Содержание

Глава 1
Введение

1.1. Введение

В 1994 году компьютерное сообщество отметило двадцатипятилетие с момента появления операционной системы UNIX. После своего рождения в 1969 году система была перенесена на множество различных аппаратных платформ, появилось большое количество ее реализаций, созданных коммерческими компаниями, учебными заведениями и научно-исследовательскими организациями. Система UNIX начала свое развитие с небольшого набора программ и за годы переросла в гибкую ОС, использующуюся для работы огромного количества программных сред и приложений. На сегодняшний день существуют варианты UNIX для самых различных аппаратных платформ, начиная от небольших встроенных процессоров, рабочих станций и настольных систем и заканчивая высокопроизводительными многопроцессорными системами, объединяющими большое количество пользователей.

Операционная система UNIX — это среда выполнения и системные службы, под управлением которых функционируют входящие в набор ОС пользовательские программы, утилиты и библиотеки. Эта книга рассказывает о разработке и реализации самой системы, не описывая программы и утилиты, выполняющиеся под ее управлением. Система UNIX начала свою жизнь в недрах Bell Telephone Laboratories (BTL), которая и ответственна за все ее ранние реализации. Сначала система получила распространение среди нескольких компаний и учебных заведений. Именно этот факт повлиял на дальнейшее развитие различных реализаций UNIX. Все они поддерживали только набор внутренних интерфейсов, приложений и возможностей, обычно ожидаемых от стандартной «системы UNIX». Внутреннее устройство каждой было различным, отличаясь семантикой интерфейсов и набором предоставляемых «дополнительных» возможностей. В этой книге уделяется особое внимание описанию основополагающих реализаций UNIX, таких как *System V Release 4* (*SVR4*) компании Novell, *Berkeley Software Distribution* (*4xBSD*) Калифорнийского университета и *Mach* университета Карнеги—Меллона. Здесь также обсуждается большое количество коммерческих ва-

риантов системы, таких как *SunOS* и *Solaris* компании Sun Microsystems, *Digital UNIX* компании Digital Equipment Corporation и *HP-UX* корпорации Hewlett-Packard Corporation.

Эта глава посвящена описанию развития систем UNIX. Сначала вы увидите краткий рассказ о рождении, становлении и принятии системы компьютерным сообществом. Затем будут описаны факторы, повлиявшие на ее эволюцию, и, наконец, будут отмечены возможные направления дальнейшего развития UNIX.

1.1.1. Краткая история

Перед тем как начать изучение операционной системы UNIX, полезно немного узнать о ее истории и эволюции. В следующих разделах мы проследим развитие UNIX от зарождения ее в недрах AT&T до нынешнего современного, немного хаотичного состояния в виде набора операционных систем, реализованных под различные платформы, различными авторами и существующих в самых различных вариантах. Более полное изложение истории развития UNIX можно найти в других публикациях, например в книге Питера Салюса (Peter Salus) «A Quarter Century of UNIX» [1]. Эта глава расскажет только об основных событиях, произошедших в истории системы UNIX.

1.1.2. Начало

В конце 60-х годов Bell Telephone Laboratories совместно с компанией General Electric и Массачусетским технологическим институтом организовали совместный проект, целью которого была разработка операционной системы под названием *Multics* [2]. Проект был аннулирован в марте 1969, но некоторые разработчики из BTL продолжили поиск интересных идей для последующей реализации. Один из участников проекта, Кен Томпсон (Kenneth Thompson), написал игровую программу под названием «Космическое путешествие» («Space Travel») и нашел для ее запуска малоиспользуемый в те годы компьютер *PDP-7* (созданный Digital Equipment Corporation). Однако этот компьютер не имел собственной среды разработки программ, поэтому Томпсон перенес свою программу на другую машину, Honeywell 635, работавшую под управлением ОС GECOS, и затем создал перфоленту со своей программой для PDP-7.

С целью совершенствования разработки «Космического путешествия» Томпсон совместно с Денисом Ритчи (Dennis Ritchie) начал разработку рабочей среды для PDP-7. Первым созданным компонентом стала простая файловая система, дальнейшее развитие которой впоследствии появилось в первых версиях UNIX и сейчас известно как *System V file system* (*s5fs*). Чуть позже ими были добавлены подсистема обработки, простой командный

интерпретатор под названием shell (который позже развился в Bourne shell, [4]), а также небольшой набор утилит. Система стала самодостаточной и не требовала больше среды GECOS. Авторы назвали свою систему *UNIX* в честь проекта Multics[1].

В следующем году Томпсон, Ритчи и Йозеф Оссанна (Joseph Ossanna) добились того, что BTL приобрела машину Digital *PDP-11* для обработки документации в отделе патентов. Затем они экспортировали UNIX на эту машину и добавили несколько утилит обработки текста, в том числе редактор ed и инструмент отображения текста runoff. Томпсон также разработал новый язык программирования *B* (усовершенствовав тем самым язык *BPCL*, [5]) и написал на нем несколько первых ассемблеров и утилит. Язык B был интерпретируемым, вследствие чего обладал низкой производительностью. Позже Ритчи усовершенствовал свою разработку, назвав результат *C*. Язык C поддерживал типы и структуры данных. Успех языка C являлся основой успешного развития системы UNIX.

ОС UNIX становилась все более популярной внутри BTL. В ноябре 1971 года Ритчи и Томпсон под руководством Дуга Мак-Илроя (Doug McIlroy) опубликовали первую редакцию «Руководства для программиста UNIX». В дальнейшем появилось ровно 10 изданий этого руководства — по количеству версий систем UNIX, созданных в лабораториях BTL.

Первые несколько реализаций ОС использовались только внутри BTL. Третья версия, вышедшая в феврале 1973, включала в себя компилятор языка C под названием cc. В том же году система была переписана на языке C, в результате чего в ноябре того же года появилась версия 4. Это в высокой степени повлияло на будущий успех системы UNIX. Томпсон и Ритчи создали первую работу о UNIX под названием «The UNIX Timesharing System» [6], которая была представлена на симпозиуме по операционным системам (*ACM Symposium on Operating Systems*, *SOSP*) в октябре 1973 года и опубликована в июле 1974 года в *Communications of ACM*[2]. Эта работа стала первой публикацией, возвестившей миру о UNIX.

---

[1] Питер Салюс в своей книге рассказывает, что этимология слова UNIX обязана своим происхождением шуткам коллег-хакеров. Multics была многопользовательской системой, а первая UNIX работала всего с двумя пользователями. Латинский корень «много» заменили на «один» («единственный»). Получилось — UNICS (Uniplexed Information and Computing Service). Название понравилось, поскольку напоминало об участии сотрудников Bell Labs в разработке Multics. Позже UNICS было изменено на UNIX. Том ван Влек, лично зная всех персонажей, «из первых рук» уточняет (http://www.multicians.org/unix.html), что название UNIX предложил Брайан Карниган как «Multics without balls». История имела продолжение. Когда хакерская коалиция распалась, Ричард Столлмен решил создать систему, «совместимую с UNIX, чтобы она была переносимой и чтобы пользователи UNIX могли бы легко на нее перейти. Аббревиатура GNU была выбрана для нее в соответствии с хакерской традицией как рекурсивный акроним выражения «GNUTs Not UNIX » («GNU — это не UNIX »)». — *Прим. ред.*

[2] Позже эта работа была подвергнута изменению и переиздана в виде книги [7].

1.1.3. Распространение

В 1956 году в результате антимонопольной судебной тяжбы Министерства юстиции США против компаний AT&T и Western Electric Company корпорация AT&T вынуждена была подписать согласительный документ с правительством. Это соглашение запрещало компании производить какое-либо оборудование, не относящееся к телефонам или телеграфу, а также вести дела в областях, отличных от «общих служб доставки сигнала».

В результате компания AT&T не могла заниматься продажей продукции, относящейся к вычислительной технике. С другой стороны, конференция SOSP показала наличие большого спроса на программное обеспечение UNIX. Корпорация AT&T распространяла свою систему в высших учебных заведениях для использования в образовательных и научных целях под простыми лицензионными условиями, что не противоречило подписанному соглашению. Компания не рекламировала созданную систему и не продвигала ее на рынке сбыта, а также не вела поддержку ее реализаций. Одним из первых лицензию на использование UNIX от компании AT&T получил университет Беркли (Berkeley) в Калифорнии. Это произошло в декабре 1973 года.

Такие условия дали возможность системе UNIX довольно быстро распространиться по всему миру. О широкой географии использования ОС говорит тот факт, что к 1975 году система была установлена в таких учебных заведениях, как Еврейский университет в Иерусалиме, университет Нового Южного Уэльса (Австралия) и университет Торонто (Канада). Первым переносом системы на новую аппаратную платформу стала ее реализация для машины *Interdata*, полностью выполненная университетом Уолонгонга (Wollongong) самостоятельно в 1976 году. Годом позже аналогичный процесс был осуществлен Ритчи и Стивом Джонсоном в BTL.

Седьмая версия операционной системы UNIX, выпущенная в январе 1979 года, являлась первой настоящей переносимой ОС, что послужило большим толчком в ее дальнейшем развитии. Эта версия изначально работала на PDP-11 и Interdata 8/32. Она, с одной стороны, была устойчивее и намного функциональнее своей предшественницы, версии 6, однако, с другой стороны, работала значительно медленнее. Существовало несколько лицензий ОС, позволяющих увеличить ее производительность при использовании в различных областях. Компания AT&T позже вставила многие из этих разработок в последующие версии UNIX. Кооперация между разработчиками и пользователями системы (которая, к сожалению, перестала быть возможной после коммерческого успеха ОС) была ключевым фактором, обусловившим быстрый рост и увеличение популярности UNIX.

Вскоре система UNIX была импортирована на другие аппаратные платформы. Корпорация Microsoft совместно с Santa Cruz Operation (SCO) перенесла систему на компьютеры под управлением процессора Intel 80x86, в результате чего появилась ОС *XENIX* — один из первых коммерческих вариантов UNIX. В 1978 году компания Digital представила свой новый 32-разрядный компьютер VAX-11 и предложила группе разработчиков отделения BTL в Холмделе,

штат Нью-Джерси, перенести UNIX на эту машину. Так возник первый вариант системы UNIX для 32-разрядного компьютера, который был назван *UNIX/32V*. Копия этой системы была передана Калифорнийскому университету, где была переработана и стала основой ОС 3BSD, появившейся в 1979 году.

1.1.4. BSD

Калифорнийский университет в Беркли получил одну из первых лицензий на операционную систему UNIX в декабре 1974 года. За несколько лет группа выпускников университета, в состав которой входили Билл Джой (Bill Joy) и Чак Хэлей (Chuck Haley), разработала несколько утилит для этой системы, в том числе редактор ex (позже сопровождавшийся vi) и компилятор языка Паскаль. Все созданные приложения были собраны в единый пакет под названием *Berkeley Software Distribution* (*BSD*) и продавались весной 1978 года по цене $50 за одну лицензию. Первые версии BSD (2BSD появилась в конце 1978 года) состояли из дополнительных приложений и утилит, сама же операционная система тогда еще не подвергалась изменению или передаче. Одной из первых разработок Джоя стала оболочка C shell [7], имевшая такие возможности, как управление заданиями и ведение истории команд, отсутвовавшие в то время в оболочке Bourne.

В 1978 году университет Беркли приобрел машину VAX-11/780 и операционную систему UNIX/32V, портирование которой на этот компьютер осуществила группа BTL в Холмделе, штат Нью-Джерси. Компьютер VAX имел 32-разрядную архитектуру и мог использовать до 4 Гбайт адресного пространства, однако имел всего лишь 2 Мбайт физической памяти. Примерно в то же время Озалп Бабаоглу (Ozalp Babaoglu) разработал систему страничной виртуальной памяти для VAX и добавил ее в ОС UNIX. В результате в конце 1979 года появилась новая версия ОС, 3BSD, которая стала первой операционной системой, созданной университетом Беркли.

После появления системы виртуальной памяти агентство DARPA (Defence Advanced Research Projects Agency) начало финансирование разработки систем UNIX в Беркли. Одной из главных задач, стоявших перед DARPA, являлась интеграция в создаваемой системе набора протоколов *TCP/IP* (Transmission Control Protocol/Internet Protocol). При финансовой поддержке агентства DARPA Berkeley выпустил несколько вариантов системы BSD, объединенных под общим названием *4BSD*: *4.0BSD* в 1980 году, *4.1BSD* в 1981[1], *4.2BSD* в 1983, *4.3BSD* в 1986 и *4.4BSD* в 1993.

Команде Беркли принадлежало авторство большого количества важных технических усовершенствований системы. Кроме уже упомянутых новшеств (виртуальной памяти и интеграции протоколов TCP/IP), в системе BSD UNIX была представлена файловая система *Fast File System* (*FFS*), надежная реализация сигналов и технология сокетов. В 4.4BSD оригинальная разработка виртуальной памяти была заменена новой версией, базирующей-

[1] Эта версия системы, в свою очередь, имела три различных варианта: 4.1a, 4.1b и 4.1c.

ся на Mach (см. раздел 1.1.7), а также были добавлены другие возможности, например файловая система с ведением журнала.

Работа над системой UNIX производилась группой CSRG (Computer Science Research Group). Однако после выпуска 4.4BSD группа приняла решение закрыть проект и завершить разработку систем UNIX. Наиболее важными причинами этого решения были:

- ✦ уменьшение финансирования и выделения грантов;
- ✦ возможности, представленные в системе BSD, к тому времени уже были реализованы во многих коммерческих проектах;
- ✦ операционная система становилась слишком большой и сложной для разработки и поддержки силами небольшой группы программистов.

Для продвижения и продажи 4.4BSD как коммерческого продукта была создана компания Berkeley Software Design, Inc (BSDI). К тому времени почти весь код оригинальной системы UNIX был заменен разработчиками из Беркли, поэтому компания BSDI утверждала, что созданная ею версия системы *BSD/386* полностью свободна от лицензионных ограничений AT&T. Однако подразделение AT&T, UNIX System Laboratories, занимавшееся разработками UNIX, все-таки подало иск против BSDI и управляющего совета Калифорнийского университета. Компания обвиняла их в нарушении авторских прав, невыполнении условий соглашения, в незаконном перехвате коммерческих секретов, а также выступала резко против использования BSDI телефонного номера 1-800-ITS-UNIX для продажи исходных кодов своей системы. Университет подал ответный иск, вследствие чего продажи BSD/386 были приостановлены. В результате 4 февраля 1994 года обе стороны договорились между собой вне здания суда и отменили иски друг к другу, после чего компания BSDI анонсировала новый продукт, *4.4BSD-lite*, продаваемый без исходных кодов примерно по $1000 за пакет[1].

[1] Разработчики BSD UNIX были вынуждены вести юридическую битву с AT&T, что замедлило темпы развития системы, а успех SVR5 и альянса единой UNIX во главе с SCO к 2000 году окончательно выбил почву из-под ног университета Беркли и приверженцев коммерциализации этой ветви развития UNIX. Спецификация 4.4BSD используется во многих коммерческих ОС, но суммарный объем продаж исчезающе мал, прежде всего, по причине сильнейшей конкуренции со стороны бесплатных систем FreeBSD (www.freebsd.org), OpenBSD (www.openbsd.org) и NetBSD (www.netbsd.org). Последние версии FreeBSD (в основе которой лежит 4.4BSD-lite) ни в чем не уступают коммерческим разработкам. FreeBSD 4.7 (октябрь 2002) улучшена относительно версии 4.6, поддерживает архитектуры Intel и ALPHA и может быть установлена из Сети с одного из множества анонимных ftp-серверов. FreeBSD 5.0 (январь 2003) явилась результатом трехлетнего труда и работает на 64-разрядных архитектурах Intel и SPARC. В ее файловой системе UFS2, следующем поколении UFS, преодолен терабайтный барьер на размер файла. Функции резервного копирования реализуются на основе Background filesystem checking (bgfsck) и моментальных снимков. Модель безопасности включает экспериментальный сервис Mandatory Access Controls (MAC) и предоставляет более гибкие возможности администраторам. Работа в многопроцессорных системах обеспечивается высоким уровнем грануляции ядра. Такие новшества, как инфраструктура устройств памяти GEOM и файловая система виртуальных устройств DEVFS, облегчают управление устройствами памяти. — *Прим. ред.*

1.1.5. System V

Вернемся снова к истории AT&T. Судебные битвы корпорации с министерством достигли кульминации в 1982 году, с выходом разграничивающего постановления. Во исполнение его предписаний корпорация Western Electric распалась на части, AT&T лишилась своих региональных отделений, которые стали основой «детей Белла» («Baby Bells»), Bell Telephone Laboratories была отделена и переименована в AT&T Bell Laboratories. Также корпорация AT&T отныне получила возможность заниматься бизнесом в компьютерных отраслях.

Работа над системой UNIX по-прежнему велась группой разработчиков из BTL, но создание внешних реализаций системы постепенно перешло к UNIX Support Group, затем к UNIX Development Group и, наконец, к AT&T Information Systems. Этими группами были созданы *System III* в 1983 году, *System V* в 1983, *System V Release 2* (*SVR2*) в 1983 и *System V Release 3* (*SVR3*) в 1987. Корпорация AT&T агрессивно продвигала System V на рынок операционных систем. На основе этой ОС было разработано несколько различных коммерческих вариантов UNIX.

В System V были впервые представлены многие новые возможности и средства. Например, сегментная реализация виртуальной памяти отличалась от варианта, предлагаемого BSD. В SVR3 появилось средство взаимодействия процессов (поддерживающее совместное использование памяти, семафоров и очередей сообщений), удаленное разделение файлов, общие библиотеки, а также поддержка `STREAMS` для драйверов устройств и сетевых протоколов. Последняя версия операционной системы System V Release 4 (*SVR4*) будет подробнее описана в разделе 1.1.10[1].

1.1.6. Коммерциализация

Увеличивающаяся популярность UNIX вызвала интерес к этой ОС со стороны нескольких компаний, которые приняли решение начать выпуск и продажу своих собственных вариантов системы. Производители брали за основу одну из базовых реализаций UNIX от AT&T или Беркли, переносили вы-

[1] В марте 2000 года компания SCO представила новую ОС на основе System V Release 5 (SVR5), усовершенствованной SVR4.2, которая включила в себя более производительную сетевую подсистему — улучшенные механизмы синхронизации процессов, планирования и управления памятью, поддержку до 32 процессоров в сервере и технологий NUMA и I2O, файлов размером до 1 Тбайт, 76 800 Тбайт внешней памяти, разбиваемой до 512 логических томов. SVR5 работает с 64-гигабайтной RAM (прямая адресация к 4 Гбайт, остальная — в режиме расширенной физической адресации (PAE)). Технология Multi-path I/O позволяет использовать несколько контроллеров ввода-вывода и удвоение дисков. Поддержана 64-разрядная журнальная отказоустойчивая файловая система VxFS компании Veritas, 64-разрядый API и системные вызовы. Дополнительно: графические Java-консоли администрирования (SCOadmin), программная поддержка RAID, кластеризация, файл-серверы и серверы печати, совместимые с Windows, и многое другое. — *Прим. ред.*

бранную систему на свои машины и добавляли в ОС свои *дополнительные* возможности. Первой компанией, начавшей в 1977 году продажу собственного коммерческого варианта UNIX, стала Interactive Systems. Продукт назывался *IS/1* и работал на PDP-11.

В 1982 году Билл Джой покинул Беркли и стал одним из создателей корпорации Sun Microsystems, выпустившей свой вариант UNIX на основе 4.2BSD и назвавшей его *SunOS* (позже компанией была разработана еще одна версия системы, *Solaris*, базирующаяся на 4.3BSD). Компании Microsoft и SCO совместно создали систему *XENIX*. Позже SCO перенесла SVR3 на платформу 80386 и выпустила эту систему под названием SCO UNIX. В 80-х годах появилось множество различных коммерческих вариантов системы UNIX, в том числе *AIX* от IBM, *HP-UX* от Hewlett-Packard Corporation и *ULTRIX* (выпущенной вслед за *DEC OSF/1*, переименованной позже в *Digital UNIX*) от компании Digital[1].

В коммерческих вариантах UNIX были представлены многие новые возможности, некоторые из которых позже были встроены и в базовые системы. В SunOS была реализована сетевая файловая система Network File System (NFS), интерфейс vnode/vfs, поддерживающий работу с различными типами файловых систем, а также новая архитектура виртуальной памяти, адаптированная позже в SVR4. Система AIX одна из самых первых начала поддержи-

[1] Выходя за «рамки повествования книги», следует заметить, что на сегодняшний день роли на рынке коммерческих систем UNIX достаточно четко определились. Интерес к коммерческим вариантам UNIX усилился в связи с успехом Linux. Радикальные перемены произошли в 1997–2000 годах, когда компания SCO представила System V Release 5, являющуюся усовершенствованной версией SVR4.2, и на ее базе выпустила ОС UnixWare 7. В настоящее время SCO оккупировала 80% рынка ОС для серверов UNIX на платформе Intel. UnixWare 7 наследовала лучшее из UnixWare 2 и SCO OpenServer 5. UnixWare 7 поддерживает процессоры Pentium, 64-разрядный Merced и 64-разрядную RISC-архитектуру (PowerPC). SCO организовала альянс с IBM и Sequent (IBM собиралась принести в жертву AIX во благо стандартного диалекта UNIX для платформ IA-32/64, поскольку компьютеры с процессорами Intel составляют половину рынка систем UNIX), Intel и др. по разработке единой UNIX, условно названной Monterey. Результатом должно было стать объединение UnixWare, элементов AIX и Sequent Dynix. Принять новую ОС в качестве стандарта согласились Compaq, Siemens Nixdorf, Hyundai, ICL HPS, Fujitsu/ICL. Ряд производителей объявили о разработке версий своих продуктов конкретно под UnixWare, в том числе Informix, Netscape, Sybase и Oracle. Увы, в 2001 году IBM отказалась от участия в проекте, и он был прекращен. Теперь, спустя два года, SCO пытается отсудить $1 млрд за то, что IBM якобы использовала конфиденциальные данные, полученные от SCO, в собственных продуктах (AIX 5). А пока суд да дело, Hewlett-Packard и Sun Microsystems пошли своей дорогой, продвигая собственные решения, HP/UX и Solaris. Впрочем, эти компании не делают погоды, поскольку рынок сбыта Solaris узок, внимание компании к платформе Intel пониженное (более 1 млн зарегистрированных пользователей), поэтому доля Sun на платформе Intel оценивается не более чем в 10%. В результате ее маркетинговые акции а-ля «бесплатная Solaris» заканчиваются возвращением в русло коммерции. Ей также требуется перенос всех возможностей со SPARC на IA-64 (на процессорах Intel поддержан только 32-разрядный режим). HP вообще не имеет версии для Intel, что отодвигает сроки ее активного внедрения на рынок. А после нужно будет решать проблему отсутствия приложений. — *Прим. ред.*

вать файловую систему с ведением журналов (Journaling File System, JFS) для UNIX. ULTRIX стала одной из первых систем UNIX с поддержкой многопроцессорной архитектуры.

1.1.7. Mach

Одной из главных причин популярности системы UNIX являлись ее простота и небольшой размер, сочетающиеся со множеством полезных утилит. Но после того как система стала поддерживать все большее количество возможностей, ее ядро постепенно становилось большим, сложным и громоздким. Многие специалисты пришли к мнению, что развитие UNIX постепенно уходит от предполагаемого пути, приведшего когда-то систему к успеху.

В середине 80-х годов разработчики из университета Карнеги—Меллона [8] в Питтсбурге приступили к созданию новой операционной системы под названием *Mach*. Их целью была разработка *микроядра*, включающего небольшой набор утилит, служащих для реализации остальных системных функций пользовательского уровня. Архитектура Mach должна была поддерживать интерфейс программирования UNIX, работать как на однопроцессорных, так и на многопроцессорных системах, а также подходить для распределенных сред. Разработчики надеялись, что, создав новую систему «с нуля», они могут избежать множества проблем, имевшихся в текущих вариантах UNIX.

Одним из первых приближений к реализации задуманных планов стало создание микроядра, в котором были выделены некоторые основные функции, в то время как большинство возможностей системы исходило от набора внешних по отношению к ядру процессов, называемых *серверами*. Система Mach также имела еще одно существенное преимущество: она никак не зависела от лицензий AT&T, что сделало ее привлекательной для многих производителей. Самой популярной версией системы стала *Mach 2.5*, и многие коммерческие ОС, такие как *OSF/1* или *NextStep*, были созданы на ее основе. Ранние версии системы имели монолитные ядра с поддержкой интерфейса 4.4BSD UNIX на высоком уровне. Первой реализацией идеи микроядра стала система Mach 3.0.

1.1.8. Стандарты

Распространенность различных реализаций UNIX привела к появлению проблем совместимости. Несмотря на то что все существующие варианты на первый взгляд «похожи на UNIX», на самом деле они имеют существенные различия между собой. Существование отличий было заложено изначально, за счет наличия двух веток развития UNIX, «официальной» системы AT&T System V и альтернативного варианта BSD, создаваемого в Беркли. Появление коммерческих вариантов UNIX еще более усложнило проблему.

Системы System V и 4BSD существенно отличались. Они имели различные несовместимые между собой файловые системы, реализации поддержки сетей и архитектуры виртуальной памяти. Некоторые различия были обусловлены дизайном ядра систем, но большинство из них находились на уровне программирования интерфейса. Это привело к невозможности создания сложных приложений, работающих без внесения каких-либо изменений в обеих операционных системах.

Все коммерческие варианты UNIX строились на основе либо System V, либо BSD, к которым производители добавляли дополнительные возможности. Именно эти добавления часто оказывались непереносимыми на иные платформы. В результате создатели приложений тратили огромное количество времени и усилий, для того чтобы их программы нормально функционировали в различных реализациях UNIX.

Для решения проблемы необходимо было разработать некий стандартный набор интерфейсов, чем и занялись несколько групп энтузиастов. В результате миру предстало множество стандартов, столь же многочисленных и отличающихся друг от друга, как и существовавшие в те времена варианты UNIX. Однако большинство производителей признала только несколько из созданных стандартов, в том числе «*Определение интерфейса System V*» (System V Interface Definition, SVID) компании AT&T, спецификации организации IEEE под названием *POSIX* и «*Руководство по переносу X/Open*» (X/Open Portability Guide) консорциума X/Open.

В каждом из стандартов описывалось взаимодействие между программами и операционной системой и не затрагивался вопрос реализации самого интерфейса взаимодействия. В них определялись наборы функций и подробно приводились их конструкции. Совместимые системы должны удовлетворять требованиям, изложенным в стандартах, однако реализация необходимых функций могла быть произведена как на уровне ядра, так и на уровне библиотек пользователя.

Стандарты также определяли поднабор функций, предлагаемых большинством систем UNIX. Теоретически, если пользователь будет использовать при написании приложения только те функции, которые входят в этот набор, то созданное приложение будет переносимо на любую систему, совместимую со стандартами. Это заставляло разработчиков программ использовать дополнительные возможности конкретного варианта системы, а также производить оптимизацию своих программ под конкретную аппаратную платформу или операционную систему только в том случае, если их исходные коды легко переносимы.

Стандарт SVID представляет собой подробную спецификацию программного интерфейса System V. Корпорация AT&T выпустила три версии стандарта — *SVID1*, *SVID2* и *SVID3*, описывающие соответственно ОС SVR2, SVR3 и SVR4 [10]. AT&T предоставила возможность производителям систем называть свои продукты System V только в том случае, если они отвечают

требованиям SVID. Корпорация также выпустила пакет *System V Verification Suite* (*SVVS*), который проверял операционные системы на соответствие SVID.

В 1986 году организация IEEE поручила специальному комитету разработать и опубликовать стандарты на среды операционных систем. Для их обозначения было придумано название POSIX (Portable Operating System based on UNIX, что переводится как «Переносимые операционные системы, основанные на UNIX»). Эти документы описывали компоненты ядра систем SVR3 и 4.3BSD. Стандарт *POSIX1003.1*, более известный как *POSIX.1*, был опубликован в 1990 году [11]. Многие производители приняли этот стандарт, так как он не ограничивался каким-то одним вариантом системы UNIX.

X/Open — это международный консорциум производителей компьютерной техники и программного обеспечения. Он был сформирован в 1984 году. Его целью являлась не только разработка новых стандартов, но и создание открытой среды *Common Applications Environment* (Общей программной среды, CAE), базирующейся на **уже** существующих стандартах. Консорциум опубликовал семитомный труд «X/Open Portability Guide» (XPG), последнее (четвертое) издание которого вышло в 1993 году [12]. Материал руководства был основан на стандарте POSIX.1, расширял его и описывал многие дополнительные области, такие как интернационализация, оконные интерфейсы и обработка данных.

1.1.10. OSF и UI

В 1987 году корпорация AT&T, осознавая непринятие общественностью ее лицензионной политики, принимает решение о закупке 20% акций Sun Microsystems. AT&T и Sun решают заняться совместной разработкой SVR4, следующей версии операционной ОС AT&T System V UNIX. Корпорация Sun объявляет, что будущая операционная система станет базироваться на SVR4, в отличие от SunOS, основу которой составляла ранее система System V.

Эти заявления вызвали бурную реакцию со стороны других производителей систем, которые поняли, что созданное объединение даст корпорации Sun огромное преимущество перед остальными производителями. В ответ группа компаний, в которую входили Digital, HP, IBM, Apollo и другие, объявила в 1988 году о создании объединения *Open Software Foundation* (*OSF*). OSF финансировалась компаниями-основателями. Основной задачей организации стала разработка операционной системы, пользовательской и распределенной вычислительной среды, не зависящей от ограничений, накладываемых лицензионными соглашениями AT&T. OSF распространила среди своих членов *Request for Technology* (Запрос на технологии, RFT) и затем выбрала из полученных предложений самые лучшие независимо от того, на кого из производителей работал их автор.

В ответ корпорации AT&T и Sun совместно с другими производителями систем, основанных на System V, в срочном порядке основали свою органи-

зацию, названную *UNIX International* (*UI*). Ее основной целью было продвижение системы SVR4 на рынке, а также выбор дальнейшего направления развития UNIX System V. В 1990 году организация UI выпустила труд под названием *UNIX System V Road Map*, в котором были выделены основные направления будущего развития системы UNIX.

В 1989 году OSF представила графический пользовательский интерфейс *Motif*, положительно встреченный многими пользователями. Позже организация выпустила первую версию своей операционной системы под названием *OSF/1*. Первая версия OSF/1 базировалась на Mach 2.5, имела совместимость с 4.3BSD и обладала некоторыми возможностями IBM AIX. Представленная система имела множество дополнительных возможностей, не поддерживаемых в SVR4, таких как полная поддержка многопроцессорных систем, динамическая загрузка и монтирование томов. В планах членов организации UI была дальнейшая разработка коммерческих операционных систем, базирующихся на OSF/1.

Объединения OSF и UI начинали с весьма высоких целей, но все равно очень быстро столкнулись с общими, не зависящими от них проблемами. Экономический спад начала 90-х, экспансия Microsoft Windows на рынок операционных систем резко уменьшили рост UNIX-систем. Организация UI ушла из компьютерного бизнеса в 1993 году, а объединение OSF было вынуждено расстаться с большинством амбициозных планов (в том числе и планов по созданию Распределенной среды управления, Distributed Management Environment). Одной из основных систем, основанных на OSF, стала DEC OSF/1, созданная компанией Digital в 1993 году. Позже компания приняла решение удалить из этой системы многие возможности, отличающие ее от своей ОС, и в 1995 году изменила имя системы на Digital UNIX.

1.1.10. SVR4 и ее дальнейшее развитие

В 1989 году вышла первая версия совместно разработанной корпорациями AT&T и Sun системы System V Release 4 (SVR4). Эта система объединила в себе возможности SVR3, 4BSD, SunOS и XENIX. В SVR4 также были добавлены новые функции, такие как изменение состава классов в режиме реального времени, командный интерпретатор *Korn shell* и новые возможности подсистемы `STREAMS`. В следующем году AT&T основала компанию UNIX Systems Laboratories для разработки и продажи систем UNIX.

В 1991 году компания Novell, Inc., создатель сетевой операционной системы NetWare для персональных компьютеров, приобрела часть акций USL и основала совместное предприятие под названием Univel. Целью новой компании стало создание версии SVR4 для настольных систем, интегрированной с ОС NetWare. Такая система была разработана в конце 1992 года и получила название *UnixWare*. После этого было выпущено еще несколько вариантов системы SVR4. Последний вариант, *SVR4.2/ES/MP*, предлагает пользователям расширенную защиту и поддержу многопроцессорных систем.

В 1993 году корпорация AT&T полностью передала USL компании Novell. В следующем году Novell получила права на торговую марку UNIX и подтверждение совместимости своих операционных систем с X/Open. В 1994 году корпорация Sun Microsystems выкупила права на использование кодов SVR4 у Novell, что освободило ее от проблем, связанных с возможным нарушением лицензионных прав и совместимости со стандартами. Система Sun, основанная на SVR4, получила название *Solaris*. Ее последняя версия — это *Solaris 2.5*. Система поддерживает многие дополнительные возможности, такие как собственное многопоточное ядро и поддержка многопроцессорных систем[1].

1.2. Причины изменений системы

Система UNIX сильно преобразилась за годы своего существования. Начав с небольшой операционной среды, использовавшейся группой людей в единственной лаборатории, на сегодняшний день система UNIX стала одной из

[1] ОС Solaris была перенесена на платформу Intel с платформы SPARC. Последние ее версии представляют собой мощные и масштабируемые системы для рабочих станций, младших серверов, корпоративных серверов и суперсерверов. Многие решения Sun не имеют аналогов, технологии компании всегда были «на гребне» (та же Java), а ее системы отличаются высокой надежностью. Но поскольку Sun приходится вести войну на всех фронтах, в первую очередь против Linux и Microsoft, она вынуждена поспевать за конкурентами. В результате производительность Solaris 8 оказывается ниже таковой для Solaris 7, надежность страдает, а проблемам интернационализации (в том числе поддержке русского языка) уделяется недостаточное внимание. За последние 4 года вышли версии для Intel 7 (или Solaris 2.7, SunOS 5.7), 8 и 9 (начало 2003). Solaris 7 работает на одно- и многопроцессорных системах, поддерживает до 4 Гбайт RAM и файловые системы до 1 Тбайт. Файловая система UFS с расширениями от Sun и 64-разрядной адресацией позволяет протоколировать события (по образу журналов). Недостатки низкой производительности NFS сглажены за счет кэширования информации непосредственно на локальном диске (файловая система CacheFS). Solaris 8 Sun противопоставила Windows 2000 Datacenter, а Solaris 9 — Microsoft Windows 2000 Server. Самая главная новация в Solaris 8 — Live Upgrade («горячее» обновление). Она позволяет администраторам устанавливать исправления в ядро и аппаратные средства на серверах SPARC без их перезагрузки. Встроенная кластеризация до 4-х процессоров (для SPARC) позволяет увеличить производительность вебсервера. Solaris 8 поддерживает стандарт IPv6 и спецификацию IPSec (для IPv4), стандарт Mobile IP для мобильных пользователей. Sun объявила о поддержке сервера каталогов LDAP и об отказе от службы NIS, вошедшей в общее пользование с ее же подачи. LDAP интегрирован в Solaris 9, что повышает ее производительность в сравнении с Solaris 8 в 5 раз. В Solaris 8 встроены две графические оболочки — CDE (Common Desktop Environment) и Open Windows, в Solaris 9 — графический интерфейс пользователя Web Start и консоль администрирования Solaris Management Console (SMC). Последняя версия — Solaris 9 — включила в себя множество новых возможностей. Это: протокол сетевой аутентификации Kerberos 5, протокол Secure Shell для безопасного установления связи с UNIX-машинами, поддержка RAID-массивов, расщепление и удвоение дисков, новая организация нитей, повышающая производительность многопроцессорных систем, защита от переполнения буфера — «бич» безопасности для любых систем, основанная на блокировке выполнения стековых операций определенными приложениями. Веб-сервисы, Java-сервер приложений, выполняющий программы е-бизнеса, средства управления предприятиями на основе интернет-технологий WBEM направлены на интеграцию множества компьютеров в один огромный вычислительный комплекс. — *Прим. ред.*

основных операционных систем, предлагаемой в самых различных вариантах многими поставщиками. В настоящее время UNIX используется на самых различных системах, начиная от небольших встроенных контроллеров и заканчивая огромными мэйнфреймами и массивно-параллельными системами. Под управлением UNIX работают самые разнообразные приложения: в офисах UNIX используется как настольная ОС; в финансовых областях с ее помощью обрабатываются крупные базы данных, а научные лаборатории применяют эту систему для сложных математических вычислений.

Изменения и рост UNIX обусловлены, в первую очередь, появлением новых задач, стоявших перед системой. Хотя на данный момент UNIX является целостной операционной системой, это не значит, что она будет оставаться неизменной в дальнейшем. Причиной постоянно происходящих изменений никак нельзя назвать изначально неверный дизайн ОС. Напротив, простота добавления в UNIX новых возможностей по мере развития технологий убедительно доказывает обратное. Не имея точного представления о цели, формах и задачах будущей системы, ее создатели начали работу с построения простых, расширяемых базовых средств, собирая и анализируя предложения по усовершенствованию системы отовсюду, от научных учреждений, коммерческих предприятий и простых пользователей.

Проведем анализ основных факторов, повлиявших на рост и совершенствование системы. В этом разделе вы увидите не только описание самих факторов, но и соображения автора относительно предполагаемых трансформаций системы UNIX в будущем.

1.2.1. Функциональные возможности

Главной причиной, подталкивающей к изменениям, является необходимость добавления новых возможностей в систему. Изначально новые функциональные средства появлялись в UNIX только как пользовательские инструменты и утилиты. Позже, когда UNIX превратилась во вполне развитую систему, разработчики стали добавлять многие дополнительные возможности прямо в ядро ОС.

Многие из новых функций системы создавались для поддержки более сложных программ. Одним из примеров таких нововведений является набор Interprocess Communication (IPC) в ОС System V, в состав которого входят поддержка разделения памяти, семафоры и очереди сообщений. Все эти возможности позволили процессам взаимодействовать, используя совместно данные, обмениваясь сообщениями и синхронизируя свои действия. Большинство современных систем UNIX также имеют несколько уровней поддержки для создания многопоточных приложений.

Возможности IPC и технологии потоков существенно помогают в разработке сложных приложений, например тех, что основаны на модели «клиент-сервер». В таких программах обычно серверная часть находится в режиме

постоянного ожидания запроса от клиентов. Если такой запрос приходит, сервер обрабатывает его и снова переходит в режим ожидания следующего. Если сервер имеет возможность обслуживания сразу несколько клиентских запросов, в этом случае предпочтительно вести их обработку параллельно. С применением технологии IPC сервер может использовать отдельный процесс для обработки каждого запроса, в то время как все выполняемые процессы могут совместно использовать одни и те же данные. Многопоточная система позволяет реализовать сервер как один процесс, имеющий несколько параллельно функционирующих потоков, использующих общее адресное пространство.

Возможно, наиболее «видимой» частью любой операционной системы является файловая система. В ОС UNIX также было добавлено множество новых возможностей, в том числе поддержка файлов *FIFO* (First In, First Out), символьных связей, а также файлов, имеющих размеры большие, чем раздел диска. Современные системы поддерживают защиту файлов, списки прав доступа, а также ограничения доступа к дискам для каждого пользователя.

1.2.2. Сетевая поддержка

За годы развития UNIX максимальным изменениям была подвергнута часть ядра, являющаяся сетевой подсистемой. Ранние версии ОС работали отдельно друг от друга и не имели возможности соединения с другими машинами. Однако распространение компьютерных сетей поставило перед разработчиками проблему необходимости их поддержки в системе UNIX. Первым занялся решением этой проблемы университет Беркли. Организация DARPA профинансировала проект встраивания поддержки TCP/IP в 4BSD. На сегодняшний день системы UNIX поддерживают большое количество сетевых интерфейсов (таких как Ethernet, FDDI и ATM), протоколов (TCP/IP, UDP/IP[1], SNA[2] и других) и средств (например, сокетов и STREAMS).

Появление возможности соединения с другими компьютерами во многих отношениях повлияло на операционную систему. Пользователи ОС сразу же захотели совместно использовать файлы, расположенные на соединенных между собой машинах, а также запускать приложения на удаленных узлах. Для удовлетворения этих требований развитие системы UNIX велось в трех направлениях:

- Разрабатывались распределенные файловые системы, позволявшие вести прозрачный доступ к файлам на удаленных узлах. Наиболее удачными из созданных систем оказались *Network File System* (*NFS*) корпорации Sun Microsystems, *Andrew File System* (*AFS*) университета Карнеги—Меллона и *Distributed File System* (*DFS*) корпорации Transarc.

[1] User Datagram Protocol/Internet Protocol, протокол пользовательских дейтаграмм/протокол Интернета.

[2] System Network Architecture (системная сетевая архитектура) компании IBM.

- Создавалось большое количество распределенных служб, позволяющих совместно использовать информацию по сети. Эти службы представляют собой обычные пользовательские программы, основанные на модели «клиент-сервер» и использующие удаленные вызовы процедур для активации действий на других компьютерах. Примерами таких программ являются *Network Information Service* (Сетевой информационный сервис, NIS) и *Distributed Computing Environment* (Распределенная вычислительная среда, DCE).
- Появлялись распределенные операционные системы, такие как *Mach*, *Chorus* и *Sprite*, имеющие различную степень совместимости с UNIX и продвигаемые на рынок как базовые технологии для построения будущих распределенных ОС.

1.2.3. Производительность

Постоянной движущей силой, заставляющей вносить изменения в системы, является увеличение их производительности. Конкурирующие между собой поставщики операционных систем тратят огромные усилия на демонстрацию того, что именно их ОС производительнее, чем другие. Почти все внутренние подсистемы претерпели большие изменения, выполненные с целью увеличения производительности систем.

В начале 90-х годов университет Беркли представил файловую систему Fast File System, благодаря интеллектуальным правилам размещения блоков на диске увеличивающую производительность системы. Позже появились более быстродействующие файловые системы, использующие внешнее размещение или технологии поддержки журналов. Увеличение вычислительных мощностей также стало основной причиной разработок в области коммуникаций между процессами, работы с памятью и многопоточных процессов. Когда для работы многих приложений оказалось недостаточно одного процессора, производители разработали многопроцессорные системы под управлением UNIX, некоторые из которых имеют сотни процессоров.

1.2.4. Изменение аппаратных платформ

Операционная система должна «шагать в ногу» с современными аппаратными технологиями. Часто это означает необходимость ее переноса на новые и более высокопроизводительные процессоры. После того как ядро UNIX было почти полностью переписано на языке C, решение этой задачи стало относительно легким. В прошлом разработчики тратили огромные усилия на отделение от общего кода программы участков, написанных для конкретного аппаратного оборудования, и включение их в отдельные модули. После проведения этих действий для переноса системы требовалось переписывать заново только отдельные модули. Обычно такие модули отвечали за обработку пре-

рываний, трансляцию виртуальных адресов, переключение контекстов и драйверы устройств.

В большинстве случаев для выживания операционной системы на рынке необходим ее постоянный перенос на вновь появляющееся оборудование. Наиболее очевидной возможностью, которую обязательно желают видеть в UNIX, является поддержка многопроцессорных систем. Ядро традиционного варианта UNIX разрабатывалось для работы на одном процессоре и не имело возможности защиты структур данных при параллельном доступе к ним одновременно нескольких процессоров. Однако позже сразу несколько производителей разработали многопроцессорные реализации системы UNIX. Большинство их них использовали традиционное ядро UNIX и добавляли собственные элементы защиты, обеспечивающие безопасность общих структур данных. Это средство называется параллелизацией. Малая часть производителей занялась построением собственного ядра на иных основах.

При более глубоком рассмотрении проблемы видно, что неравномерность развития различных аппаратных технологий оказала глубокое влияние на разработку операционных систем. С тех времен, как была создана первая версия UNIX для PDP-7, средняя скорость процессоров увеличилась примерно в 100 раз[1]. Объемы памяти и дискового пространства, выделяемого для одного пользователя, возросли более чем в 20 раз. С другой стороны, скорость доступа к памяти и дискам увеличилась всего лишь в 2 раза.

В 70-х годах производительность систем UNIX ограничивалась скоростью работы процессора и размером памяти. Но вскоре в ядре системы стали использоваться такие технологии, как свопинг и, чуть позже, страничная организация памяти (технология, позволявшая выполнять большое количество процессов на малых объемах памяти). По мере развития вычислительной техники скорость доступа к памяти и процессору стала играть меньшую роль, а сама система занималась в большей степени вводом-выводом, основное время занимаясь переносом страниц между дисками и оперативной памятью. Это являлось важной причиной появления новых разработок в области файловых систем, хранения информации и архитектур виртуальной памяти, целью которых была оптимизация дисковых процедур. Именно для этого были разработаны *Redundant Arrays of Inexpensive Disks* (массивы недорогих дисков с избыточностью, RAID) и происходило быстрое распространение структурированных файловых систем.

1.2.5. Улучшение качества

Все преимущества функциональности и скорости работы можно запросто свести на нет, если в систему заложены ошибки. Разработчики вносили мно-

[1] Частота процессора Intel 4004 в первом миникомпьютере (1971 г.) составляла 1 МГц. Соответственно на 2002 год приходится говорить о росте производительности в 1000 раз. — *Прим. ред.*

жество изменений в дизайн систем, чтобы сделать их более устойчивыми, стараясь добиться увеличения надежности программного обеспечения.

Изначальный механизм оповещения был ненадежен и неэффективен по многим причинам. Однако позже его реализация была пересмотрена (сначала разработчиками из Беркли, затем — AT&T), и в результате появилась новая, более устойчивая система оповещения, получившая название *надежных сигналов*.

Система BSD, точно так же, как и System V, не была застрахована от возможных отказов. В системах UNIX перед записью на диск данные хранятся некоторое время в памяти. Следовательно, существует потенциальная возможность их потери в случае отказа, а также нарушения целостности файловой системы. В различных вариантах UNIX предлагается стандартная утилита fsck(8), проверяющая и восстанавливающая поврежденные файловые системы. Эта операция занимает ощутимое количество времени и может длиться десятки минут на крупных серверах, имеющих диски больших объемов. Многие современные системы UNIX поддерживают файловые системы, использующие технологию поддержки журналов, что увеличивает доступность и стабильность системы и устраняет потребность в применении fsck.

1.2.6. Глобальные изменения

За последние три десятилетия произошли огромные изменения в принципах использования компьютеров. В 70-х годах вычислительная система представляла большой централизованный компьютер размером с комнату, поддерживающий работу пользователей, которые подключались к нему при помощи терминалов. Применялись системы разделения времени, в которых центральный компьютер разделял процессорные ресурсы среди пользователей. Терминалы представляли собой простые устройства, умеющие чуть больше, чем вывод данных в текстовом режиме.

В 80-х годах началась эра *рабочих станций*, оборудованных высокоскоростными графическими дисплеями, имеющими возможность вывода информации в нескольких окнах, в каждом из которых выполняется оболочка UNIX. Рабочие станции идеальны для интерактивного использования и имеют достаточную вычислительную мощность для работы с обычными пользовательскими приложениями. На рабочих станциях обычно в один момент времени работает один пользователь, но они могут поддерживать многих пользователей. Появление высокоскоростных сетей позволило рабочим станциям соединяться между собой, а также с другими компьютерами.

Позже появилась новая модель вычислений, называемая *клиент-серверными вычислениями*, в которой один или более мощных централизованных компьютеров, называемых *серверами*, предоставляют различные службы индивидуальным рабочим станциям, или *клиентам*. Для хранения файлов пользователей стали применяться *файловые серверы*. *Серверы приложений* — это

компьютеры, оснащенные одним или несколькими мощными процессорами, на которых пользователи могут выполнять задачи, требующие большого объема вычислений (например, решение математических уравнений). На *серверах баз данных* выполняется специальная программа, обрабатывающая запросы к базам данных, поступающие от клиентов. Обычно серверы представляют собой мощные высокопроизводительные машины с быстродействующими процессорами и большими объемами оперативной и дисковой памяти. Клиентские рабочие станции имеют меньшую производительность, размеры памяти и объемы дисков, но они, как правило, оснащаются высококачественными мониторами и предоставляют пользователю большие интерактивные возможности.

Постепенно рабочие станции становились все более производительными, различия между клиентами и серверами все больше стирались. Более того, централизованное выполнение необходимых служб на небольшом количестве серверов приводило к перегрузкам сетей и самих серверов. Результатом стало изменение подхода к построению компьютерных систем и появление новой технологии *распределенных вычислений*. При использовании этой модели компьютеры совместно предоставляют какую-либо сетевую службу. Каждый узел сети может иметь собственную файловую систему и предоставлять доступ к ней для других узлов. В результате узел функционирует как сервер по отношению к локальным файлам и как клиент по отношению к файлам, хранящимся на других узлах. Такой подход уменьшает перегрузки сети, а также количество сбоев в ее работе.

Система UNIX была адаптирована для различных моделей вычислений. Например, ранние варианты системы поддерживали только локальную файловую систему. Поддержка сетевых протоколов привела к разработке распределенных файловых систем. Некоторые из них, такие как AFS, требовали специализированных серверов. Позже файловые системы были переработаны и стали распределенными, что дало возможность каждому компьютеру в сети выступать в роли клиента и сервера.

1.2.7. Поддержка различных приложений

Система UNIX изначально разрабатывалась для применения в простых средах разделения времени, таких как исследовательские лаборатории или университеты. Системы позволяли некоторому количеству пользователей выполнять несложные программы обработки и редактирования текстов и математических вычислений. После того как UNIX обрела популярность, ею стали пользоваться для более широкого спектра приложений. К началу 90-х годов UNIX использовалась в физических и космических лабораториях, мультимедийных рабочих станциях для обработки звука и видео, а также во встроенных контроллерах, использующихся в критически важных системах.

Каждое приложение обычно имеет различные требования к системе, что подхлестнуло разработчиков систем изменять их в соответствии с этими требованиями. Мультимедийные и встроенные приложения требуют гарантий доступности ресурсов и определенной скорости отклика на запросы. Научные приложения требуют одновременной работы нескольких процессоров. Для реализации этих требований в некоторых системах UNIX появились возможности работы в режиме реального времени, такие как процессы с фиксированным приоритетом, разграничение использования процессоров и возможность сохранения данных в памяти.

1.2.8. Чем меньше, тем лучше

Одним из преимуществ оригинального дизайна системы UNIX был ее небольшой размер, простота и ограниченный набор основных функций. Изначальным подходом к системам было предоставление простых инструментов, которые можно гибко комбинировать между собой, используя такие инструменты, как конвейер (pipe). Однако ядро традиционного варианта системы было достаточно монолитным, следовательно, его расширение представляло собой непростую задачу. Чем больше функций добавлялось в ядро, тем оно все больше разрасталось и усложнялось, все дальше уходя от его первоначального размера, не превышающего 100 Кбайт, постепенно достигнув объема в несколько мегабайтов. Объемы памяти компьютеров стали увеличиваться, вместе с тем поставщики систем и их пользователи игнорировали этот факт. Но именно он сделал UNIX менее пригодной для использования на небольших персональных компьютерах и портативных системах.

Многие стали понимать, что такое изменение системы не предвещает ничего хорошего, так как она становится слишком большой, перенасыщенной и неорганизованной. Были потрачены большие усилия на переработку системы или на написание нового варианта, который основывался бы на оригинальной философии UNIX, но имел бы большую расширяемость и модульность. Наиболее удачной реализацией такой системы стала Mach, которая выступила основой для последующих коммерческих ОС, примерами которых являются OSF/1 и NextStep. Система Mach впоследствии использовала архитектуру микроядра (см. раздел 1.1.7), в которой небольшое ядро предоставляет средство для выполнения программ, а серверные задачи на пользовательской уровне предоставляют все остальные функции.

Не все попытки контроля размера ядра имели успех. К сожалению, микроядра не могли иметь производительность, сравнимую со скоростью работы традиционного монолитного ядра, в первую очередь, по причине издержек, накладываемых передачей сообщений. Менее амбициозные проекты, такие как модульность, ядра со страничной поддержкой и динамическая загрузка, имели больший успех, так как позволяли загружать или выгружать из памяти компоненты ядра по мере надобности.

1.2.9. Гибкость системы

В 70-х и начале 80-х годов ядро систем UNIX было недостаточно гибким. Ядро поддерживало только один тип файловой системы, набор алгоритмов планирования, а также формат выполняемых файлов (рис. 1.1). ОС имела некоторую степень гибкости только по отношению к переключателям символьных и блочных устройств, позволяя различным типам устройств иметь доступ к системе через общий интерфейс. Развитие распределенных систем в середине 80-х годов стало очевидной причиной для того, чтобы UNIX стала поддерживать как удаленные, так и локальные файловые системы. А такие возможности, как разделяемые библиотеки, потребовали поддержки различных форматов выполняемых файлов. Система UNIX стала поддерживать эти форматы, оставив традиционный `a.out` для совместимости. Одновременное сосуществование мультимедийных приложений и программ, функционирующих в режиме реального времени, потребовало поддержки планирования для различных типов приложений.

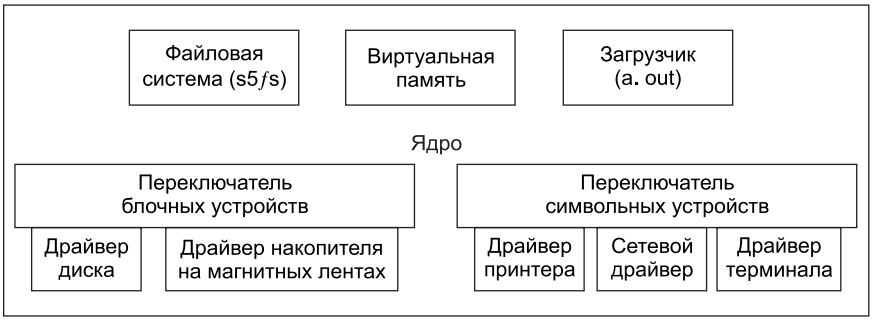

Рис. 1.1. Традиционное ядро UNIX

В итоге UNIX превращалась в более гибкую операционную систему, которая бы поддерживала несколько различных методов выполнения одной и той же задачи. Это потребовало разработки многих более гибких структур, таких как интерфейс vnode/vfs, системный вызов `exec`, планирование процессов, а также сегментную архитектуру памяти. Современное ядро UNIX похоже на систему, показанную на рис. 1.2. В каждом из внешних кругов представлен интерфейс, который может быть реализован различными способами.

1.3. Оглянемся назад, посмотрим вперед

UNIX заметно изменилась после своего первого представления. Несмотря на достаточно скромный «выход в свет» она завоевала большую популярность. Во второй половине 80-х годов операционную систему UNIX выбрали многие коммерческие организации, учебные заведения и научно-исследователь-

ские лаборатории. Чаще всего результатом приобретения того или иного варианта системы был осознанный выбор. Позже позиции UNIX были потеснены корпорацией Microsoft, предлагающей операционные системы семейства Windows. Постепенно ОС UNIX стала проигрывать в битве за рынок настольных систем. В этом разделе мы проведем анализ причин ее успеха и популярности, а также факты, не позволившие UNIX покорить мир.

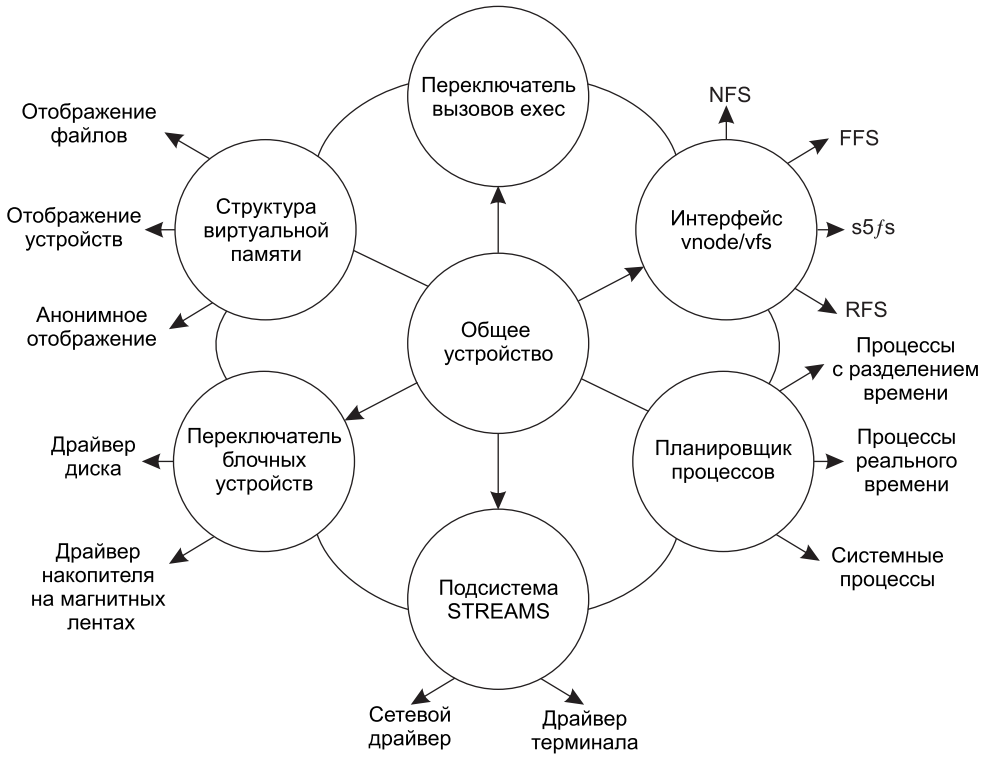

Рис. 1.2. Современное ядро UNIX

1.3.1. Преимущества UNIX

UNIX завоевала такую высокую степень популярности, на которую, возможно, даже не рассчитывали ее разработчики. Одной из главных причин успеха являлся способ распространения системы. Корпорация AT&T, ограниченная законами в своих возможностях, продавала лицензии и исходные коды системы по достаточно низкой цене, поэтому UNIX стала популярной среди многих пользователей по всему миру. Так как в комплект поставки входили и исходные коды, пользователи имели возможность экспериментировать с ними, улучшать их, а также обмениваться друг с другом созданными изменениями. Корпорация AT&T встраивала многие из новшеств в следующие версии системы.

Разработчики из Беркли также поддерживали эту традицию. Развитие системы UNIX оказалось очень открытым процессом. Добавления поступали из учебных заведений, коммерческих организаций и от хакеров-энтузиастов из разных континентов и стран мира. Даже после коммерциализации UNIX многие производители ОС поддержали концепцию открытых систем и сделали свои разработки доступными для других, создавая спецификации открытых систем, такие как NFS.

Оригинальная версия UNIX имела хороший дизайн, который являлся базисом последующего успеха более поздних реализаций и вариантов системы. Одну из сильнейших сторон системы можно охарактеризовать выражением «красота в краткости» [12]. Ядро небольшого размера имело минимальный набор основных служб. Утилиты небольшого размера производили малый объем манипуляций с данными. Механизм *конвейера* совместно с программируемой оболочкой позволял пользователям комбинировать эти утилиты различными способами, создавая мощные, производительные инструменты.

Файловая система UNIX является примером вышеописанного подхода. В отличие от других операционных систем, обладающих сложными методами доступа к файлам, такими как *Indexed Sequential Access Method* (Индексированный последовательный доступ к файлам, ISAM) или *Hierarchical Sequential Access Method* (Иерархический последовательный доступ к файлам, ISAP), система UNIX интерпретирует файлы как простую последовательность байтов. Приложения могут записывать в содержимое файлов любую структуру и использовать различные собственные методы доступа, не ставя об этом в известность операционную систему.

Многие системные приложения используют для представления своих данных символьные строки. К примеру, важнейшие базы данных, такие как `/etc/passwd`, `/etc/fstab` и `/etc/ttys`, являются обычными текстовыми файлами. Возможно, что структурированное хранение информации в двоичном формате представляется более эффективным, однако представление в виде текста позволило пользователям легко просматривать и производить манипуляции с этими файлами без применения специальных инструментов. Текст является общей, универсальной и обладающей высокой степенью переносимости формой данных, которые можно легко обрабатывать множеством различных утилит.

Еще одной особенностью системы UNIX стал простой унифицированный интерфейс с устройствами ввода-вывода. Представляя все устройства как файлы, система позволяет пользователям применять один и тот же набор команд и системных вызовов для доступа и работы с различными устройствами, равно как и для работы с файлами. Разработчики могут создавать программы, производящие ввод-вывод, не заботясь, с чем именно производится обмен, с файлом, терминалом пользователя, принтером или другим устройством. Таким образом, используемый совместно с перенаправлением данных интерфейс ввода-вывода системы UNIX — простой и одновременно мощный.

Причиной успеха и распространения UNIX стала ее высокая степень переносимости. Большая часть ядра системы написана на языке C. Это по-

зволило относительно легко адаптировать UNIX к новым аппаратным платформам. Первая реализация системы появилась на популярной тогда машине PDP-11 и затем была перенесена на VAX-11, имевшую не меньшую популярность. Многие производители «железа» после создания новых компьютеров имеют возможность просто перенести на них уже имеющуюся систему UNIX вместо того, чтобы создавать для своих разработок операционную систему заново.

1.3.2. Недостатки UNIX

Известно, что любая медаль имеет две стороны. Рассказав о преимуществах системы UNIX, необходимо привести и ее недостатки. Один из наиболее объективных обзоров UNIX был создан не кем иным, как Денисом Ритчи. В январе 1987 года на конференции USENIX в разделе «Ретроспективы UNIX» Ритчи сделал доклад, где провел анализ многих недостатков системы [13], которые кратко описаны ниже.

Хотя UNIX изначально была весьма простой системой, это вскоре закончилось. Например, AT&T добавила в стандартную библиотеку ввода-вывода буферизацию данных, что повысило ее эффективность и сделало программы переносимыми на не-UNIX-системы. Библиотека разрасталась и стала более сложной, чем системный вызов, лежащий в ее основе. Системные вызовы `read` и `write` были неделимыми операциями над файлами, в то время как буферизация библиотеки ввода-вывода уничтожила эту цельность.

UNIX сама по себе являлась замечательной операционной системой, однако большинству пользователей нужна была не сама система, а, в первую очередь, возможность выполнять определенное приложение. Пользователей не интересовала элегантность структуры файловой системы или модели вычислений. Они хотели работать с определенными программами (например, текстовыми редакторами, финансовыми пакетами или программами для создания изображений), потратив на это минимум расходов и усилий. Недостатки простого унифицированного (обычно графического) пользовательского интерфейса в первых системах UNIX были основной причиной его неприятия массами. Как сказал Ритчи: «UNIX является простой и понятной системой, но чтобы понять и принять ее простоту, требуется гений (или, как минимум, программист)».

Получилось так, что элегантность и эстетичность, свойственная UNIX, требует от пользователей, желающих эффективно работать в системе, творческого мышления и определенной изобретательности. Однако большинство пользователей предпочитают простые в изучении интегрированные многофункциональные программы, подобные тем, что применяются на персональных компьютерах.

В какой-то степени система UNIX явилась жертвой своего собственного успеха. Простота лицензионных условий и переноса на различные аппарат-

ные платформы стала причиной неконтролируемого роста и лавинообразного распространения различных реализаций ОС. Каждый пользователь имел право вносить свои собственные изменения в систему, в результате группы разработчиков часто создавали несовместимые между собой варианты. Изначально существовали две основные ветви развития UNIX, разрабатываемые компаниями AT&T и BSD. Каждая реализация имела оригинальную файловую систему, архитектуру памяти, сигналы и принципы работы с терминалами. Позже другие поставщики предложили новые варианты UNIX, стараясь привести их к некоторой степени совместимости с реализациями AT&T и BSD. Однако чем дальше, тем все менее предсказуемой становилась ситуация, а разработчикам приложений требовалось все больше усилий, чтобы приспособить свои программы ко всем различным вариантам UNIX.

Стандартизация систем стала лишь частичным решением проблемы, так как встретила определенное сопротивление. Поставщики стремились добавить в свои разработки какие-либо уникальные функции, стараясь создать продукт, имеющий отличия от остальных, и тем самых показать его преимущества среди конкурирующих вариантов.

Ричард Рашид (Richard Rashid), один из разработчиков системы Mach, предложил свою версию причин неудач UNIX. Во вступлении к курсу лекций о системе Mach (Mach Lecture Series [16]) Рашид сказал, что причиной создания ОС Mach стало наблюдение за эволюционированием системы UNIX, которая имела минимальные возможности для построения инструментов пользователя. Большие сложные инструменты создавались путем комбинирования множества простых функций. Однако такой подход не был перенесен на ядро системы.

Традиционное ядро UNIX было недостаточно гибким и расширяемым, оно имело минимальные возможности для дополнительного использования кода. Позже разработчики стали просто добавлять новые коды в ядро системы, делая его основой для новых функциональных средств. Ядро очень быстро стало раздутым, сложным и абсолютно немодульным. Разработчики Mach попытались решить эти проблемы, переписав систему заново с нуля, взяв за основу небольшое количество основных функций. В современных реализациях UNIX вышеописанная система решается различными способами, например добавляются гибкие структуры к подсистемам, как это было описано в разделе 1.2.9.

1.4. Границы повествования книги

Эта книга описывает современные системы UNIX. С целью полноты описания и передачи определенного исторического контекста приводится краткий рассказ о возможностях ранних реализаций ОС. В настоящее время существует множество вариантов системы UNIX, каждая из которых по-своему уникальна. Мы постарались разделить все системы на два типа: базовые и коммерческие

варианты. Базовые ОС включают в себя System V, 4BSD и Mach. Другие варианты происходят от одной из базовых ОС и содержат различные дополнительные возможности и расширения, созданные их разработчиками. Список таких систем включает в себя SunOS и Solaris 2.x корпорации Sun Microsystems, AIX компании IBM, HP-UX от Hewlett-Packard, а также ULTRIX и Digital UNIX от корпорации Digital.

Эта книга не специализируется на каком-то специфическом варианте или реализации системы UNIX. Вместо этого здесь проводится анализ важнейших разработок, их архитектуры и подхода к решению многих проблем. Книга уделяет внимание прежде всего системе SVR4, но вы можете найти здесь достаточно подробный рассказ о 4.3BSD, 4.4BSD и Mach. При рассказе о коммерческих вариантах ОС максимальное внимание уделяется SunOS и Solaris 2.x. Причиной повышенного акцента является не только успех систем компании Sun Microsystems на рынке ОС, но прежде всего то, что именно эта компания была разработчиком многих технических решений, интегрированных в базовые варианты систем, а также то, что компания создала большое количество книг по своим ОС.

В тексте книги часто приводятся общие ссылки на традиционный или коммерческий варианты системы UNIX. Под традиционными вариантами мы имеем в виду SVR3, 4.3BSD и их более ранние реализации. Часто в тексте книги встречаются фразы, касающиеся каких-либо возможностей традиционных систем (например, «в традиционных системах UNIX поддерживался один тип файловой системы»). Несмотря на то что между SVR3 и 4.3BSD имеются различия в каждой подсистеме, между ними есть и много общего. Общие фразы типа приведенной выше обращают внимание как раз на такие свойства систем. При рассмотрении современных ОС UNIX мы подразумеваем системы SVR4, 4.4BSD, Mach, а также реализации, основанные на них. Таким образом, общий комментарий, наподобие «современные системы UNIX поддерживают какую-либо реализацию журнальной файловой системы», означает, что такая возможность имеется во многих современных системах, но необязательно во всех из них.

1.5. Дополнительная литература

1. Accetta, M., Baron, R., Golub, D., Rashid, R., Tevanian, A., and Young, M, «Mach: A New Kernel Foundation for UNIX Development», Proceedings of the Summer 1986 USENIX Technical Conference, Jun. 1986, pp. 93–112[1].

[1] Все публикации прошлых лет в рамках конференции USENIX доступны на сайте http://www.usenix.org в формате PDF (кроме последних нескольких месяцев). Там же размещена полная библиография и имеется возможность поиска по автору, дате и ключевым словам. — *Прим. ред.*

2. Allman, E. «UNIX: The Data Forms», Proceedings of the Winter 1987 USENIX Technical Conference, Jan. 1987, pp. 9–15.
3. American Telephone and Telegraph, «The System V Interface Definition (SV1D)», Third Edition, 1989.
4. Bostic, K., «4.4BSD Release», Vol. 18, No. 5, Sep.–Oct. 1993, pp. 29–31.
5. Bourne, S., «The UNIX Shell», The Bell System Technical Journal, Vol. 57, No. 6, Part 2, Jul.–Aug. 1978, pp. 1971–1990.
6. Gerber, C. «USL Vs. Berkeley», UNIX Review, Vol. 10, No. 11, Nov. 1992, pp. 33–36.
7. Institute for Electrical and Electronic Engineers, Information Technology, «Portable Operating System Interface (POSIX) Part 1: System Application Program Interface (API) [C language]», 1003.1–1990, IEEE, Dec. 1990.
8. Joy, W. N., Fabry, R. S., Leffler, S. J., McKusick, M. K., and Karels, M. J., «An Introduction to the C Shell», UNIX User's Supplementary Documents, 4.3 Berkeley.
9. «Software Distribution», Virtual VAX-11 Version, USENIX Association, 1986, pp. 41–46.
10. Organick, E. J., «The Multics System: An Examination of Its Structure», The MIT Press, Cambridge, MA, 1972.
11. Rashid, R. F., «Mach: Technical Innovations, Key Ideas, Status», Mach 2.5 Lecture Series, OSF Research Institute, 1989.
12. Richards, M., and Whitby-Strevens, C., «BCPL: The Language and Its Compiler», Cambridge University Press, Cambridge, UK, 1982.
13. Ritchie, D. M., and Thompson, K., «The UNIX Time-Sharing System», The Bell System Technical Journal, Vol. 57, No. 6, Part 2, pp. 1905–1930, Jul.–Aug. 1978.
14. Ritchie, D. M., «Unix: A Dialectic», Proceedings of the Winter 1987 USENIX Technical Conference, Jan. 1987, pp. 29–34.
15. Salus, P. H., «A Quarter Century of UNIX», Addison-Wesley, Reading, MA, 1994.
16. Thompson, K., and Ritchie, D. M., «The UNIX Time-Sharing System», Communications of the ACM, Vol. 17, No. 7, M. 1974, pp. 365–375.
17. «The X/OPEN Portability Guide (XPG)», Issue 4, Prentice-Hall, Englewood Cliffs, NJ, 1993.