

Современные технологии разработки программного обеспечения

История UNIX систем

UNIX — семейство переносимых, многозадачных и многопользовательских операционных систем.

Идеи, заложенные в основу UNIX, оказали огромное влияние на развитие компьютерных операционных систем. В настоящее время UNIX-системы признаны одними из самых исторически важных ОС.

Первая система UNIX была разработана в подразделении Bell Labs компании AT&T. С тех пор было создано большое количество различных UNIX-систем. Юридически право называться «UNIX» имеют лишь те операционные системы, которые прошли сертификацию на соответствие стандарту Single UNIX Specification. Остальные же, хотя и используют сходные концепции и технологии, называются UNIX-подобными операционными системами.

Особенности

Основное отличие UNIX-подобных систем от других операционных систем заключается в том, что это изначально многопользовательские многозадачные системы. То есть в один и тот же момент времени сразу множество людей может выполнять множество вычислительных задач (процессов). Даже популярную во всём мире систему Microsoft Windows нельзя назвать полноценной многопользовательской системой, так как кроме как на некоторых серверных версиях, в один и тот же момент за одним компьютером с Windows может работать только один человек. В Unix может работать сразу много людей, при этом каждый из них может выполнять множество различных вычислительных процессов, которые будут использовать ресурсы именно этого компьютера.

Вторая колоссальная заслуга Unix в её мультиплатформенности. Ядро системы написано таким образом, что его легко можно приспособить практически под любой микропроцессор, а не только под популярное семейство i-386 (i-686).

UNIX имеет и другие характерные особенности:

- использование простых текстовых файлов для настройки и управления системой;
- широкое применение утилит, запускаемых из командной строки;
- взаимодействие с пользователем посредством виртуального устройства — терминала;
- представление физических и виртуальных устройств и некоторых средств межпроцессового взаимодействия в виде файлов;
- использование конвейеров из нескольких программ, каждая из которых выполняет одну задачу.

Применение

В настоящее время UNIX-системы распространены в основном среди серверов, а также как встроенные системы для различного оборудования. Среди ОС для рабочих станций и домашнего применения UNIX и UNIX-подобные ОС занимают после Microsoft Windows второе (OS X), третье (GNU/Linux) и многие последующие места.

Первые UNIX-системы

История операционной системы UNIX началась в 1969 году в одном из подразделений AT&T Bell Laboratories, когда на «малоиспользуемой» машине DEC PDP-7 Кен Томпсон, Деннис Ричи и другие, прежде занимавшиеся созданием ОС Multics, начали работу над операционной системой, названной ими первоначально Unics (UNiplexed Information and Computing System). В течение первых 10 лет развитие UNIX происходило, в основном, в Bell Labs. Соответствующие начальные версии назывались «Version n» (Vn) и предназначались для ЭВМ DEC PDP-11 (16-битовая) и VAX (32-битовая).

Версии Vn разрабатывались группой Computer Research Group (CRG) в Bell Labs. Поддержкой занималась другая группа, Unix System Group (USG). Разработкой также занималась группа Programmer's WorkBench (PWB), привнесшая систему управления исходным кодом sccs, именованные каналы и ряд других идей. В 1983 году эти группы были объединены в одну, Unix System Development Lab.

Основные характеристики

ОС UNIX имеет следующие основные характеристики:

- переносимость;

- *вытесняющая многозадачность* на основе процессов, работающих в изолированных адресных пространствах в виртуальной памяти;
- поддержка одновременной работы многих пользователей;
- поддержка асинхронных процессов;
- иерархическая файловая система;
- поддержка независимых от устройств операций ввода-вывода (через специальные файлы устройств);
- стандартный интерфейс для программ (программные каналы, IPC) и пользователей (командный интерпретатор, не входящий в ядро ОС);
- встроенные средства учета использования системы.

Архитектура ОС UNIX

Архитектура ОС UNIX — многоуровневая. На нижнем уровне, непосредственно над оборудованием, работает ядро операционной системы. Функции ядра доступны через интерфейс системных вызовов, образующих второй уровень. На следующем уровне работают командные интерпретаторы, команды и утилиты системного администрирования, коммуникационные драйверы и протоколы, - все то, что обычно относят к системному программному обеспечению. Наконец, внешний уровень образуют прикладные программы пользователя, сетевые и другие коммуникационные службы, СУБД и утилиты.

Основные функции ядра

Основные функции ядра UNIX (которое может быть *монолитным* или *модульным*) включают:

- планирование и переключение процессов;
- управление памятью;
- обработку прерываний;
- низкоуровневую поддержку устройств (через драйверы);
- управление дисками и буферизация данных;
- синхронизацию процессов и обеспечение средств межпроцессного взаимодействия (IPC).

Системные вызовы

Системные вызовы обеспечивают:

- сопоставление действий пользователя с запросами драйверов устройств;
- создание и прекращение процессов;
- реализацию операций ввода-вывода;
- доступ к файлам и дискам;
- поддержку функций терминала.

Системные вызовы преобразуют процесс, работающий в режиме пользователя, в защищенный процесс, работающий *в режиме ядра*. Это позволяет процессу вызывать защищенные процедуры ядра для выполнения системных функций.

Системные вызовы обеспечивают программный интерфейс для доступа к процедурам ядра. Они обеспечивают управление системными ресурсами, такими как память, пространство на дисках и периферийные устройства. Системные вызовы оформлены в виде библиотеки времени выполнения. Многие системные вызовы доступны через командный интерпретатор.

Пользовательские процессы и процессы ядра

Пользовательские процессы образуют следующие два уровня и:

- защищены от других пользовательских процессов;
- не имеют доступа к процедурам ядра, кроме как через системные вызовы;
- не могут непосредственно обращаться к пространству памяти ядра.

Пространство (памяти) ядра — это область памяти, в которой процессы ядра (процессы, работающие в контексте ядра) реализуют службы ядра. Любой процесс, выполняющийся в пространстве ядра, считается работающим в режиме ядра. Пространство ядра - привилегированная область; пользователь получает к ней доступ только через интерфейс системных вызовов. Пользовательский процесс не имеет прямого доступа ко всем инструкциям и физическим устройствам, — их имеет процесс ядра. Процесс ядра также может менять карту памяти, что необходимо для *переключения процессов* (смены контекста).

Пользовательский процесс работает в режиме ядра, когда начинает выполнять код ядра через системный вызов.

Обмен данными между пространством ядра и пользовательским пространством

Поскольку пользовательские процессы и ядро не имеют общего адресного пространства памяти, необходим механизм передачи данных между ними. При выполнении системного вызова, аргументы вызова и соответствующий идентификатор процедуры ядра передаются из пользовательского пространства в пространство ядра. Идентификатор процедуры ядра передается либо через аппаратный регистр процессора, либо через стек. Аргументы системного вызова передаются через пользовательскую область вызывающего процесса.

Пользовательская область процесса содержит информацию о процессе, необходимую ядру:

- корневой и текущий каталоги, аргументы текущего системного вызова, размеры сегмента текста, данных и стека для процесса;
- указатель на запись в таблице процессов, содержащую информацию для планировщика, например, приоритет;
- таблицу дескрипторов файлов пользовательского процесса с информацией об открытых файлах;
- стек ядра для процесса (пустой, если процесс работает в режиме пользователя).

Пользовательский процесс не может обращаться к пространству ядра, но ядро может обращаться к пространству процесса.

Системное программное обеспечение

ОС UNIX обеспечивает ряд стандартных системных программ для решения задач администрирования, переконфигурирования и поддержки файловой системы, в частности:

- для настройки параметров конфигурации системы;
- для переконфигурирования ядра (если она необходима) и добавления новых драйверов устройств;
- для создания и удаления учетных записей пользователей;
- создания и подключения физических файловых систем;
- установки параметров контроля доступа к файлам.

Для решения этих задач системное ПО (работающее в пользовательском режиме) часто использует *системные вызовы*.

Операционная система FreeBSD

FreeBSD – свободно распространяемая, Unix-подобная операционная система с открытым исходным кодом. Является потомком *AT&T Unix*, ответвление BSD, образовавшееся в Калифорнийском университете в Беркли. Работает на PC-совместимых системах x86, в том числе Microsoft Xbox, а также на DEC Alpha, Sun UltraSPARC, IA-64, AMD64, PowerPC, NEC PC-98, ARM.

Краткая история операционной системы FreeBSD

Операционная система Unix, зародилась в 1969 году, в научно-исследовательском подразделении, компании AT&T, Bell Labs. В то время, монополия на компьютерную индустрию принадлежала государству, и компания AT&T не могла продавать свое программное обеспечение. В результате, в 1976, AT&T решила бесплатно предоставить свое программное обеспечение и его исходные коды, университетской среде. Одни в этом случае получили возможность сэкономить значительные средства, и доступ к технологии, вторые, бесплатную площадку для экспериментов и целое поколение специалистов, выросших на их оборудовании.

В конце 70-х годов, в компании AT&T, была создана рабочая группа поддержки Unix (Unix Support Group, USG), в последствии преобразованная в систему лабораторий Unix (Unix System Laboratories, USL). В задачи USL, входила раскрутка операционной системы Unix, в качестве коммерческого продукта, что с успехом и делалось, система продавалась предприятиям за большие деньги. Разработка системы продолжалась и в Bell Labs и в USG.

Группа по исследованию компьютерных систем (Computer Systems Research Group, CSRG) Калифорнийского университета, участвовавшая в разработке и усовершенствовании системы Unix, стала центральным хранилищем исходных кодов и их улучшений. Группа собирала изменения, оценивала, упаковывала и передавала остальным обладателям действительной лицензии AT&T Unix. Кроме того, CSRG, работала с подразделением министерства обороны США, управлением перспективных исследовательских проектов (Advanced Research Projects Agency, ARPA), с целью реализации в Unix, различных функциональных возможностей, например, стек протоколов TCP/IP. Итоговая коллекция программного обеспечения Калифорнийского университета, получила название BSD (Berkley Software Distribution). Первый выпуск BSD был создан в 1977 году. В конце концов финансирование CSRG прекратилось, после чего, университетом в Беркли, было принято решение, открыть исходный код BSD для широкой публики.

Первым дистрибутивом операционной системы FreeBSD, стала FreeBSD версии 1.0, выпущенная в 1993 году. За основу была взята лента 4.3BSD-Lite ("Net/2"), университета в Беркли,

с многочисленными добавлениями из проекта 386BSD, и уже в мае 1994 года, была выпущена FreeBSD 1.1 RELEASE.

Другие BSD системы

4.4BSD-Lite породила не только операционную систему FreeBSD, но и массу других проектов, из которых можно выделить следующие:

- **NetBSD** — во многом подобна FreeBSD, а их команды делят между собой разработчиков и сам программный код. Основное назначение NetBSD, предоставить надежную и безопасную операционную систему, которая может быть перенесена практически на любую аппаратную платформу с минимальными усилиями. Так NetBSD работает на VAX, PocketPC, высокопроизводительных серверах SPARC и Alpha.

- **OpenBSD** — ответвилась от проекта NetBSD в 1996 году. Основным отличием от других BSD систем, изначальная ориентированность на повышенную безопасность. Одним из дочерних проектов OpenBSD, является пакет OpenSSH, использующийся практически всеми операционными системами и производителями аппаратного обеспечения.

- **MacOS X** — Стабильная операционная система с дружелюбным интерфейсом, использующаяся на компьютерах компании Apple, на процессорах PowerPC и Intel, так-же содержит в себе значительные участки кода BSD.

Возникновение GNU

GNU -свободная Unix-подобная операционная система, разрабатываемая

Акроним GNU был выбран из-за того что GNU является Unix-подобной операционной системой, но отличается от Unix тем, что является свободным и не содержит его кода.

История возникновения GNU

К 1990 году в рамках проекта GNU, основанного Ричардом Столлманом, были разработаны и постоянно развивались свободные программы, составляющие основной инструментарий для разработки программ на языке Си: текстовый редактор Emacs, компилятор языка Си gcc, отладчик программ gdb, командная оболочка bash, библиотека важнейших функций для программ на Си libc. Все эти программы были написаны для операционных систем, похожих на UNIX. Поэтому в них использовались стандартные для UNIX системные вызовы — POSIX. При помощи системных вызовов программы получают доступ к оперативной памяти, файловой системе, устройствам ввода и вывода. Благодаря тому, что системные вызовы выглядели более-менее стандартно во всех реализациях UNIX, программы GNU могли работать (с минимальными изменениями или вообще без изменений) в любой UNIX-подобной операционной системе.

С помощью имевшихся инструментов GNU можно было бы писать программы на Си, пользуясь только свободными программными продуктами, однако свободного UNIX-совместимого ядра, на основе которого могли бы работать все эти инструменты, не существовало. В такой ситуации разработчики GNU вынуждены были использовать одну из патентованных реализаций UNIX, то есть вынуждены были следовать принятым в этих операционных системах архитектурным решениям и технологиям и основывать на них свои собственные разработки. Мечта Столлмана о научной разработке ПО, свободной от решений, движимых коммерческими целями, была неосуществима, пока в основе свободной разработки лежало патентованное UNIX-совместимое ядро, исходные тексты которого оставались тайной для разработчиков.

Становление LINUX систем

В 1991 году Линус Торвалдс, финский студент, чрезвычайно увлёкся идеей написать совместимое с UNIX ядро операционной системы для своего персонального компьютера с процессором ставшей очень широко распространённой архитектуры Intel 80386. Прототипом для будущего ядра стала операционная система MINIX: совместимая с UNIX операционная система для персональных компьютеров, которая загружалась с дискет и умещалась в очень ограниченной в те времена памяти персонального компьютера. MINIX был создан Эндрю Таненбаумом в качестве учебной операционной системы, демонстрирующей архитектуру и возможности UNIX. Именно полноценное ядро для своего ПК и хотел сделать Линус Торвалдс. Название своему ядру он дал freax, но позже оно было изменено хозяином ftp сервера на Linux — гибрид имени создателя и слова UNIX.

В 1991 году Линус Торвалдс, финский студент, чрезвычайно увлёкся идеей написать совместимое с UNIX ядро операционной системы для своего персонального компьютера с процессором ставшей очень широко распространённой архитектуры Intel 80386. Прототипом для будущего ядра стала операционная система MINIX: совместимая с UNIX операционная система для

персональных компьютеров, которая загружалась с дискет и умещалась в очень ограниченной в те времена памяти персонального компьютера. MINIX был создан Эндрю Таненбаумом в качестве учебной операционной системы, демонстрирующей архитектуру и возможности UNIX. Именно полноценное ядро для своего ПК и хотел сделать Линус Торвальдс. Название своему ядру он дал freax, но позже оно было изменено хозяином ftp сервера на Linux — гибрид имени создателя и слова UNIX.

GNU и Linux

Однако как нельзя сделать операционную систему без ядра, так и ядро будет бесполезно без утилит, которые использовали бы его возможности. Благодаря проекту GNU Линус Торвальдс сразу получил возможность использовать с Linux свободные утилиты: bash, компилятор gcc, tar, gzip и многие другие уже известные и широко используемые приложения, которые могли работать с его UNIX-совместимым ядром. Так Linux сразу попал в хорошее окружение и в сочетании с утилитами GNU представлял собой очень интересную среду для разработчиков программного обеспечения даже на самой ранней стадии своего развития.

Принципиальным шагом вперёд было именно то, что из ядра Linux и утилит и приложений GNU впервые стало возможно сделать полностью свободную операционную систему, то есть работать с компьютером и, более того, разрабатывать новое программное обеспечение, пользуясь только свободным программным обеспечением. Идеал полностью некоммерческой разработки, сформулированный Столлманом, теперь мог быть воплощён в жизнь.

Вскоре появлялись теоретические возможности воплощения идеала, но это не означало его немедленной практической реализации. Совместимость Linux и утилит GNU была обусловлена тем, что и то, и другое писалось с ориентацией на одни и те же стандарты и практику. Однако в рамках этой практики (то есть при наличии множества различных UNIX-систем) оставался большой простор для несовместимости и различных решений. Поэтому на начальном этапе разработки ядра каждое заработавшее на Linux приложение GNU было для Линуса очередным достижением. Первыми стали bash и gcc. Таким образом, сочетание GNU и Linux давало возможность создать свободную операционную систему, но само по себе ещё не составляло такой системы, потому что Linux и различные утилиты GNU оставались разрозненными программными продуктами, написанными разными людьми, не всегда принимавшими в расчёт то, что делали другие. Основным же свойством любой системы является согласованность её компонентов.

Возникновение дистрибутивов

После определённого периода разработки на Linux уже стабильно работал ряд важнейших утилит GNU. Скомпилированное ядро Linux с небольшим комплектом скомпилированных уже на Linux утилит GNU составляло набор инструментов для разработчика программного обеспечения, желающего использовать свободную операционную систему на своём персональном компьютере. В таком виде Linux уже не только годился для разработки Linux, но и представлял собой операционную систему, в которой можно было уже выполнять какие-то прикладные задачи. Конечно, первое, чем можно было заниматься в Linux — писать программы на Си.

Когда задача получить компьютер с постоянно работающей на нём системой Linux стала востребованной и довольно распространённой, разработчики в хельсинкском и тexasском университетах создают собственные наборы дискет, с которых скомпилированное ядро и основные утилиты можно записать на жёсткий диск, после чего загружать операционную систему прямо с него. Эти наборы дискет стали первыми прототипами современных дистрибутивов Linux — комплектов программного обеспечения, на основе которых можно получить работающую операционную систему на своём компьютере. Нужно отметить, что в дистрибутив Linux с самого начала входили программные продукты GNU. На самом деле, всякий раз, когда говорится «операционная система Linux», подразумевается «ядро Linux и утилиты GNU». Фонд свободного ПО рекомендует называть это операционной системой Linux.

Всё программное обеспечение для Linux было открытым, поэтому вскоре стало появляться всё больше прикладных программ для Linux, которые использовались всё большим сообществом, отчего становились надёжнее и получали всё новую функциональность. В конце концов возникает идея, что из Linux и GNU-приложений для Linux целенаправленными усилиями небольшой группы разработчиков можно делать целостные операционные системы, подходящие для очень широкого круга пользователей и продавать эти системы пользователям за деньги как аналог и альтернативу существующим патентованным операционным системам.

Всякий раз, когда свободное программное обеспечение оказывается востребованным, немедленно возникает множество альтернативных решений — так произошло и с дистрибутивами

Linux. После 1995 года возникло (и продолжает возникать) огромное количество коммерческих компаний и свободных сообществ, которые ставят своей задачей подготовку и выпуск дистрибутивов Linux. У каждого из них — свои особенности, своя целевая аудитория, свои приоритеты. К настоящему времени на рынке дистрибутивов выделилось несколько лидеров, которые предлагают более или менее универсальные решения и наиболее широко известны и используются. Помимо уже названных Red Hat и Debian следует назвать в ряду дистрибутивов, ориентированных на рядового пользователя, немецкий SuSE и французский Mandriva (до 2005 года — Mandrake), среди адресованных специалистам — Gentoo. Но помимо «крупных» игроков на рынке дистрибутивов есть гораздо большее количество менее распространённых дистрибутивов. Теперь перед пользователем, желающим установить Linux, встаёт вопрос выбора дистрибутива. Критерии выбора — и задачи, которые предполагается решать с помощью Linux, и уровень подготовки пользователя, и технологии, и предстоящие контакты с тем сообществом, которое занимается разработкой дистрибутива

Краткая характеристика современных дистрибутивов LINUX

Linux Mint

Некогда он был популярнейшим дистрибутивом Linux, но сейчас его перегнали MX Linux и Manjaro. Тем не менее Mint выгодно отличается от них своей стабильностью и надёжностью. Это очень простая в работе система, основанная на Ubuntu.

Linux Mint оснащён лёгким и понятным интерфейсом (оболочка Cinnamon для современных компьютеров и MATE для старых машин) и удобным менеджером приложений, так что у вас не возникнет проблем с поиском и установкой программ.

Плюсы: простота, забота об обычных пользователях. Вам не потребуется каких-то специфических знаний, чтобы установить и использовать Mint.

Минусы: большое количество предустановленного ПО, которое может никогда не пригодиться.

Альтернатива: Zorin OS — дистрибутив, основанный на Ubuntu. Тоже рассчитан на новичков. Его интерфейс напоминает Windows, что должно облегчить переход неопытным пользователям.

Manjaro

Основан на Arch — невероятно мощном и функциональном дистрибутиве. Его философия KISS (Keep It Simple, Stupid), в противоположность названию, делает Arch слишком сложным для начинающих. Устанавливается этот дистрибутив только через командную строку.

Дистрибутив поставляется со множеством оболочек рабочего стола на выбор: функциональной KDE, GNOME для планшетных экранов, Xfce, LXDE и другими. Установив Manjaro, вы можете быть уверены, что первыми получите обновления.

Плюсы: AUR, благодаря которому можно без лишних движений установить любое приложение. Всегда свежее ПО.

Минусы: своеобразное оформление оболочек рабочего стола. Впрочем, вам ничего не мешает заменить его.

Альтернативы: Fedora — дистрибутив от компании Red Hat, своего рода тестовый полигон, на котором пользователи пробуют самые новые фишки и возможности приложений и системы. Chakra — ещё один дистрибутив, основанный на Arch, но оснащённый красивым интерфейсом KDE и удобным установщиком.

Arch

По сути, Arch — это конструктор, из которого опытный пользователь построит что угодно. Если вы хотите разобраться, как устроен Linux, или поэкспериментировать — вам определённо стоит попробовать Arch. Если же консоль вызывает страх, то дистрибутив будет слишком сложен для вас.

Плюсы: AUR, предоставляющий огромное количество приложений, всегда свежее ПО, широкие возможности настройки, великолепная полезная Wiki, высокое быстродействие.

Минусы: дистрибутив не подходит для новичков. Вам придётся изучать много команд и читать руководства.

Альтернативы: Slackware — один из старейших дистрибутивов. Его поклонники говорят: «Если вы изучите Slackware — вы узнаете весь Linux». Отличается стабильностью и консервативностью. Gentoo — в этом дистрибутиве нет удобных магазинов приложений. Вместо этого вам предстоит вручную компилировать нужные вам программы из исходных кодов, любезно предоставленных разработчиками. В теории это даёт небольшой бонус к быстродействию системы.

Kubuntu

Kubuntu красив, функционален и легко настраивается. С ним легко справятся даже начинающие пользователи. Это стабильная и отполированная система, которая предоставляет все необходимые функции для домашнего настольного ПК.

Плюсы: большой ассортимент пакетов, замечательный набор приложений KDE и огромное количество настроек интерфейса.

Минусы: в Kubuntu используется стабильная версия KDE, а это значит, что новейшие фишки этой оболочки приходят сюда с запозданием. Если хотите опробовать свежайшую KDE, к вашим услугам KDE Neon.

Альтернативы: openSUSE — система, поддерживаемая компанией Novell. Дистрибутив достаточно прост для начинающих, использует графическую среду KDE. Имеет два режима обновления: Leap (стабильный, как в Ubuntu) и Tumbleweed (скользящий релиз с самыми интересными новинками, как в Arch).