информацией между клиентом и сервером.

Существует несколько реализаций SSH под разные операционные Системы – Linux, Un\*x, Windows, Android, IOS и др.

Соединение между клиентом и сервером – защищённое.

Подключение клиента выполняется с правами указанного при подключении пользователя сервера.

Авторизация клиента может быть:

- парольная
- с использованием алгоритмов открытых ключей.

# Возможности SSH для удалённой работы

SSH позволяет:

- получить удалённую интерактивную консоль на сервере:

```
$ ssh <user>@<server>
```

- выполнить команду на сервере:

```
$ ssh <user>@<server> <command>
```

- передавать файлы между клиентом и сервером

```
$ scp <user>@<server>:/remote/file /local/file
```

```
$ scp /local/file <user>@<server>:/remote/file
```

Создание ключей на клиенте:

```
$ ssh-keygen [-t <type>] [-f /path/to/key]
```

Тип ключа:

- dsa : ключи DSA, устарели, только для старых систем,
- rsa : ключи RSA, рекомендуется длина ключа – 2048 или 4096 бит,
- ed25519 : ключи ED25519, предпочтительно для новых систем

# Работа с репозиториями Git

Клонирование репозитория с сервера через SSH:

- `$ git clone <user>@<server>:<repo> [<local_copy_dir>]`
- Добавление, удаление из рабочей копии:  
`$ git add <path>; git rm [-r] <path>`
- Запоминание (коммит) изменений:  
`$ git commit`
- Создание и переключение ветвей:  
`$ git branch [-r]; git checkout [-b] <branch>`
- Передача изменений на сервер:  
`$ git push [--all]`
- Получение и объединение изменений:  
`$ git fetch; git merge <ветка>`  
`$ git pull`

# Задание на лабораторную работу

- Выполнить удалённую регистрацию в системе.
- Изучить структуру каталогов сервера.
- Посмотреть доступные команды в системе, вызвать справочное руководство по каким-либо из них.
- Создать текстовый файл, используя редактор *vi*.
- Используя команду *su*, получить привилегии суперпользователя системы.
- Изменить пароли пользователя и суперпользователя системы.
- Создать новую учётную запись пользователя.
- Зарегистрироваться в системе под созданным в п. 7 пользователем, убедиться в возможности использования им команды *su*.  
Удалить учётную запись пользователя *student*.
- Получить список пакетов, установленных в системе.  
Настроить список репозиториев пакетов для системы *apt-get*.
- Провести обновление системы до текущего состояния репозитория.
- Установить веб-сервер *lighttpd*, запустить сервер. Проверить работу веб-сервера.
- Настроить его автоматический запуск при загрузке системы.
- Перезагрузить систему.
- Убедиться, что веб-сервер *lighttpd* автоматически запустился после перезагрузки системы.
- Настроить доступ по открытому ключу к тестовому серверу SSH.
- Разместить файл с полученной из репозитория цитатой на веб-сервере.

# Формат данных к лабораторной работе

Виртуальный сервер: lab-99.edu.cbias.ru

VEID: 230199

Login: student  
Password: password

Доступ к виртуальному серверу по SSH:

Имя сервера: ssh.edu.cbias.ru

Номер порта: 22199

Доступ к серверу по HTTP: <http://lab-99.edu.cbias.ru>

---

Установленная в виртуальном сервере система:

ALT Linux Server 11.0 x86\_64 (branch p11)

Список репозиториев:

rpm [p11] ftp://ftp-distr/ALTLinux/p11/branch x86\_64 classic

rpm [p11] ftp://ftp-distr/ALTLinux/p11/branch noarch classic

rpm http://ftp.ossrg.ru/APT/APT x86\_64 ALT\_p11 OSSG\_p11

rpm http://ftp.ossrg.ru/APT/APT noarch ALT\_p11 OSSG\_p11

---

Описания лабораторных работ и прочие материалы:

<http://edu.cbias.ru>

Вопросы, комментарии, предложения: [edu-2025@cbias.ru](mailto:edu-2025@cbias.ru)