

# Периодическая таблица дистрибутивов Linux

(С) В.А.Костромин, январь 2005 г.

*Эта статья - смелая попытка рассортировать многочисленные дистрибутивы Linux по "атомным весам", установить их основные характеристики и классифицировать по базовым признакам. Попутно читатель может ознакомиться с историей Linux и даже узнать имя изобретателя первого дистрибутива.*

В.А.Костромин (kos AT rus-linux.net)

## 1. Что такое дистрибутив Linux?

Операционная система Linux состоит из многочисленных компонентов, важнейшим из которых является ядро (kernel), сообщество разработчиков которого возглавляет Линус Торвальдс. Но операционная система состоит не только из ядра. Для ее работы необходима еще масса других программных средств: драйверы аппаратных устройств, утилиты управления файловой системой, программы для организации взаимодействия с пользователем и так далее. В отличие от других типов операционных систем (например, Windows, Solaris или HP-UX) отдельные компоненты операционной системы разрабатываются и поддерживаются не какой-то одной фирмой, а независимыми группами разработчиков, которые работают на принципах Open Source и отдают разработанные ими продукты в общественное пользование на условиях Стандартной Общественной Лицензии (General Public License - GPL). Уже к моменту появления ядра Linux значительная часть программных компонент, необходимых для запуска системы, была разработана в рамках проекта GNU, что, собственно, и позволило Торвальдсу в достаточно сжатые сроки создать ОС, которая получила его имя. Можно еще добавить, что кроме операционной системы, для работы пользователя необходимы различные прикладные программы. В случае Linux эти программы в большинстве своем тоже разрабатываются группами энтузиастов "на общественных началах" и распространяются под лицензией GPL.

Поскольку все компоненты Linux-систем распространяются на условиях GPL, может сложиться впечатление, что любой человек может собрать коллекцию свободного ПО и установить Linux на свой компьютер. И какая-то степень правдоподобия в таком утверждении есть. Однако тот, кто задумал осуществить такой проект, должен по крайней мере представлять, какие исполняемые файлы и библиотеки необходимы для того, чтобы успешно запустить систему, а также знать, где должны размещаться системные файлы, как организовать загрузку системы и как правильно ее сконфигурировать. Кроме того, необходимо обеспечить разрешение взаимозависимостей и противоречий между разными пакетами (и версиями пакетов), что является довольно нетривиальной задачей. Самые первые версии Linux, появившиеся в 1991 году, состояли из двух дискет. Первая дискета была загрузочной и содержала ядро, а вторая - корневую файловую систему и основные утилиты, разработанные в рамках проекта GNU. Копии этих дискет можно было загрузить с сервера университета в Хельсинки. Конфигурирование и настройка системы производились вручную и были очень сложными. Поэтому до появления первых дистрибутивов установить Линукс на свой компьютер мог только достаточно подготовленный специалист, можно сказать эксперт в UNIX.

Положение изменилось после появления так называемых дистрибутивов. Дистрибутив Linux отличается от простого набора пакетов программного обеспечения в первую очередь наличием программы инсталляции, которая позволяет рядовому пользователю установить систему на свой компьютер, причем сделать это не привлекая высококвалифицированного эксперта. Если попытаться дать формальное определение дистрибутива, то получится что-то вроде:

**Дистрибутив Linux** – это набор пакетов программного обеспечения, включающий базовые компоненты операционной систем (в том числе, ядро Linux), некоторую совокупность программных приложений и программу инсталляции, которая позволяет установить на компьютер пользователя операционную систему GNU/Linux и набор прикладных программ, необходимых для конкретного применения системы.

Первые дистрибутивы Linux появились вскоре после того, как Линус Торвальдс выпустил разработанное им ядро Linux под лицензией GPL. Поскольку значительная часть другого необходимого программного обеспечения уже была разработана в рамках проекта GNU, отдельные программисты (и группы программистов) начали разрабатывать как программы инсталляции, так и другие прикладные программы, пользовательский интерфейс, программы управления пакетами и выпускать свои дистрибутивы Linux.

Первый дистрибутив Linux был создан Оуэном Ле Бланк (Owen Le Blanc) в Манчестерском компьютерном центре (Manchester Computing Centre, MCC) в Англии. Первый релиз этого дистрибутива, получившего имя MCC Interim Linux, стал доступен для всех желающих с ftp-сервера Манчестерского университета в феврале 1992 г.

Примерно в то же время сотрудниками университета Техаса был создан дистрибутив TAMU.

В октябре 1992 появился разработанный Питером Мак-Дональдом (Peter McDonald) дистрибутив Softlanding Linux System (SLS), который был первым дистрибутивом, включающим в себя такие элементы, как X Window System и поддержка TCP/IP.

Ни один из этих дистрибутивов не имел хорошей поддержки. В конце 1992 года Патрик Фолькердинг (Patrick Volkerding) выпустил дистрибутив, в значительной части основанный на SLS, который он назвал "Slackware" и который является старейшим дистрибутивом из тех, которые до сих пор активно развиваются. На основе дистрибутива Slackware германской фирмой S.U.S.E (акроним от немецкого "Software- und System Entwicklung), основанной в 1992 году как консультативная группа по ОС UNIX, был создан дистрибутив SuSE Linux, версия 1.0 которого вышла в 1994 году. Позже SuSE интегрировал дистрибутив Jurix Флориана Ла Роше (Florian La Roche).

Еще один проект по разработке дистрибутива, Debian, был начат Яном Мёрдоком (Ian Murdock) 16 августа 1993 года как альтернатива коммерческим дистрибутивам Linux. Ян хотел создать систему, распространяемую абсолютно свободно и открыто, в духе Linux и GNU. Позже разработка Debian была профинансирована проектом GNU: Free Software Foundation, который выделил деньги на один год, с ноября 1994 по ноябрь 1995, что позволило Я.Мердоку в течение этого периода уделять проекту Debian все свое время.

Дистрибутив Red Hat, который включал в себя некоторые аспекты дистрибутива Bogus (например, механизм пакетов), был основан в 1993 году. На основе Red Hat было создано множество других дистрибутивов, в том числе многие коммерческие дистрибутивы, например, Caldera, Mandrake и TurboLinux.

С тех пор число дистрибутивов постоянно растет, возможно, в силу той относительной легкости, с которой дистрибутив может быть создан из отдельных пакетов, поставляемых независимыми разработчиками. По состоянию на 14 января 2005 года сайт DistroWatch.com (на котором ведется учет разных дистрибутивов) насчитывал 373 дистрибутива. Поддержка некоторых из них уже прекращена, но все же еще более 300 разработок были "живы". Только за 2004 год появилось более сотни новых дистрибутивов. И это еще не конец, потому что чуть ли не ежедневно появляются новые и новые дистрибутивы!

Как же сориентироваться в этой массе дистрибутивов, чем отличаются разные дистрибутивы, по каким критериям можно их как-то классифицировать? И как выбрать тот вариант системы, который более всего подходит для конкретной ситуации?

## 2. Критерии классификации дистрибутивов

Приведенные выше цифры количества дистрибутивов, конечно, впечатляют, но давние пользователи Linux знают, что ядро и большинство программ одинаковы во всех дистрибутивах, и что состав пакетов, входящих в разные дистрибутивы, различается в основном версиями или сборками от конкретного производителя. Кроме того, пользователь может добавить в свою систему практически любой необходимый ему пакет, в крайнем случае скомпилировав его из исходных кодов или разработав самостоятельно (при наличии соответствующей квалификации, разумеется). Так что, с одной стороны, практически не существует двух одинаковых экземпляров операционной системы Linux, а с другой стороны, необходимую для конкретного применения конфигурацию системы можно получить, взяв за основу любой из существующих дистрибутивов. Однако трудоемкость настройки системы на конкретное применение все же существенно зависит от того, какой дистрибутив взят за основу.

Поскольку число дистрибутивов Линукс очень велико, ознакомиться на практике с каждым дистрибутивом, чтобы сделать обоснованный выбор, уже не представляется возможным. Следовательно, актуальной становится проблема какой-то классификации дистрибутивов, выделения существенных характеристик, которые могут служить критериями выбора дистрибутива. Достаточно много материалов, дающих хотя бы частичный ответ на поставленные вопросы, можно найти на уже упоминавшемся сайте DistroWatch.com ([\[1\]](http://distrowatch.com)). Эти материалы и легли в основу настоящей статьи, в которой будет предпринята попытка классификации дистрибутивов Линукс по нескольким критериям.

Признаков, по которым различаются отдельные дистрибутивы существует очень много. Вот только некоторые из них:

- ориентация (или назначение) дистрибутива на конкретную область применений (*например, для организации файервола, для работы в качестве маршрутизатора, для создания особо защищенной системы или для использования на домашнем компьютере обычного пользователя с упором на мультимедийные приложения*),
- требования к аппаратному обеспечению (*некоторые дистрибутивы ориентированы (оптимизированы) на компьютеры класса Pentium, другие устанавливаются и на 486-е; SUSE LiveCD 9.2, например, не запускается на компьютере у которого объем ОЗУ меньше 256 МБайт*),
- используемая графическая оболочка (*например, KDE, GNOME или XFce*),

- наличие средств локализации, обеспечивающих поддержку необходимых вам языков (*например, в некоторых LiveCD дистрибутивах не предусмотрена поддержка русского языка, так что русскоязычным пользователям придется приложить дополнительные усилия для русификации системы*),
- носитель, с которого может запускаться система (*например, одна или несколько дискет, CD, только установка на жесткий диск*),
- организация процедуры начальной загрузки системы (*BSD или System V*),
- используемая система управления пакетами (*например, dpkg в Debian, RPM в Fedora Core*),
- структура каталогов файловой системы (см. [\[5.6\]](#)),
- родословная или история происхождения дистрибутива (*Новые дистрибутивы создаются, в основном, не на пустом месте, а на основе одного из уже существующих дистрибутивов.*)
- состав базового устанавливаемого ПО (см. [\[7\]](#)),
- доступность дополнительных пакетов,
- наличие и состав коммерческих программ, включенных в дистрибутив,
- процедура определения аппаратуры,
- предоставляемые инструменты управления системой,
- и так далее ...

Давайте попытаемся выделить из этого списка те признаки, которые могут служить основой для какой-то классификации дистрибутивов. Начнем с того, что отбросим некоторые критерии, не имеющие практического значения.

### 3. Признаки несущественные

Некоторые из перечисленных признаков можно исключить без долгих обоснований. Например, доступность дополнительных пакетов практически одинакова для всех дистрибутивов, поскольку любую программу можно скомпилировать из исходных кодов или же создать пакет в специфичном для данного дистрибутива формате. Точно по тем же основаниям можно исключить признак наличия в составе дистрибутива коммерческого ПО, так как никто не может запретить вам купить такое ПО и добавить его в свою систему. Средства локализации включаются сейчас во все дистрибутивы без исключения, разве что не всегда дистрибутив содержит нужные шрифты. Этот признак становится еще более несущественным в связи с наметившимся переходом на Unicode. Еще несколько признаков давайте рассмотрим чуть подробнее.

#### Структура файловой системы

А.Федорчук в статьях [\[5.6\]](#) предлагал использовать в качестве одного из критериев классификации дистрибутивов структуру каталогов файловой системы. Вероятно, это отличие было существенно в то время, когда была написана статья [5]. В последнее же время набирает ход процесс разработки стандартов для Линукс. В рамках организации [Linux Standard Base](#), ставящей своей целью наладить взаимодействие между разными дистрибутивами и разработать общие стандарты на те или иные компоненты системы (а точнее, даже чуть раньше возникновения этой организации), был разработан стандарт на структуру каталогов файловой системы ([Filesystem Hierarchy Standard](#) - FHS). Следование требованиям стандарта существенно упрощает жизнь создателям программных приложений, потому что обеспечивает нахождение необходимых

для приложения компонент программного обеспечения (в частности, библиотек) в известных и строго определенных местах. Поэтому структура каталогов приняла примерно одинаковый вид во всех основных дистрибутивах. Дистрибутив SuSE, например, когда-то упрекали за неполное соответствие стандарту FHS, но в версии 9.2 он уже полностью соответствует версии 2.3 этого стандарта. Даже если фактическая структура каталогов отличается от стандартной, добиться соответствия можно путем использования символических ссылок. Так что выделить по признаку различия структуры файловой системы какие-то четко очерченные группы дистрибутивов не представляется возможным.

### **Используемая графическая оболочка**

Внешний вид экранов компьютеров, на котором запущены различные оконные менеджеры и графические оболочки может различаться очень существенно. Тем не менее, вряд ли стоит используемый менеджер дисплея или графическую среду (KDE vs GNOME) брать за основу какой-то классификации дистрибутивов. Чаще всего дистрибутив изначально предусматривает возможность выбора между несколькими графическими оболочками. А, кроме того, такие оболочки есть не что иное как один из пакетов программного обеспечения и пользователь всегда имеет возможность скачать и установить такой пакет, даже если он первоначально и не входил в выбранный дистрибутив.

### **Родословная дистрибутива**

Большая часть современных дистрибутивов ведет свою родословную либо от Red Hat, либо от Debian. Но происхождение дистрибутива, то есть выяснение того, от какого из ранее существовавших дистрибутивов он отпочковался, представляет, на мой взгляд, чисто исторический интерес. Любой новый дистрибутив со временем отдаляется от своего прародителя и может мигрировать как в сторону другого дистрибутива, так и стать основателем какой-то совершенно новой ветви в "дистрибутивостроении". Такие дистрибутивы, например, как Mandrake, Conectiva или PLD, произошли, как известно, от Red Hat, но сейчас уже представляют собой вполне самостоятельные разработки. Более того, существуют уже другие дистрибутивы, являющиеся производными от названных. Так что, хотя происхождение и может в какой-то степени характеризовать дистрибутив для опытных пользователей, все же служить критерием классификации оно не может. *Если вас интересует какие дистрибутивы от каких произошли, то на DistroWatch имеется страничка, на которой эти данные приведены.*

### **Процедура определения аппаратуры**

Судя по отзывам в разных источниках, средства определения аппаратного обеспечения компьютера в разных дистрибутивах различаются довольно существенно. Широко известно, что дистрибутив Knoppix отличается в этом смысле в лучшую сторону. Однако, я полагаю, что преимущество это временное. В силу открытости всего программного обеспечения Линукс достижения и наработки одного из разработчиков быстро становятся достоянием всех, а поэтому идеи, найденные Клаусом Кнопเปอร์ом, вскорости будут реализованы и в других дистрибутивах. Именно этот свободный обмен идеями и позволяет быстро развиваться свободному софту вообще.

### **Состав базового устанавливаемого ПО**

Чтобы использовать этот критерий, нужно вначале договориться, что мы имеем в виду, когда говорим "базовое ПО". Это понятие было введено и рассматривалось в статьях А.Федорчука (см. [\[6.7\]](#)). Только его трактовка этого термина имеет отношение скорее к операционной системе как таковой, а не к конкретным дистрибутивам. Когда же речь идет о дистрибутивах, а тем более об их классификации по этому признаку, за "базовое" следовало бы принять тот набор пакетов, который устанавливается программой инсталляции независимо от желания пользователя, то есть даже тогда, когда на этапе выбора пакетов для установки вы снимете отметку со всех пакетов. Но, насколько я знаю, ни одна программа инсталляции не останавливается на той точке, когда установлено это самое "базовое ПО". Так что для выяснения того, насколько разнятся составы этого базового ПО в разных дистрибутивах, пришлось бы провести специальные исследования. Трудоемкий получается критерий! А, кроме того, насколько я могу судить по результатам своих экспериментов по установке Red Hat Linux 9 Cyrillic Edition в минимальной конфигурации (см. [\[8\]](#)), этот набор трудно назвать минимально необходимым для запуска компьютера. Так что какая уж тут базовость?

### **Инструменты управления системой**

Набор тех программных средств, которые служат для конфигурирования, настройки и оптимизации системы, в каждом дистрибутиве разный. И мнения о том, какие средства для этого лучше всего, сильно

разнятся. Одни авторы считают, что кроме редактирования конфигурационных скриптов "ручками", никакие средства и не нужны, другие полагают, что для пользователя необходимы конфигураторы, работающие в графическом режиме. Причем набор графических инструментов для настройки системы практически в каждом дистрибутиве разный (очень красивый инструмент YaST я почему-то не встречал ни в одном другом дистрибутиве). Более того, насколько я помню, даже в последовательных версиях Red Hat этот набор сильно менялся от версии к версии. Если бы существовало небольшое количество стандартных инструментов настройки, из которого создатели дистрибутивов выбирали бы себе наиболее подходящий по их мнению вариант, можно было бы строить критерий классификации по этому признаку. А поскольку многообразие этих средств сравнимо с количеством дистрибутивов, то разумного критерия классификации на этом признаке построить невозможно.

### Носитель, с которого запускается система

Среди тех сотен дистрибутивов, которые существуют в настоящее время, имеются как дистрибутивы, устанавливающиеся на жесткий диск и запускающиеся с него, так и дистрибутивы, которые не требуют установки на жесткий диск. В качестве носителя системы может выступать CD-ROM, USB pendrive, дискета или несколько дискет и даже виртуальный диск, созданный в оперативной памяти компьютера. Однако классифицировать дистрибутивы по этому признаку вряд ли целесообразно. В конце концов, мы пытаемся построить классификацию для того, чтобы облегчить выбор дистрибутива для того или иного применения. И не важно, откуда мы систему запустим, важно то, какую задачу система будет выполнять. Ведь ни один дистрибутив не создается с целью просто запустить систему. Даже дистрибутивы на дискетах создаются для запуска системы в качестве маршрутизатора, файервола, для использования компьютера, на котором такая система, в качестве удаленного терминала или станции для выхода в Интернет.

### Требования к аппаратуре

Что касается требований к аппаратному обеспечению компьютера, на котором будет запускаться система, то этот признак, на мой взгляд, сливается с признаком назначения дистрибутива. Просто среди сфер применения дистрибутива нужно завести отдельную категорию - дистрибутивы для "слабых" компьютеров. В эту категорию частично войдут и те дистрибутивы, которые запускаются с дискет, если у них нет другого предназначения.

Итак, из первоначально приведенного списка признаков для рассмотрения остаются только организация процедуры начальной загрузки системы, используемая система управления пакетами и сфера применения (ориентация или назначение) дистрибутива. Эти три признака классификации дистрибутивов надо рассмотреть подробнее, чем мы далее и займемся.

## 4. Средства управления пакетами ПО

Как было сказано в приведенном выше определении, дистрибутивы состоят из отдельных пакетов, каждый из которых содержит какое-то приложение, утилиту или сервис. Отдельный пакет может содержать, например, веб-браузер, библиотеку для работы с графическими файлами в формате PNG, набор шрифтов и так далее.

Программное обеспечение, содержащееся в пакете, поставляется в одном из двух основных видов:

- в виде **бинарных файлов**, которые предназначены для непосредственной установки в вашу систему, без какой-либо дополнительной обработки (например, компиляции).
- в виде **исходных текстов**, которые обычно содержат текст на каком-то языке программирования, заархивированный в формате tar и упакованный программой gzip, а также вспомогательные файлы, необходимые для компиляции приложения из файлов пакета.

Существуют также пакеты, которые можно при желании отнести как к первому, так и ко второму виду. Это пакеты, которые содержат скрипты и конфигурационные файлы, страницы руководств в формате man/info, информацию о копирайтах и другую документацию. С одной стороны, они представляют собой такие же "исходные тексты", с другой - устанавливаются в систему без всякой дополнительной обработки, как и исполняемые файлы. Но этот тип пакетов играет вспомогательную роль и для нас какого-либо интереса не представляет. А вот количество и состав пакетов первых двух типов играет в классификации дистрибутивов самую существенную роль.

Вспомним, во-первых, что все программное обеспечение в Linux является открытым, то есть поставляется вместе с исходными кодами. И поэтому для каждого бинарного пакета на дистрибутивных дисках найдется



соответствующий ему пакет с исходными кодами. А вот обратное утверждение уже не имеет места, так как существуют дистрибутивы, в которых число бинарных пакетов сильно ограничено. Это так называемые "Source-based" дистрибутивы, то есть дистрибутивы, основанные на исходных кодах. Их создатели предполагают, что пользователь такого дистрибутива может самостоятельно скомпилировать и установить в систему любое нужное ему приложение. Но ведь для того, чтобы запустить процесс компиляции, система в какой-то минимальной конфигурации уже должна работать. Должны быть установлены загрузчик, ядро, архиваторы `tar` и `gzip` (чтобы развернуть пакет с исходниками), компилятор со всем сопутствующим инструментарием (линкером, ассемблером и т.д.), библиотеки функций языка Си, утилиты для работы с файлами и текстами (`find`, `grep`, `awk`, `sed`), без которых сборка и установка программ просто невозможна. Эта проблема решается двумя способами: либо компиляция проходит на какой-то другой системе, либо необходимый минимум прекомпилированных пакетов устанавливается из дистрибутива, а остальные компилируются уже в полученной таким образом среде. Самый яркий пример реализации подхода, основанного на компиляции всей системы из исходных кодов - проект [Linux From Scratch](#), который не является дистрибутивом в прямом смысле этого слова, а представляет собой набор инструкций по созданию системы из набора пакетов с исходными кодами (см. [10]).

Что же касается пакетов с прекомпилированным программным обеспечением, то для установки как в процессе инсталляции, так и в уже установленной системе требуются специальные средства управления пакетами. Дело в том, что установка программного обеспечения из пакетов обычно связана с разрешением так называемых "зависимостей". Например, пакет, содержащий компилятор GNU C (`gcc`) "зависит" от пакета `binutils`, который включает в себя компоновщик и транслятор. Если пользователь попытается установить `gcc` без предварительной установки `binutils`, процесс установки пакета с GCC скорее всего завершится сообщением об ошибке. Поэтому в состав пакета обычно включается не только бинарный код исполняемой программы, но еще и некая служебная или мета-информация: название программы, данные о разработчике, информация о других пакетах, которые необходимы для того, чтобы данное ПО корректно работало (чаще всего это необходимые данному приложению библиотеки), контрольные суммы, информация о том, как правильно сконфигурировать пакет, и как его корректно удалить, если необходимость в его использовании отпала.

Пакет обычно представляет собой один архивный файл, в котором содержится как само устанавливаемое программное обеспечение, так и необходимая служебная информация (мета-информация). Иногда эта мета-информация содержится не в самом пакете, а во внешней базе данных. Но для того, чтобы этой информацией воспользоваться, необходима еще какая-то программа установки, которая инсталлирует приложение из пакета в систему, используя по мере необходимости мета-информацию пакета. Очевидно, что эта программа сильно зависит от того, какая именно мета-информация содержится в пакете и как она организована. Таким образом и получилось, что создатели некоторых дистрибутивов разработали каждый свою структуру пакетов и свои специализированные средства управления пакетами, которые позволяли бы установить программу из пакета, используя содержащуюся в нем мета-информацию.

**Система управления пакетами** - это набор инструментов, предназначенных для автоматизации процессов установки, обновления, конфигурирования и удаления пакетов программного обеспечения определенного формата.

Наиболее известными (или распространенными) системами управления пакетами являются:

- **RPM/YUM** — менеджер пакетов Red Hat. Сейчас аббревиатура RPM расшифровывается обычно рекурсивно (**RPM = RPM Package Manager**), но первоначально ее расшифровывали как менеджер пакетов Red Hat (**Red Hat Package Manager**), поскольку разработана она была для дистрибутива Red Hat. В настоящее время она используется и во многих других дистрибутивах.
- **dpkg/APT** — система управления пакетами \*.deb дистрибутива **Debian**, тоже портированная в настоящее время в другие дистрибутивы. Пакеты \*.deb представляют собой просто два tar-архива, сжатых с помощью `gzip`: в одном архиве содержится управляющая информация, в другом - данные. Стандартным средством управления такими пакетами является консольная программа `dpkg`, дополненная оболочкой **APT (Advanced Packaging Tool)**.
- **tgz** или **tar.gz** — стандартный набор из двух программ `tar` + `gzip`, иногда дополненный некоторыми дополнительными управляющими файлами. Используется в дистрибутиве **Slackware** и некоторых других, не обеспечивает разрешения зависимостей. От Source-based дистрибутивов эта система отличается тем, что внутри `tar.gz`-архивов находятся заранее скомпилированные программы.
- система портежей дистрибутива **Gentoo**, которая представляет собой набор файлов `ebuild`, содержащих информацию о том, как получить (из любых доступных источников - сети, локального

диска и т.д.), скомпилировать и установить пакет в системе Gentoo, используя консольную команду emerge. Обычно пакет ПО в этом случае содержит исходные коды программ, и приложение компилируется прямо в процессе инсталляции, за счет чего оптимизируется для конкретной машины. Хотя этим способом могут устанавливаться и заранее откомпилированные программы, но такой вариант используется только в исключительных случаях, например, при инсталляции системы на очень медленные машины.

- YaST - утилита, разработанная Novell и используемая в дистрибутиве SuSE.

Компиляцию пакетов из исходных кодов тоже можно рассматривать как один из вариантов системы управления пакетами. От перечисленных выше систем он отличается только тем, что пакеты ПО в source-based дистрибутивах почти не содержат в своем составе заранее скомпилированных программ (кроме тех, которые абсолютно необходимы, типа ядра и компилятора), так что единственный способ установки нового пакета заключается в непосредственной компиляции его из исходных кодов.

Наиболее распространенным средством управления пакетами программного обеспечения остается программа RPM. Правда, она обладает тем недостатком, что задача разрешения зависимостей ложится в основном на плечи пользователя. Программа RPM только сообщает, что таких-то пакетов в системе не хватает, а их поиск и установку пользователь должен выполнить самостоятельно. Поэтому многие основанные на RPM дистрибутивы сейчас используют также заимствованный из Debian инструмент APT, который появляется иногда и под другими именами. Дебиановский DEB и TGZ из Slackware (и его производные) тоже распространены достаточно широко. Кроме названных были изобретены еще несколько средств управления пакетами. Примерами могут служить SLP из дистрибутива Stampede, который имеет несколько интересных особенностей, и система управления пакетами дистрибутива JBLinux.

Таблица 1, заимствованная с сайта DistroWatch.com, показывает распространенность различных систем управления пакетами. Имейте в виду, что в отличие от аналогичной таблицы на DistroWatch, я не ставил задачу перечислить в правом столбце таблицы все дистрибутивы, использующие тот или иной способ управления пакетами, а привел только по несколько примеров.

**Таблица 1.**

<b>Система управления пакетами</b>	<b>Число дистрибутивов, использующих эту систему</b>	<b>Примеры дистрибутивов</b>
DEB	121	Adamantix, Damn Small, Debian, GNUsstep, Helix, Knoppix, Kurumin, Linspire, MEPIS, Morphix, Ubuntu, UserLinux, Xandros
RPM	111	Fedora, Hancorn, Linare, Linux XP, Lycoris, Magic, Mandrake, Novell, PLD, Red Flag, Red Hat, SOT, SUSE, Trustix, Turbolinux, Voodoo
TGZ (TAR.GZ, TAR.BZ2, TBZ, DPAK, PKG)	38	Arch, Blin, CRUX, Definity, GoboLinux, LiveCD Router, Sentinix, Slackware, SLAX, STUX, Vector
Source-based	10	Core, LFS, Lunar, Sorcerer, Source Mage
APT-RPM	9	ALT, Ark, ASP, CLE, Conectiva, Lorma, PLD, Vine, Yellow Dog
Portage	9	Gentoo, Gentoox, iBox, Jollix, Knopperdisk, Shark, SystemRescue

BOX	1	Pingwinek
CCS	1	Specifix
CGZ	1	MirOS
INSTALL	1	Athene
NBA	1	Nasgaia
OLM	1	Onebase
RUBYX	1	Rubyx
UHU	1	UHU

Нужно сказать, что провести четкую границу между разными системами управления пакетами иногда довольно затруднительно. Кроме того, существует программа [Alien](#), которая позволяет конвертировать пакеты одной системы в пакеты другой. Если вы хотите использовать пакет из дистрибутива, отличного от установленного на вашем компьютере, вы можете воспользоваться этой программой для конвертации этого пакета в формат, используемый вашей системой, после чего сможете его установить.

## 5. Сценарии начальной загрузки

Способ организации и размещения стартовых сценариев (или скриптов) - это второй существенный признак, по которому дистрибутивы Linux отличаются друг от друга.

Для начала давайте вспомним, что существует два разных стиля начальной загрузки операционной системы типа UNIX, происхождение которых уходит корнями в историю развития UNIX-систем: так называемый стиль BSD (используемый также в таких системах как FreeBSD, NetBSD и OpenBSD), и стиль System V (или стиль АТТ). Различие между ними проявляется в организации и размещении стартовых сценариев (скриптов), обеспечивающих управление процессами загрузки системы. В классических BSD-системах эти файлы хранятся в каталоге /etc и их имена начинаются с префикса "rc". В системах семейства System V файлы сценариев располагаются в каталоге /etc/init.d, а ссылки на них созданы в каталогах /etc/rc0.d, /etc/rc1.d и т.д. Второй вариант организации является более четким и позволяет аккуратнее выполнять останов системы. Если вы хотите подробнее узнать о различиях в двух названных стилях загрузки, прочитайте главу 2 книги [6].

Большая часть дистрибутивов Linux использует стиль System V. К этому классу относятся Debian, все клоны Red Hat, включая Mandrake и российские дистрибутивы ASPlinux и ALT Linux. В стиле BSD организована загрузка в дистрибутиве Slackware и его производных. Однако тот или иной стиль сценариев начальной загрузки выдерживается не очень четко. К примеру, структура каталогов, в которых хранятся инициализационные скрипты, в основных дистрибутивах существенно различается:

### Листинг 1.

<b>Fedora Core (Red Hat)</b>	/etc/init.d/-----символьная ссылка на /etc/rc.d/init.d/ /etc/rc.d/-----скрипты rc, rc.sysinit и rc.local   /init.d/-----множество файлов-скриптов /rc0.d/-----символьные ссылки @K*.* и @S*.* /rc1.d/-----символьные ссылки @K*.* и @S*.* ..... /rc6.d/-----символьные ссылки @K*.* и @S*.* /rcS.d/-----символьные ссылки @K*.* и @S*.*



<b>Knoppix (клон Debian)</b>	/etc/init.d/-----скрипты <b>rc, rcS, reboot</b> и множество других, /etc/rc0.d/-----символьные ссылки <b>@K*.*</b> и <b>@S*.*</b> /rc1.d/-----символьные ссылки <b>@K*.*</b> и <b>@S*.*</b> ..... /rc6.d/-----символьные ссылки <b>@K*.*</b> и <b>@S*.*</b> /rcS.d/-----символьные ссылки <b>@K*.*</b> и <b>@S*.*</b>
<b>Slackware</b>	/etc/rc.d/--скрипты <b>rc.0, rc.4, rc.6, rc.K, rc.M, rc.S, rc.local, rc.syslog, rc.nfsd, rc.sysvinit, rc.netdevice, rc.keymap</b> и другие скрипты, имена которых имеют вид <b>rc.*</b> .
<b>Gentoo</b>	/etc/init.d/--скрипты с самыми разными именами.
<b>SuSE</b>	/etc/rc.d/-----символьная ссылка на /etc/init.d/ /etc/init.d/-----множество файлов с разными именами, в частности <b>boot.*</b> /boot.d/-----символьные ссылки <b>@K*.*</b> и <b>@S*.*</b> /rc0.d/-----символьные ссылки <b>@K*.*</b> и <b>@S*.*</b> /rc1.d/-----символьные ссылки <b>@K*.*</b> и <b>@S*.*</b> ..... /rc6.d/-----символьные ссылки <b>@K*.*</b> и <b>@S*.*</b> /rcS.d/-----символьные ссылки <b>@K*.*</b> и <b>@S*.*</b>

Как видите, даже в Red Hat и Debian, которые следуют стилю System V, структура каталогов несколько отличается. А SuSE, хотя и происходит от Slackware, движется в сторону стиля System V, по крайней мере в организации структуры каталогов. А вот Red Hat, наоборот, завел скрипт rc.local, напоминающий одноименный сценарий во FreeBSD. В процессе загрузки он выполняется последним. Правда, в последних версиях Red Hat и Fedora Core этот скрипт "пустой", то есть практически не используется. И авторы книги [6] не советуют добавлять в него собственные команды, рекомендуя лучше воспользоваться средствами System V.

Но структура каталогов и названия инициализационных скриптов, может быть, и не самое главное. Как известно, в процессе начальной загрузки системы вначале загружается ядро, которое запускает процесс init, всегда имеющий идентификатор 0. Процесс init выполняет инструкции, заданные в файле /etc/inittab. В листинге 2 приведены выдержки из файла /etc/inittab для нескольких дистрибутивов. Приводится не полный текст файла /etc/inittab, а только по 4 группы самых важных строк, определяющих процесс загрузки. Во всех дистрибутивах кроме приведенных инструкций определяются еще действия по комбинации клавиш <Ctrl>+<Alt>+<Del> и выполняется запуск виртуальных терминалов (только в Fedora Core и SuSE для этого вызываются процессы mingetty, в Gentoo и Slackware - agetty, в Knoppix - /bin/bash -login).

#### Листинг 2.

<b>Fedora Core (Red Hat)</b>	id:3:initdefault:  # System initialization. si::sysinit:/etc/rc.d/rc.sysinit  l0:0:wait:/etc/rc.d/rc 0 l1:1:wait:/etc/rc.d/rc 1 l2:2:wait:/etc/rc.d/rc 2 l3:3:wait:/etc/rc.d/rc 3 l4:4:wait:/etc/rc.d/rc 4 l5:5:wait:/etc/rc.d/rc 5 l6:6:wait:/etc/rc.d/rc 6  x:5:respawn:/etc/X11/prefdm -nodaemon
------------------------------	--

<b>Knoppix (клон Debian)</b>	<pre> id:5:initdefault:  # Boot-time system configuration/initialization script. si::sysinit:/etc/init.d/rcS  # What to do in single-user mode. ~~:S:respawn:/bin/bash -login &gt;/dev/tty1 2&gt;&amp;1 &lt;/dev/tty1  10:0:wait:/etc/init.d/knoppix-halt 11:1:wait:/etc/init.d/rc 1 12:2:wait:/etc/init.d/rc 2 13:3:wait:/etc/init.d/rc 3 14:4:wait:/etc/init.d/rc 4 15:5:wait:/etc/init.d/rc 5 16:6:wait:/etc/init.d/knoppix-reboot  # Run X Window session from CDROM in runlevel 5 w5:5:wait:/bin/sleep 2 x5:5:wait:/etc/init.d/xsession start </pre>
<b>Slackware</b>	<pre> id:3:initdefault:  # System initialization (runs when system boots). si:S:sysinit:/etc/rc.d/rc.S  su:1S:wait:/etc/rc.d/rc.K rc:2345:wait:/etc/rc.d/rc.M 10:0:wait:/etc/rc.d/rc.0 16:6:wait:/etc/rc.d/rc.6  # Runlevel 4 used to be for an X window only system x1:4:wait:/etc/rc.d/rc.4 </pre>
<b>Gentoo</b>	<pre> id:3:initdefault:  # System initialization (runs when system boots). si:S:sysinit:/sbin/rc boot  ~~:S:wait:/sbin/sulogin 10:0:wait:/sbin/rc shutdown 11:1:wait:/sbin/rc single 12:2:wait:/sbin/rc nonetwork 13:3:wait:/sbin/rc default 14:4:wait:/sbin/rc default 15:5:wait:/sbin/rc default 16:6:wait:/sbin/rc reboot z6:6:respawn:/sbin/sulogin  # Used by /etc/init.d/xdm to control DM startup. x:a:once:/etc/X11/startDM.sh </pre>

SuSE	<pre> # The default runlevel is defined here id:5:initdefault:  # First script to be executed, if not booting in emergency (-b) mode si::bootwait:/etc/init.d/boot  # /etc/init.d/rc takes care of runlevel handling 10:0:wait:/etc/init.d/rc 0 11:1:wait:/etc/init.d/rc 1 12:2:wait:/etc/init.d/rc 2 13:3:wait:/etc/init.d/rc 3 #14:4:wait:/etc/init.d/rc 4 15:5:wait:/etc/init.d/rc 5 16:6:wait:/etc/init.d/rc 6  # what to do in single-user mode ls:S:wait:/etc/init.d/rc S ~:S:respawn:/sbin/sulogin </pre>
------	--

Как видите, во всех случаях порядок действий примерно одинаков:

- Вначале задается уровень выполнения (runlevel).
- Затем выполняются действия начальной инициализации системы, не зависящие от заданного уровня. В разных дистрибутивах для этого вызываются разные скрипты:
  - в Fedora Core - /etc/rc.d/rc.sysinit
  - в Knoppix - /etc/init.d/rcS
  - в Slackware - /etc/rc.d/rc.S (то есть скрипт перехода в Single User mode):
  - в Gentoo - /sbin/rc boot
  - в SuSe - /etc/init.d/boot
- Затем выполняется скрипт перехода на заданный уровень выполнения. При этом, если в Fedora Core и Gentoo для каждого уровня выполнения задана своя строка, то в Slackware для уровней со 2-го по 5-ый выполняются одни и те же действия.
- Наконец, во всех дистрибутивах, кроме SuSE, отдельная строка задает процедуры запуска графической оболочки.

При использовании стиля System V используется один и тот же скрипт перехода на заданный уровень, а то, что он делает, определяется содержимым каталога /etc/rc.d/rcN.d. Этот каталог содержит перечень ссылок на скрипты запуска тех системных сервисов, которые должны работать на уровне N. Сами скрипты размещаются в каталоге /etc/init.d или /etc/rc.d/init.d.

В отличие от стиля System V в BSD-стиле каждому уровню загрузки соответствует свой сценарий. И сначала всегда происходит переход на уровень S (однопользовательский), а затем уже переход на заданный уровень. Стиль BSD в наиболее чистом виде (из рассматриваемых примеров) представлен в дистрибутиве Slackware.

Поскольку стиль System V взят за основу при создании стандарта LSB (Linux Standard Base), дистрибутивы, ранее использовавшие стиль BSD, в последнее время заботятся о совместимости с System V. Slackware обеспечивает такую совместимость начиная с версии 7.0. Достигается это путем использования сценария rc.sysinit, который производит поиск всех сценариев стиля System V в каталоге /etc/rc.d и выполнит их, если уровень загрузки соответствующий. Это полезно, если вы пользуетесь коммерческим программным обеспечением, которое ориентируется на стиль System V. В то же время, вы можете пользоваться и BSD сценариями.

### **Ориентация дистрибутива на область применения**

А вот о направленности дистрибутива на определенную сферу применения поговорить стоит. Сфера применения - понятие довольно широкое. Как уже было сказано выше под этот термин подходят и требования к аппаратному обеспечению компьютера (дистрибутивы для 486-х, для компьютеров без CD-ROM и т.д.), и дистрибутивы, запускаемые с "нетрадиционных" носителей (с CD-ROM, flash-дисков, внешних или переносных жестких дисков и т.д.). И число градаций или групп, выделяемых по этому признаку, может быть очень большим. Я составил по материалам из Интернет таблицу 2, в которой приведены отдельные примеры того, на какие сферы применения могут быть ориентированы дистрибутивы. Предупреждаю, что большая часть пояснений дана на основе найденных в сети описаний и не была проверена на личном опыте, так что за ошибки прошу строго не судить.

**Таблица 2.**

<b>Ориентация</b>	<b>Основные представители</b>
-------------------	-------------------------------

Дистрибутивы  
общего  
назначения

**[Fedora Core](#)** - дистрибутив общего назначения, для рабочих станций, персональных компьютеров и серверов. Разрабатывается сообществом независимых разработчиков при поддержке фирмы Red Hat.

**[Debian GNU/Linux](#)** - дистрибутив, отличающийся тем, что содержит самую большую коллекцию пакетов программного обеспечения (свыше 15000). В его разработке участвует огромное количество независимых разработчиков со всего мира. Debian поддерживает самое большое число аппаратных платформ и славится легкостью обновления программного обеспечения. С другой стороны считается, что установка этого дистрибутива более трудна, чем других. И в дистрибутив включается только строго некоммерческое ПО.

**[SuSE](#)** - один из самых популярных в Европе дистрибутивов, приобретенный в конце 2003 года фирмой Novell. В его состав входит уникальная утилита конфигурирования YaST, которая служит для установки и обновления пакетов, настройки сети и файервола, выполнения разнообразных административных задач.

**[Mandrakelinux](#)**. Самый легкий дистрибутив для начального освоения. Первоначально это был просто вариант Red Hat, оптимизированный для процессора Pentium, но со временем он отошел от своего предка довольно далеко, сохраняя тем не менее полную совместимость с Red Hat.

**[Slackware](#)**. Старейший дистрибутив из числа активно поддерживаемых. Консольная система управления пакетами. Ориентирован на применения, в которых требуются повышенная безопасность и стабильность.

**[ALT Linux](#)**. Российский дистрибутив (первоначально - просто русифицированный Mandrake), поддерживающий собственный репозиторий пакетов Sisyphus, и использующий систему управления пакетами APT от Debian.

**[ASP Linux](#)**. Российский дистрибутив, основанный на Red Hat.

**[Gentoo](#)**. Дистрибутив для разработчиков и сетевых профессионалов. Все программное обеспечение поставляется в исходных кодах. Этот уникальный подход применяется с двумя намерениями: обеспечить возможность запуска на любой аппаратной платформе и оптимизировать производительность путем компиляции исполняемого кода непосредственно на том компьютере, где будет работать система. Инструкции по инсталляции прилагаются и достаточно полны, но требуется изрядное количество "ручной работы". Фактически Gentoo представляет собой автоматизированный Linux From Scratch.

**[Linux From Scratch](#)**. Это не дистрибутив, а документ, описывающий как собрать свою собственную операционную систему, основанную на ядре Linux. Некоторые пользователи выбирают Linux from Scratch из-за той гибкости, которую он предоставляет. Gentoo предоставляет тот же самый уровень гибкости, но автоматизирует значительную часть работы.

**[Linspire](#)**. Дистрибутив, который вначале назывался **Lindows** и позиционировался как замена Windows на десктопах. Основан на коде Xandros и Debian. Дополнительное ПО может устанавливаться с помощью менеджера пакетов APT от Debian, но полная совместимость с Debian не гарантируется. В состав этого дистрибутива включается значительное количество проприетарного софта.

**[PLD Linux Distribution](#)**. Этот дистрибутив разработан в Польше.

Дистрибутивы для мэйнфреймов	<a href="#"><u>Linux on zSeries</u></a>
Серверные дистрибутивы	<p><a href="#"><u>Red Hat Enterprise Linux</u></a>,  <a href="#"><u>SuSE Linux Enterprise Server</u></a>  <a href="#"><u>ALT Linux Master</u></a> – универсальный дистрибутив GNU/Linux, включающий в себя множество подготовленных к эксплуатации программных решений для серверов и рабочих станций.</p> <p><a href="#"><u>ASP Linux Server</u></a> включает все необходимые средства для создания корпоративных серверов различных классов и поддерживает различные серверные архитектуры, включая многопроцессорные, и системы с большими объемами памяти. Централизованная настройка большинства сервисов и программ существенно облегчает их установку, использование и администрирование.</p> <p><a href="#"><u>LTSP</u></a> - Linux Terminal Server Project. Проект по созданию терминальных серверов, позволяющих использовать "слабые" компьютеры в качестве удаленных терминалов.</p>
Защищенные дистрибутивы	<p><a href="#"><u>Trustix</u></a>. Дистрибутив, обеспечивающий повышенную безопасность. В отличие от SELinux, который является набором модификаций ядра, а не дистрибутивом, позволяет установить полноценную систему.</p> <p><a href="#"><u>SELinux</u></a>. Улучшенная в части безопасности версия ядра Linux, созданная американским Агентством Национальной Безопасности (National Security Agency - NSA).</p> <p><a href="#"><u>Tinfoil Hat Linux</u></a>. Дистрибутив для людей, параноидально заботящихся о безопасности.</p> <p><a href="#"><u>Trinux</u></a>. Дистрибутив, оснащенный набором средств для обеспечения безопасности.</p> <p><a href="#"><u>"Утес-К"</u></a>. Защищённая интегрированная система обработки конфиденциальной информации «Утёс-К» является универсальным дистрибутивом Linux, содержащим более 2500 пакетов программ и подробную документацию. Система сертифицирована Гостехкомиссией по 5 классу СВТ защиты конфиденциальной информации, 4 уровню контроля отсутствия недеklarированных возможностей — сертификат N685.</p>
Дистрибутивы для мультимедиа	<p><a href="#"><u>MoviX</u></a>. Предназначен исключительно для воспроизведения мультимедийных файлов (просмотр видео, прослушивание аудио). Но зато уж делающий это очень хорошо.</p> <p><a href="#"><u>Agnula</u></a>.  <a href="#"><u>Dynebolic</u></a>.</p>
Дистрибутивы для маршрутизаторов и файрволов	<p><a href="#"><u>Coyote Linux</u></a>. Дистрибутив Linux на дискете. Позволяет связать локальную и Интернет сети. Может быть использован в качестве межсетевого экрана.</p> <p><a href="#"><u>Linux Router Project</u></a>. Проект по созданию полноценного Linux рутера на одной дискете.</p> <p><a href="#"><u>Gibraltar</u></a>.  <a href="#"><u>IPCop Firewall</u></a>.  <a href="#"><u>Sentry Firewall</u></a>.  <a href="#"><u>SmoothWall</u></a>.</p>



Дистрибутивы для встроенных систем	<p><a href="#"><b>Embedded Debian</b></a>.  <a href="#"><b>ELKS Linux</b></a> = Embeddable Linux Kernel subset (ELKS).  <a href="#"><b>Linux Microcontroller Project</b></a> - создание ОС на основе Linux для встроенных систем.</p>
Дистрибутивы для "слабых" компьютеров	<p><a href="#"><b>Vector Linux</b></a> - дистрибутив, основанный на Slackware. Задумывался как небольшой, быстрый дистрибутив, предназначенный как для старых, так и для современных моделей компьютеров.  <a href="#"><b>Small Kernel Project</b></a> - маленький Linux для i386, требующий всего 2Mb ОЗУ и 40Mb на жестком диске.  <a href="#"><b>ttylinux</b></a> запускается даже на 386-х компьютерах, требует всего 4 МБ на жестком диске и 6 МБ ОЗУ. Основное назначение - дозвониться до провайдера и выйти в Интернет.</p>
Дистрибутивы, запускаемые с CD и bootable business card, т.е. CD объемом не более 50 MB.	<p><a href="#"><b>Knoppix</b></a>. Дистрибутив, запускающийся с CD и славящийся тем, что прекрасно определяет аппаратную конфигурацию компьютера, а потому успешно работающий с самым разным "железом". Основоположник огромного ряда дистрибутивов, запускающихся с CD.  <a href="#"><b>PCLinuxOS</b></a>. Легкий в использовании дистрибутив, основанный на Mandrake 9.2. Но затем он начал развиваться независимо, ориентируясь на использование на десктопах.  <a href="#"><b>FeatherLinux</b></a> - урезанная версия Knoppix, запускающаяся с business card CD или USB pendrive. . Требуется 115 МБ дискового пространства и включает в себя все ПО, необходимое большинству пользователей ежедневно.  <a href="#"><b>Damn Small Linux</b></a> - великолепный дистрибутив, объемом 50 МБ.</p>
Дистрибутивы для USB	<p><a href="#"><b>Flonix</b></a> - дистрибутив, запускаемый с USB flash-drive.  <a href="#"><b>Flash Puppy</b></a> - очень маленький, но полнофункциональный дистрибутив, полностью загружающийся на виртуальный диск, то есть работающий полностью в оперативной памяти.  <a href="#"><b>SPBLinux</b></a> - разработанный в Санкт-Петербурге дистрибутив, запускаемый с USB flash-drive и дискет.</p>

Мини-дистрибутивы, запускаемые с дискет	<p><a href="#">BasicLinux</a> - дистрибутив, размещающийся на 2 дискетах и работающий на самых минимальных конфигурациях компьютеров.</p> <p><a href="#">muLinux</a> - Расширяемый мини-дистрибутив (Linux на одной дискете, X-windows на другой), который может очень много: может быть ftp, http, samba, file, printer, fax, dial-on-demand, dial-in сервером, имеет плагины: Perl, TEX, GCC, TCL, EMU (DOS and Wine) и так далее. Удобная установка, настройка, возможна конфигурация через http.</p> <p><a href="#">tomsrtbt</a> - дистрибутив, размещающийся на нескольких дискетах, но тем не менее, включающий в себя даже графическую оболочку.</p> <p><a href="#">BrutalWare Linux</a> - Linux на трех дискетах.</p> <p><a href="#">Linux one disk</a> - Linux на одной дискете. Предназначен для восстановления "рухнувших" систем. Один из лучших однодисковых дистрибутивов.</p> <p><a href="#">Linux router на одной дискетке</a> - Router/firewall для маленьких сетей.</p> <p><a href="#">Linux Svnviewer one floppy</a> - Linux+VNCviewer на одной дискете.</p> <p><a href="#">LOAF</a> - Linux On A Floppy - проект "Linux на одной дискете".</p> <p><a href="#">Monkey</a> - Mini Linux - дистрибутив на 5-ти дискетах. Есть Иксы, поддержка Ethernet карт и многое другое.</p> <p><a href="#">Pocket Linux</a> - Linux на одной дискете, SSH + BOOTP + Modem supporting.</p>
Дистрибутивы, запускаемые из-под Windows	<p><a href="#">Cooperative Linux</a> - дистрибутив, запускаемый из-под Microsoft Windows без применения коммерческого ПО, предназначенного для создания виртуальных компьютеров.</p>

Замечу, что на сайте Linux Weekly News приведена гораздо более полная таблица, классифицирующая дистрибутивы по их назначению (см. [3]).

## 7. Итоговая таблица и заключение

Подытоживая вышеизложенное, можно констатировать, что существует три существенных критерия классификации дистрибутивов Linux:

- стиль загрузки;
- система управления пакетами;
- назначение дистрибутива.

И если свести все это в единую таблицу, получим "периодическую систему дистрибутивов Linux", представленную в таблице 3.

**Таблица 3.**

	Стиль и система управления пакетами
--	-------------------------------------

Назначение дистрибутива	System V RPM/YUM	System V DEB	System V APT-RPM	System V TGZ	System V Source	System V Portage	System V Другое	BSD RPM/YUM	BSD DEB	BSD APT-RPM	BSD TGZ	BSD Source	BSD Portage	BSD Другое
Серверные дистрибутивы	Red Hat Enterprise Linux, SuSE,													
Общего назначения	Fedora Core, Mandrake,	Debian,	ASPLinux, ALT Linux		LFS,	Gentoo,					Slackware, Arch,			
Защищенные	Trustix,													
Для мультимедиа														
Маршрутизаторы														
Для встроенных систем														
Для "слабых" систем и мини														
Запускаемые с нетрадиционных носителей (Live CD, USB и т.д.)		Knoppix, DamnSmall												
Запускаемые из-под Windows														

Эта таблица может наращиваться "по вертикали" путем добавления новых сфер применения Линукс. К сожалению, разнести все существующие на сегодняшний день дистрибутивы по клеточкам этой таблицы я не в состоянии. Занес туда только те дистрибутивы, с которыми сам знаком. Если вы работали в другими дистрибутивами и можете мне помочь в заполнении таблицы, пишите на адрес **kos at rus-linux dot net**. Я постараюсь заполнять эту таблицу по мере поступления таких сообщений.

## Ссылки:

1. [DistroWatch.com](http://distrowatch.com)
2. [http://en.wikipedia.org/wiki/Linux\\_distribution](http://en.wikipedia.org/wiki/Linux_distribution)
3. [The Linux Weekly News comprehensive list of distributions](http://lwn.net/Articles/111111)
4. [Distribution Reviews](http://www.linuxreviews.org/)
5. А.Федорчук, "[Демоны, пингины и пользователи. Начнем с пингинов.](#)"

6. А.Федорчук, ["О дистрибутивах Linux"](#)
7. А.Федорчук, ["Кое-что об ОС, Unix'ax, Linux'ax и BSD"](#)
8. В.Костромин, "Red Hat Linux 9 Cyrillic Edition с точки зрения пользователя". Часть 1. Инсталляция.  
[1.2. Минимальная конфигурация. Базовый набор.](#)
9. Э.Немет, Г.Снайдер, С.Сибасс, Т.Хейн, "UNIX: руководство системного администратора", Киев, БХВ, 1999 г.
10. Джерард Бикманс, [Linux From Scratch](#), версия 5.0, перевод: Виталий Катраев.
11. Sean Russell, [RPM Hell. A Perfect Example of Good Software Crippled by Bad Design.](#)

[ опубликовано 31/01/2005 ]