

ПРИЗНАЧЕННЯ ТА СТРУКТУРА ПЕРШОГО СЕКТОРА HDD. УСТАНОВКА ЗАВАНТАЖУВАЧА ОС. МУЛЬТИЗАВАНТАЖЕННЯ ОС.

Сектор (англ. sector) — ділянка доріжки магнітного диску, що є мінімальною фізично адресованою одиницею пам'яті. Першим фізичним сектором на жорсткому диску або іншому пристрої зберігання інформації, що розбивається на логічні диски (розділи), є **головний завантажувальний запис**.

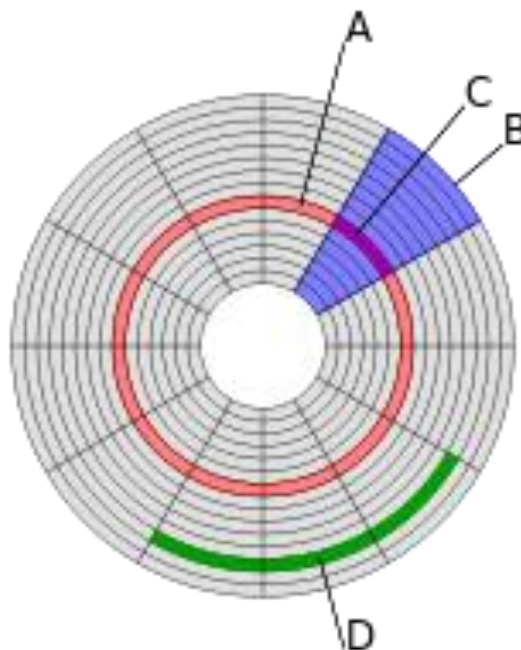


Рисунок 1 - Структура диска: (A) доріжка; (B) геометричний сектор; (C) сектор доріжки; (D) кластер

Для більш ефективного використання місця на диску файлова система може об'єднувати сектори в кластери, розміром від 512 байт (один сектор) до 64 кбайт (128 секторів). Перехід до кластерів відбувся тому, що розмір таблиці FAT був обмежений, а розмір диска збільшувався. У разі FAT16 для диска об'ємом 512 Мб кластер буде складати 8 Кб, до 1 Гб - 16 Кб, до 2 Гб - 32 Кб і так далі.

Через популярність файлової системи NTFS та інших, в яких блок розміщення файлів становить 4 Кб, з 2011 року комп'ютерні диски почали відформатовуватися на 4К сектори, що дозволило збільшити обсяг диска на 7 -

11 %. Старі версії операційних систем можуть не підтримувати новий формат дисків.

Головний Завантажувальний Запис - Master Boot Record (MBR)

Спочатку визначимося з термінологією. Під словами **Master Boot Record (MBR)** розуміють часто дві різні речі, які в деяких випадках можуть бути еквівалентні, а в деяких - ні. Перше поняття - власне Master Boot Record. Це запис (програмний код і дані), який завантажується в пам'ять з вінчестера і забезпечує розпізнання логічних розділів на ньому, визначає активний розділ і завантажує з нього завантажувальний запис (**завантажувач**) (Boot Record - BR), який продовжить запуск операційної системи (ОС). І друге поняття - Завантажувальний Сектор, **Master Boot Sector (MBS)** - це сектор, розташований на циліндрі 0, площини (головка) 0 і має номер 1. У більшості випадків MBS містить весь необхідний код і всі дані, тому його вмістом і є MBR, проте бувають випадки, коли код і дані не поміщаються в одному секторі (просто не вистачає місця або з міркувань безпеки), і тоді код цього сектора забезпечує завантаження у пам'ять інших секторів. В цьому випадку MBR - це сукупність всіх секторів, які повинні бути завантажені, а MBS - всього лише перший сектор.

1. Розглянемо випадок, коли MBR і MBS - одне і те ж, і будемо називати їх більш звичним терміном MBR. Такий MBR, що забезпечує завантаження будь-якої ОС і займає тільки MBS, зазвичай називають терміном **Generic MBR**.

Взагалі MBR з'явився на жорстких дисках починаючи з MS DOS версії 3.0, у більш ранніх версіях жорсткий диск форматувався як дискета, і в першому секторі розташовувався BR. Відповідно диск являв собою один розділ і не міг бути розбитий на логічні частини - правда, при тих розмірах дисків, які тоді випускалися, це було неактуально.

Розмір сектора на жорсткому диску - 512 байт. Цього простору цілком вистачає для розміщення там необхідного і коду, і даних. Однак тільки одна структура повинна там бути присутньою обов'язково - це сигнатура. Цим словом називається спеціальна, суворо встановлена, послідовність з 2 байт з

шістнадцятковим значеннями 55h AAh, яка записується в останні 2 байти сектора і відповідно має зсув від початку сектора 1FEh. Якщо хоча б один з двох останніх байтів відрізняється за значенням, вважається, що перший сектор не є MBR і не містить осмисленої інформації. Якщо комп'ютер при старті, прочитавши перший сектор, не виявить правильної сигнатури, він не буде передавати управління розташованому там коду, навіть якщо він правильний, а видасть повідомлення про те, що головний завантажувальний запис не знайдено. Або буде пробувати знайти її на інших пристроях - наприклад, на дискеті. Зауважимо, що BR також містить сигнатуру 55h AAh в останніх двох байтах.

Розглянемо структуру першого сектора. Перед сигнатурою, впритул до неї, розташовані 4 блоки даних по 16 байтів кожний (відповідно зі зміщенням від початку сектора 1BEh, 1CEh, 1DEh, 1EEh). Сукупність цих блоків називається **ТАБЛИЦЯ РОЗДІЛІВ, PARTITION TABLE (PT)**, а кожен окремий запис - елементом таблиці розділів (Partition Table Entry) або просто **розділом (Partition)**. Цих 16 байтів цілком достатньо, щоб вказати всі необхідні характеристики розділу, а саме: тип розділу, ознака активності розділу, початковий і кінцевий сектори розділу в форматі: Циліндр (доріжка) - Головка (сторона) - Сектор (Cylinder - Head - Sector, CHS), відносний номер першого сектора (щодо MBR) і кількість секторів в розділі.

Решту вільного простору сектора зайнято програмним кодом, який забезпечує розбір PT, пошук активного розділу, завантаження в пам'ять BR цього розділу та передачу йому управління. Легко підрахувати, що на код залишається $512 - 4 * 16 - 2 = 446$ байт. Цього простору з надлишком вистачає для виконання зазначених дій.

Отже, загальна структура MBR може бути представлена наступною таблицею:

Зсув	Довжина	Опис
000h	446	Код завантажувача
1BEh	64	Таблиця розділів
	16	Розділ 1
1CEh	16	Розділ 2
1DEh	16	Розділ 3
1EEh	16	Розділ 4
1FEh	2	Сигнатура (55h AAh)

Кожен 16-байтний блок, що описує один розділ, має наступну структуру:

Зсув	Довжина	Опис
00h	1	Ознака активності розділу
01h	1	Початок розділу - головка
02h	1	Початок розділу - сектор (біти 0-5), доріжка (біти 6,7)
03h	1	Початок розділу - доріжка (старші біти 8,9 зберігаються в байті номера сектора)
04h	1	Код типу розділу
05h	1	Кінець розділу - головка
06h	1	Кінець розділу - сектор (біти 0-5), доріжка (біти 6,7)
07h	1	Кінець розділу - доріжка (старші біти 8,9 зберігаються в байті номера сектора)
08h	4	Зсув першого сектора
0Ch	4	Кількість секторів розділу

Код типу розділу являє собою однобайтовий ідентифікатор. Якщо його значення - 00h, то вважається, що в даному елементі РТ не містяться дані про розподіл, і його вміст ігнорується. Будь-яке нульове значення означає, що в зазначеному просторі знаходиться розділ певного типу. Деякі значення однозначно вказують тип розділу, деяким відповідають кілька можливих типів, і визначення конкретного типу покладається на операційну систему, інші зарезервовані для майбутнього використання.

Ознайомтеся з таблицею типів розділів, які створюються операційними системами Windows NT / 2000 / XP:

Код	Тип розділу
01h	12-бітна FAT
04h	16-бітна FAT до 32 Мбайт
05h	Розширений розділ
06h	16-бітна FAT понад 32 Мбайт
07h	Windows NT NTFS (і деякі інші - тип визначається по вмісту BR)
0Bh	32-бітна FAT
0Ch	32-бітна FAT з використанням розширеного управління INT13
0Eh	LBA VFAT (то ж що і 06h, з використанням розширеного управління INT13)
0Fh	LBA VFAT (то ж що і 05h, з використанням розширеного управління INT13)
17h	Прихований розділ NTFS
1Bh	Прихований розділ 32-бітної FAT (то ж що 0Bh)
1Ch	Прихований розділ 32-бітної FAT з використанням розширеного управління INT13 (то ж що 0Ch)
1Eh	Прихований розділ LBA VFAT (то ж що і 06h, з використанням розширеного управління INT13)
86h	Розділ FAT-16 stripe-масиву Windows NT
87h	Розділ NTFS stripe-масиву Windows NT
B6h	Дзеркальний master-розділ FAT-16 Windows NT
B7h	Дзеркальний master-розділ NTFS Windows NT
C6h	Дзеркальний slave-розділ FAT-16 Windows NT
C7h	Дзеркальний slave-розділ NTFS Windows NT

Ознака активності розділу - тобто ознака того, що операційну систему слід завантажувати саме з цього розділу - може мати значення 80h (розділ активний) і 00h (розділ не активний). У загальному випадку кількість активних

розділів має бути не більше 1 (інакше як зробити вибір?). Якщо активних розділів немає - значить з цього жорсткого диска ОС не може бути завантажена. Інші значення вважаються помилковими і ігноруються. Втім, рішення про передачу управління приймає код завантажувача, тому значення байта ознаки «завантажувальний» - аксіома тільки для стандартних завантажувачів.

Трибайтовий блок адреси початку і адреси кінця розділу мають ідентичний формат. Тут фактично використовується упаковка значень для того, щоб вони мали мінімальний обсяг. Формат упаковки повністю відповідає тому, як ці дані передаються процедурам роботи з жорстким диском (Int 13h), що знаходяться в BIOS комп'ютера, тому і накладні обчислювальні витрати виходять мінімальними. При цьому циліндри і доріжки нумеруються, починаючи з нульового значення, а сектора - з першого - так склалося історично.

Сектор, на який вказує адреса початку розділу, містить в собі спеціальний запис, яка називається **ЗАВАНТАЖУВАЛЬНИМ ЗАПИСОМ (BR)**. Його призначення і склад ми розглянемо пізніше.

Зсув першого сектору розділу - це фактично номер цього сектору якщо всі сектори жорсткого диска перенумерувати починаючи з 0 (відповідно до нумерації, яка застосовується Int 25h / 26h) в порядку зростання спершу по секторах однієї доріжки, далі в порядку збільшення номерів головок і, нарешті, циліндрів. Природно, що всі ці значення пов'язані простими залежностями, адже у них інформація надлишкова. Наведемо формули залежності між ними.

Отже, якщо позначити:

C_M - циліндр, на якому розташовується MBR;

H_M - доріжка, на якій розташовується MBR;

S_M - сектор, в якому розташовується MBR;

$C_S, H_S, S_S, C_E, H_E, S_E$ - то ж, для секторів початку (S) і кінця (E) розділу;

H_H - кількість доріжок у жорсткого диска;

S_H - кількість секторів на одній доріжці у жорсткого диска,

то:

Абсолютний номер сектора, в якому знаходиться РТ:

$$\text{Num}_{\text{PT}} = C_{\text{M}} * H_{\text{H}} * S_{\text{H}} + H_{\text{M}} * S_{\text{H}} + S_{\text{M}} - 1$$

Абсолютний номер сектора початку розділу:

$$\text{Num}_{\text{S}} = C_{\text{S}} * H_{\text{H}} * S_{\text{H}} + H_{\text{S}} * S_{\text{H}} + S_{\text{S}} - 1$$

Абсолютний номер сектора кінця розділу:

$$\text{Num}_{\text{E}} = C_{\text{E}} * H_{\text{H}} * E_{\text{H}} + H_{\text{E}} * E_{\text{H}} + E_{\text{E}} - 1$$

Зсув першого сектора розділу:

$$\text{Offset}_{\text{S}} = \text{Num}_{\text{S}} - \text{Num}_{\text{PT}}$$

Кількість секторів розділу:

$$\text{Amount} = \text{Num}_{\text{E}} - \text{Num}_{\text{S}} + 1$$

З наведених формул, випливає, що важливе значення має те, скільки доріжок має один циліндр жорсткого диска і скільки на кожній доріжці секторів. Ці значення залежать як від геометрії жорсткого диска, так і від обраного в установках BIOS режиму трансляції. Тому диск, поділений на розділи в одному режимі трансляції, може виявитися нечитабельним при зміні режиму трансляції.

Для IDE-накопичувачів існують кілька режимів трансляції:

CHS (Cylinder-Head-Sector) - при цьому геометрія диска вважається такою, якою він її повідомляє комп'ютеру. Не спокушайтеся - у більшості накопичувачів реальна геометрія зовсім не така. Однак контролер, що входить до складу жорсткого диска, проводить необхідні перетворення самостійно, і то, як він це робить, для нас несуттєво. При цьому максимальна кількість циліндрів - 1024 (від 0 до 1023), доріжок - 16 (від 0 до 15), секторів - 63 (від 1 до 63), а максимальний обсяг диску, доступний в даній трансляції без застосування спеціальних програм - 504 Мбайти. При дискових операціях адреса сектору передається BIOS комп'ютера контролеру жорсткого диска без зміни.

LBA (Logical Block Addressing) - у цьому режимі трансляції використовується не та геометрія диска, яку він повідомляє BIOS комп'ютера. Виконується перетворення до формату, коли кількість циліндрів не перевищує 1024, а кількість секторів на доріжку рівне 63. Наведена кількість доріжок при цьому залежить від BIOS комп'ютера і об'єму жорсткого диска і може дорівнювати 16, 64, 128 або 255, останні версії BIOS як правило використовують приведення до

255 (0-254) доріжках незалежно від обсягу накопичувача. При зверненні до диска передані в LBA-трансляції номер циліндра, головки і сектора перераховуються в абсолютний номер сектора, і саме він передається BIOS комп'ютера контролеру жорсткого диска для виконання операції. При обсязі накопичувача понад 8 Гбайт кількість циліндрів виходить більшою 1024, тому на комп'ютерах, які не підтримують роботу розширеного режиму Int 13h, без установки спеціального програмного забезпечення простір за межами 8 Гбайт недоступний (незалежно від режиму трансляції). Іноді, втім, допомагає оновлення BIOS комп'ютера.

LARGE, або ECHS (Extended CHS) - при цьому режимі трансляції проводиться перетворення числа циліндрів до значення менше 1024 за рахунок кратного збільшення кількості доріжок. Утворена кількість доріжок строго кратна реальній, кількість секторів на доріжку не змінюється. Втім, у більшості сучасних накопичувачів кількість секторів на доріжку рівно 63. Максимально доступний обсяг на комп'ютерах, які не підтримують роботу розширеного режиму Int 13h, залежить від реальної кількості доріжок, але не більше 8 Гбайт (наприклад, якщо диск має 16 доріжок, то в LARGE трансляції їх може бути 16, 32, 48 ... 240, але не 255, тому що 255 не кратне 16, а максимальний доступний обсяг - $1024 * 240 * 63 * 512 / (1024^3) = 7.38$ Гб).

При використанні трансляції LBA або LARGE за рахунок округлення до цілих значень кілька секторів в кінці накопичувача можуть "випасти" з описаного простору і стати недоступними. Втім, втрати зазвичай невеликі.

Що ж стосовно SCSI-накопичувачів, то вони завжди працюють в режимі LBA-трансляції. Вірніше, контролер SCSI представляє геометрію накопичувача відповідно до вимог LBA-трансляції, а сам при зверненні до диска передає йому абсолютну адресу сектора. Природно, що саме на BIOS SCSI-контролера покладається обов'язок виробляти необхідні перерахунки.

Повернемося до MBR. Як уже сказано, він містить 4 блоки даних про елементи таблиці розділів. Це означає, що максимальна кількість розділів, яка може бути описана в MBR, дорівнює чотирьом. Однак це аж ніяк не означає, що максимальне число розділів, на які може бути розділений накопичувач, дорівнює чотирьом. Для подолання цього бар'єру був введений спеціальний тип розділу з кодом 05h - **розширений розділ (Extended Partition)**.

РОЗШИРЕНИЙ РОЗДІЛ відрізняється від всіх інших типів розділів. По-перше, він описує не розділ, а область простору накопичувача, в якій розташовані інші розділи. При цьому кількість розділів, що знаходяться в ньому, теоретично не обмежена. Правда, ті розділи, які розташовані в цій області, дещо "обмежені в правах", найістотнішим обмеженням є те, що вони не можуть бути активними (вірніше, можна зробити так, що з такого розділу можна завантажити систему, але штатні засоби більшості існуючих ОС цього не дозволяють, доведеться використовувати спеціальні засоби). По-друге, в MBR має бути присутній тільки один запис про розширений розділ. Вірніше, їх можна зробити і більше (хоч всі чотири), але як поведе себе ОС, зустрівши таке, передбачити важко. Наприклад MS-DOS 6.20 просто ігнорує все розширені розділи, крім першого в списку, як ніби їх взагалі немає. По-третє, на відміну від інших типів розділів в тому секторі, який прописаний в структурі як сектор початку розділу, міститься аж ніяк не BR. Там знаходиться фактично ще один MBR, який має сигнатуру і таблицю розділів, але зазвичай не містить програмного коду (зазвичай сектор, що містить таблицю розділів, але не містить коду початкового завантаження, називають **Abstract MBR**). Втім, оскільки там немає активних розділів, то і код ні до чого.

У таблиці розділів такого сектора є зазвичай один або два записи. Перший описує звичайний розділ (Partition), причому цей розділ повинен повністю перебувати всередині простору Extended Partition. Якщо звичайний розділ займає не весь простір, в таблиці розділів з'являється другий елемент, який описує простір, що залишився як Extended Partition. У наступному секторі так само описується один розділ і, якщо місце залишилося, ще запис про Extended Partition. І так триває до тих пір, поки простір не закінчиться. Фактично всі записи про розширені розділи представляють собою пов'язаний ланцюг (Extended Partition Chain), у якому від дискового простору відщипують шматочки на звичайні розділи, поки місце не скінчиться. Помилка в будь-якому елементі цього ланцюга призведе до його розсіпання, в результаті всі записи після розриву не будуть знайдені ОС, а займаний ними простір ОС буде вважати незайнятим.

При заповненні ланцюга зазвичай ОС дотримується декількох правил. По-перше, описаний в черговій "ланці" ланцюга звичайний розділ не повинен розташовуватися в середині, оскільки тоді для опису отриманих двох шматків незайнятого простору потрібно в PT цього елемента ввести два записи про два

різних розширених розділах, а ОС зазвичай ігнорують всі такі записи крім першого, і в результаті частина дискового простору випаде з розбиття. По-друге, як правило запис про звичайний розділ робиться так, щоб він займав початкову область розширеного розділу, а наступний елемент ланцюга розділів - залишок.

Які ж проблеми можуть виникнути з вмістом MBR? По-перше, фізичне або логічне руйнування, тобто пошкодження поверхні або інша механічна проблема або руйнування сервометок, що не дає можливості прочитати цей сектор з диска. І друга - це руйнування всієї або частини інформації, що міститься в секторі, в результаті чого розділи або не можуть бути знайдені операційною системою, або їх параметри визначаються невірно.

Найлегший випадок - це руйнування сигнатури. При цьому ОС вважає, що в секторі міститься якась випадкова інформація, "сміття", а сам накопичувач взагалі не поділений на розділи і ніякої інформації на ньому немає. Для відновлення достатньо всього лише будь-яким засобом прямого доступу до секторів диска (найбільш популярний DISKEDIT з пакета NORTON UTILITIES) відновити сигнатуру. Більшість ОС потрібно перезавантажити, оскільки ОС як правило при старті зчитують інформацію про розбиття диска на розділи і далі в процесі роботи її зміни не враховують.

Більш складний випадок - це руйнування коду. При цьому втрачається можливість зробити завантаження операційної системи з накопичувача, а спроба завантаження як правило закінчується "зависанням" комп'ютера. У той же час якщо завантажитися з іншого носія (інший жорсткий диск, дискета, завантажувальний CD-ROM і т.п.), то вся інформація на накопичувачі доступна для використання. В цьому випадку рекомендується використовувати штатні засоби відновлення коду завантажувача, які є в кожній ОС. Наприклад, в ОС Windows для цієї мети використовується програма FDISK.EXE, що запускається з ключем / MBR.

Руйнування коду - не настільки рідкісний випадок, як може здатися. Як правило, подібна неприємність відбувається, коли на один накопичувач послідовно встановлюються різні ОС в один або різні розділи. Будь-яка ОС при установці бажає мати в MBR власний код, але далеко не всі піклуються про збереження того коду, який був в секторі раніше. Наприклад ОС Windows переписує код завантажувача, не ставлячи до відома користувача і безповоротно

знищуючи старий вміст. Втім, такий випадок якраз не дуже страшний, оскільки всі завантажувачі ОС дуже схожі. Проблема виникає, якщо для розбиття накопичувача на розділи використовувалися програмні засоби третіх фірм, такі як EZ-drive, ODM або наприклад SpeedStore, для яких саме поняття MBR і MBS не еквівалентні. Через інші ідеології завантаження код, що знаходиться в MBS, виконує іншу функцію, а саме пошук і завантаження частини коду MBR, що розташовується в інших секторах диска. Заміна коду призводить до того, що ця функція втрачається, і відповідно виникають проблеми різного характеру. Користувач повинен бути досить обережний, якщо використовує подібні програмні засоби - відновлення коду в таких випадках може виявитися вельми непростою справою.

І найбільш важкий і неприємний випадок - це руйнування самої таблиці розділів. Втім, як відомо, біда ніколи не приходить одна, і найчастіше руйнуються всі три компоненти MBR, але саме руйнування PT призводить до найбільш тяжких наслідків, оскільки при цьому втрачається можливість доступу до інформації, що зберігається на накопичувачі. PT може бути зруйнована повністю, а може і частково - тобто частина елементів зруйнована, а решта цілі. Бувають випадки, коли PT, що знаходиться в MBR, цілий, а зруйнований запис про розділи в одному з ланок ланцюга Extended Partition. Однак оскільки структура MBR і структура Extended Partition абсолютно ідентичні (за винятком того, що в елементах Extended Partition відсутній код), методика відновлення в обох випадках одна і та ж.

Існує достатня кількість програм, які вміють відновлювати зруйновані PT. Якість їх роботи різна і залежить головним чином від того, наскільки складним було розбиття диска на розділи, чи всі типи розділів, які були на диску, відомі програмі і чи немає крім руйнування PT ще якихось пошкоджень інформації в інших секторах, особливо в BR розділах.

Отже, для роботи буде потрібно: завантажувальна дискета з будь-яким DOS, на яку скопійований файл DISKEDIT.EXE, олівець, папір, калькулятор (втім, калькулятор є в DISKEDIT) і трохи мізків. Бажано, звичайно, щоб і DOS, і DISKEDIT були свіжіші. Я використовую DOS 7.10 від Windows 98 SE і DISKEDIT з пакета NORTON UTILITIES 2002. Ніякі драйвери нам на цьому етапі не потрібні, ну крім тих випадків, коли накопичувач підключений до старого SCSI-контролера і без завантаження драйвера просто не видно. Можна

завантажити драйвер мишки - буде трохи зручніше. Тепер головна дрібниця - крім зазначених файлів на дискеті не повинно бути нічого! Це важливо - якщо на дискеті немає конфігураційного файлу, DISKEDIT.EXE запуститься в режимі тільки читання (Read-Only) і ніякі дані на диску не будуть змінені.

А дискету взагалі б закрити від запису.

Завантажуємося з дискети. Запускаємо DISKEDIT.EXE. Після завантаження натискаємо клавіші Alt-D (або через меню Object - Drive). Виводиться вікно з доступними дисковими пристроями. Спершу зазначимо, що нам потрібні фізичні пристрої (Physical disks), а потім виберемо потрібний диск (припустимо, Hard Disk 1) і натиснемо ОК. При цьому в якості діапазону перегляду будуть обрані всі сектори диска від першого до останнього. Це нам і потрібно.

Спочатку попросимо програму виконати перегляд диска і знайти всі сектори, які можуть бути елементами ланцюга Extended Partition або BR. І хоча при цьому DISKEDIT буде просто шукати сектори, що мають сигнатуру, а не аналізувати вміст - результати можуть здорово полегшити роботу. Правда доведеться застатися терпінням - процес це не швидкий, та до того ж кожен знайдений сектор потрібно буде брати на олівець.

Отже, Tools - Find Object - Partition / Boot. Кожен раз, коли трапляється сектор з сигнатурою, пошук зупиняється, на екран виводиться дамп сектора, а в правому нижньому кутку - номер сектора. Саме ці адреси і потрібно брати на олівець. Втім, якщо при розбитті диска на розділи не використовувалися особливо витончені методи, всі сектора, що цікавлять нас, будуть розташовуватися на нульовій або першій стороні в першому секторі, тобто Side 0 або 1, Sector 1. Решта сектора, наприклад який-небудь Cyl 12, Side 4, Sector 52 можна сміливо проігнорувати - це випадково. Правда, ми отримуємо абсолютний номер сектора, але це не страшно, для "правильних" секторів номер буде без остачі ділитися на кількість секторів на доріжку, зазвичай 63 (інші значення зустрічаються зараз набагато рідше - 17, 26, 40, 56, тому майте на увазі, що на Вашому конкретному накопичувачі можливо доведеться використовувати інше число). А цю цифру ми можемо подивитися через меню (Info - Drive Info). На жаль, кількість сторін і циліндрів там може виявитися неправильним (що не відповідає використовуваній трансляції), але і це не страшно. Записавши номер чергового сектора, продовжуємо пошук (можна через меню Tools - Find Again, можна просто Ctrl-G).

І так поки не отримаємо повідомлення що об'єкт не знайдено. У цей момент у нас на руках (вірніше, на папері) всі номери секторів, в яких присутня сигнатура. Кількість сторін і циліндрів там може виявитися неправильною (що не відповідає використовуваній трансляції), але і це не страшно. Тепер обробляємо список, відсіваючи явно випадкові номери (це які не діляться на 63), і особливо виділяючи пари номерів, які розрізняються на 63. Ці пари - не що інше як пара з елемента Extended Partition і BR описаного в ньому розділу.

Тепер відкладемо на хвилинку листок з цифрами і спробуємо пригадати, якого розміру розділи були на диску. Так, великими мазками, 600 мегабайт, 12 гігабайт ... запишемо все що пам'ятаємо. Якщо сума не дорівнює обсягу накопичувача - або щось забулося, або невірнo згадалося, або був простір, не належав жодному розділу (а що, буває ... знаю не один випадок, коли десятигігабайтний жорсткий диск розбивався на комп'ютері, материнська плата якого не розуміла більше восьми гігабайт, а при апгрейді це якось не згадалося ... ось так 2 гігабайти і зависли). На цьому етапі бажано згадати ще й типи файлових систем в кожному з розділів.

Тепер візьмемо обидва листка і спробуємо поєднати отримані дані. Мегабайт - дві тисячі секторів, гігабайт - два мільйони ... приблизно. Але зазвичай вдається абсолютно однозначно накласти одне на інше. Корисно буває намалювати довгий прямокутник, розставити на ньому межі, відповідні знайденим секторам, і поділити на шматочки, що відповідають розмірам розділів. Навіть якщо однозначності немає - не проблема. При поєднанні даних рекомендую пам'ятати, що деякі BR і елементи ланцюга Extended Partition можуть бути зруйновані (і відповідно не будуть знайдені), причому найбільш часто руйнуються BR першого (по положенню на накопичувачі) розділу і BR активного розділу.

Втім, на даному етапі, мабуть, все ... і не тому, що далі нікуди, а з іншої причини - для подальшої роботи з відновлення потрібен аналіз інших структур, які розташовуються вже в "знайдених" розділах - це BR, FAT / MFT, каталоги та ін ... ми їх поки не розглядали. Втім, іноді і знайденої і згаданої інформації достатньо.

Іноді знання структури розділів застосовують абсолютно для інших цілей - наприклад, для створення свого власного розбиття диска на розділи.

При створенні нестандартного розбиття диска на розділи, крім вимог, описаних раніше (один розширений розділ і ін.). Рекомендую також враховувати

те, в якому порядку ОС MS-DOS і Windows призначають розділах букви логічних дисків. Букви присвоюються, починаючи з С: (А: і В: зарезервовані для дисководів гнучких дисків, можливо віртуальних).

Порядок підключення такий:

1. Розділ, з якого завантажується ОС (при завантаженні з жорсткого диска).
2. Перші первинні розділи інших жорстких дисків в порядку їх нумерування (ініціалізації) BIOS комп'ютера.
3. Розділи в Extended Partition жорстких дисків в порядку їх нумерування (ініціалізації) BIOS комп'ютера, в порядку їх запису в Partition Table дисків.
4. Решта первинні розділи дисків, в порядку їх запису в Partition Table по порядку їх нумерування (ініціалізації) BIOS комп'ютера.
5. Пристрої, що формуються драйверами, що запускаються в файлах config.sys і autoexec.bat, в порядку їх формування та ініціалізації, якщо формованому пристрою не призначається в явній формі певна буква або діапазон букв.
6. Для ОС, які можуть впізнавати і підключати накопичувачі, які не ініціалізуються BIOS комп'ютера (не описані в установках CMOS накопичувача) - розділи цих накопичувачів відповідно до правил 3 і 4 в порядку ініціалізації накопичувачів операційною системою.

Розділи невідомих ОС типів HE ініціалізуються і букви їм не присвоюються. Слід пам'ятати, що ОС сімейства Windows NT мають штатні засоби перевизначення букв логічних дисків.

Приклад 1. Ручне розбиття на розділи.

- Параметри накопичувача в LBA-трансляції (взяті з BIOS - Autodetect Hard Disk) - Cylinders 1216, Heads 255, Sectors 63, Capacity 10 Gb.
- Бажане розбиття: система 2 Гбайт, дані - 2 Гбайт, ігри і дистрибутиви - інше. Бажано розмістити системний розділ в кінці диска (по тестах там найшвидша область), ігри - на початку диска (по завіреннях фахівців, найбільш часто дані пошкоджуються саме там).
- Операційна система - Windows 98 SE rus, всі розділи - FAT-32.

Виходячи з бажаного розбиття, бачиться наступна схема: спочатку розширений розділ з двома логічними дисками в ньому - 6 і 2 Гбайт, потім

первинний активний розділ 2 Гбайт, або 3 первинних розділи в зазначеному порядку і з зазначеними розмірами. Останній варіант нам не підходить (неважливо з яких міркувань). Особливості роботи програми FDISK обраної ОС не дозволяють виконати розбиття штатно: якщо спочатку ми створимо первинний розділ, він буде перебувати на початку накопичувача, якщо ж ми спочатку створимо розширений розділ, то програма відмовляється створювати первинний.

Приймаємо рішення провести розбиття за допомогою FDISK наскільки можливо, а потім доробити вручну. Перший етап: створити розширений розділ з двома дисками; другий: вручну додати запис про первинний розділ.

Перший етап проблем не викликає: завантажуюмося з дискети, створюємо extended partition розміром 8 Гбайт, і в ньому два логічних диски - 6 Гбайт і 2 Гбайт. Перезавантажуємося з дискети, переконуємося, що на диску з'явилися (але недоступні - адже не форматовані розділи!) Диски C: і D:. Форматуємо їх за допомогою стандартного FORMAT і в процесі форматування переконуємося, що диск C: має розмір 6 Гбайт, диск D: - 2 Гбайт. Запускаємо DISKEDIT і дивимося вміст MBR. У ньому є такий запис:

```

Starting location Ending location Relative Number of
System Boot Side Cylinder Sector Side Cylinder Sector Sectors Sectors
EXTEND No 1 0 1 254 972 63 63 15631182
unused No 0 0 0 0 0 0 0 0
unused No 0 0 0 0 0 0 0 0
unused No 0 0 0 0 0 0 0 0

```

Нам потрібно додати запис про первинний розділ. Тип (System) буде 0Ch (FAT32x), ознака завантаження встановлена, початок розділу за адресою 973/0/1, кінець розділу за адресою 1215/254/63, відносний сектор початку розділу 15631245 ($973 * 255 * 63 + 0 * 63 + 1 - 1$), відносний сектор кінця розділу 19535039 ($1215 * 255 * 63 + 254 * 63 + 63 - 1$), кількість секторів 3903795 ($19535039 - 15631245 + 1$).

Вельми істотна тонкість - якщо номер циліндра більший 1023, у відповідне поле вноситься значення 1023 - на жаль, це максимальне значення, яке можна туди записати - а правильне значення ОС розрахує виходячи з заданої кількості секторів.

Переводимо DISKEDIT в режим Read-Write (Tools-Configuration) і в другому рядку вписуємо розраховані дані. Після введення і перевірки всіх значень виходимо з DISKEDIT клавішею Esc, а на питання, що робити зі змінами,

відповідаємо - записати (write). Перезавантажуємо комп'ютер. Переконаємося, що на диску тепер три розділи - C: - недоступний, D: - 6 Гбайт, E: - 2 Гбайт. Форматуємо диск C: з перенесенням на нього системних файлів, в процесі форматування переконаємося, що диск C: має розмір 2 Гбайт. Оновлюємо код MBR командою FDISK / MBR, виймаємо дискету, перезавантажуємо комп'ютер, переконаємося, що ОС завантажилася, диски C:, D: і E: доступні і мають розміри 2, 6 і 2 Гбайт. Запускаємо NDD і переконаємося, що помилок ні в таблиці розділів, ні на дисках немає. Розбиття закінчено.

Тепер таблиця розділів при перегляді через DISKEDIT виглядає так:

```
Starting location Ending location Relative Number of
System Boot Side Cylinder Sector Side Cylinder Sector Sectors Sectors
EXTEND No 1 0 1 254 972 63 63 15631182; Розширений розділ
FAT32x Yes 0 973 1 254 1023 63 15631245 3903795; Первинний розділ (C :)
unused No 0 0 0 0 0 0 0 0
unused No 0 0 0 0 0 0 0 0
```

Якщо поставити курсор на рядок із записом про розширений розділ і натиснути Enter, то DISKEDIT автоматично перенесе перегляд в сектор, на який вказує адреса початку розділу (для первинного розділу це буде BR, для розширеного - елемент ланцюга розділів). У нашому випадку ми побачимо такий вміст елемента ланцюга розділів:

```
Starting location Ending location Relative Number of
System Boot Side Cylinder Sector Side Cylinder Sector Sectors Sectors
FAT32x No 2 0 1 254 728 63 63 11711259; Логічний розділ (D :)
EXTEND No 0 729 1 254 972 63 11711322 3919923; Розширений розділ
unused No 0 0 0 0 0 0 0 0
unused No 0 0 0 0 0 0 0 0
```

Продовжимо рух по ланцюгу розділів в наступний елемент:

```
Starting location Ending location Relative Number of
System Boot Side Cylinder Sector Side Cylinder Sector Sectors Sectors
FAT32x No 1 729 1 254 972 63 63 3919860; Логічний розділ (E :)
unused No 0 0 0 0 0 0 0 0
unused No 0 0 0 0 0 0 0 0
unused No 0 0 0 0 0 0 0 0
```

Черговий елемент ланцюга не містить записи у розширеному розділі. Ланцюг закінчився.

Мультизавантаження (англ. Multi-boot) — технічна можливість вибору завантаження операційної системи при початковому запуску комп'ютера. Для налаштування такої можливості може знадобитися спеціальний завантажувач.

Мультизавантаження дозволяє використовувати декілька операційних систем на одному комп'ютері, що може бути корисно для:

- розробки і дослідження ПЗ або нової комп'ютерної архітектури;
- розробки, тестування і вивчення операційних систем;
- роботи з ПЗ, яке підтримується на різних ОС;
- забезпечення різних умов безпеки;
- зниження витрат на додаткове обладнання.

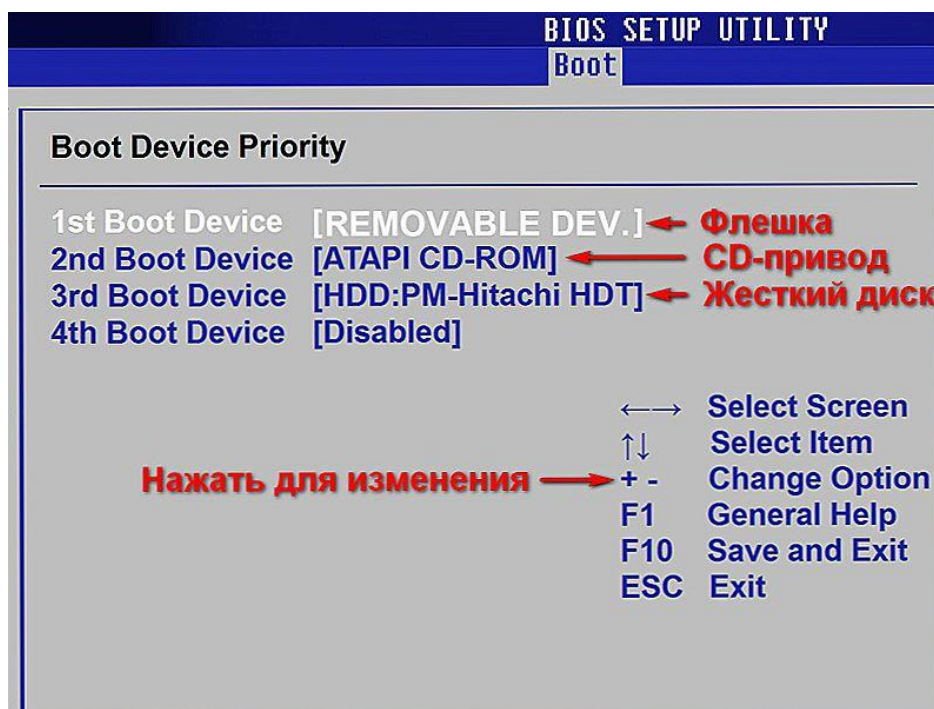
У x86-системах мультизавантаження можливе при використанні NTLDR, LILO, GRUB та інших завантажувачів, які підтримують таку функціональність. Альтернативою мультизавантаження може стати віртуалізація. Використання цього способу вимагає установки гіпервізора і налаштування віртуальних машин, після чого на ці віртуальні машини встановлюються необхідні ОС. Таким чином, можна одну ОС «розгорнути всередині» іншої. Перевагою такого підходу є контроль і безпека.

Установка **GRUB** доцільна в тому випадку, якщо його фізично не існує на пристрої, іншими словами, він може бути стертий після інсталяції іншої системи (найчастіше Windows). Іноді трапляється й таке, що копії завантажувача не функціонують належним чином або GRUB, за бажанням власника ПК, замінюється іншим типовим загрузчиком. Якими б не були причини, установка GRUB, в будь-якому випадку, повинна проводитися виключно з "живою" системою (Knoppix, наприклад). Розглянемо способи установки і настройки завантажувача GRUB.

Завантажувач дозволяє вибрати, яку запустити операційну систему.

У разі наявності не одного жорсткого диска, користувач може налаштувати в BIOS порядок, згідно з яким буде здійснюватися запуск конкретних жорстких дисків в момент завантаження системи. Завдяки цій особливості, нові ПК мають можливість завантажувати систему як із зовнішнього диска, так і з USB-накопичувача. Природно, змінивши порядок

завантаження в BIOS, користувач може завантажувати систему за допомогою CD і DVD-дисків.



За допомогою BIOS ви можете відрегулювати порядок завантаження носіїв.

Щоб в майбутньому не доводилося розбиратися з завантажувачами і думати про те, яким чином повинна здійснюватися настройка GRUB, користувач повинен знати, як встановити дві операційні системи від різних виробників таким чином, щоб вони не конфліктували між собою. Як показує практика, найрозумніше встановити спочатку завантажувач системи Linux, щоб згодом мати можливість вибору щодо запуску систем.

Однак, якщо на пристрої встановлена одна з останніх версій ОС від компанії Microsoft, то в засобі завантаження системи можна зробити так, щоб вона самостійно "турбувався" про GRUB. Величезною перевагою такої можливості є відсутність необхідності зайвий раз розбиратися з особливостями MBR.

Серед інших способів, налаштувати нормальну працездатність двох систем, можна вдавшись до додаткової установки GRUB. Завантажувач необхідно розташувати безпосередньо в завантажувальний сектор головного розділу і згодом відзначити його в якості активного. Цей метод також виключає необхідність використовувати MBR, проте він може спрацювати тільки з

основними розділами та файловими системами, що не впливають на завантажувальний сектор розділу.

Після того, як буде прийнято рішення зробити GRUB install, **користувачеві необхідно створити резервну копію MBR.** Справа в тому, що неправильно внесені зміни "допоможуть покласти" як Windows, так і Linux. У цьому випадку неминуче знадобиться система Knoppix або інсталяційний DVD-диск для Windows, які допоможуть відредагувати завантажувальний сектор. Виправити свої дії можна буде в кілька разів швидше, якщо в наявність буде резервна копія MBR. Для цього слід виконати наведені нижче команди:

```
root # dd if = / dev / sda of = / boot / bootsektor.scsi bs = 1 count = 446
```

Відновлення завантажувального сектора може бути здійснено тільки за допомогою команди:

```
root # dd if = / boot / bootsektor.scsi of = / dev / sda bs = 1 count = 446
```

Однак, слід врахувати, що за допомогою цих команд користувачеві вдасться змінити лише 446 байт MBR.

Інсталяція на USB-накопичувач

Завдяки сучасним технологіям, кожен бажаючий може завантажити операційну систему за допомогою USB-накопичувача. Якщо GRUB install буде проводитися безпосередньо з зовнішнього носія інформації, а не за допомогою Windows або Linux, користувач повинен перевірити правильність розпізнавання накопичувача материнською платою.

Перш ніж почати роботу, слід відформатувати флешку одним з двох відомих способів. За допомогою першого методу вона очиститься як superfloppy, іншим - як жорсткий диск. Вибирати варіант очищення слід виходячи з особливостей BIOS.

Наступним кроком повинна стати активізація підтримки USB-накопичувача, для чого в BIOS передбачений окремий параметр. При розпізнаванні USB-накопичувача як окремого завантажувального диска, в BIOS буде змінений список, згідно з яким GRUB черзі списує дані з носіїв. В якості першого диска стане USB-накопичувач, інші жорсткі диски стануть доступні за

допомогою hdl і hd2. Якщо виникне необхідність, перед GRUB install можна змінити /boot/grub/devices.map.

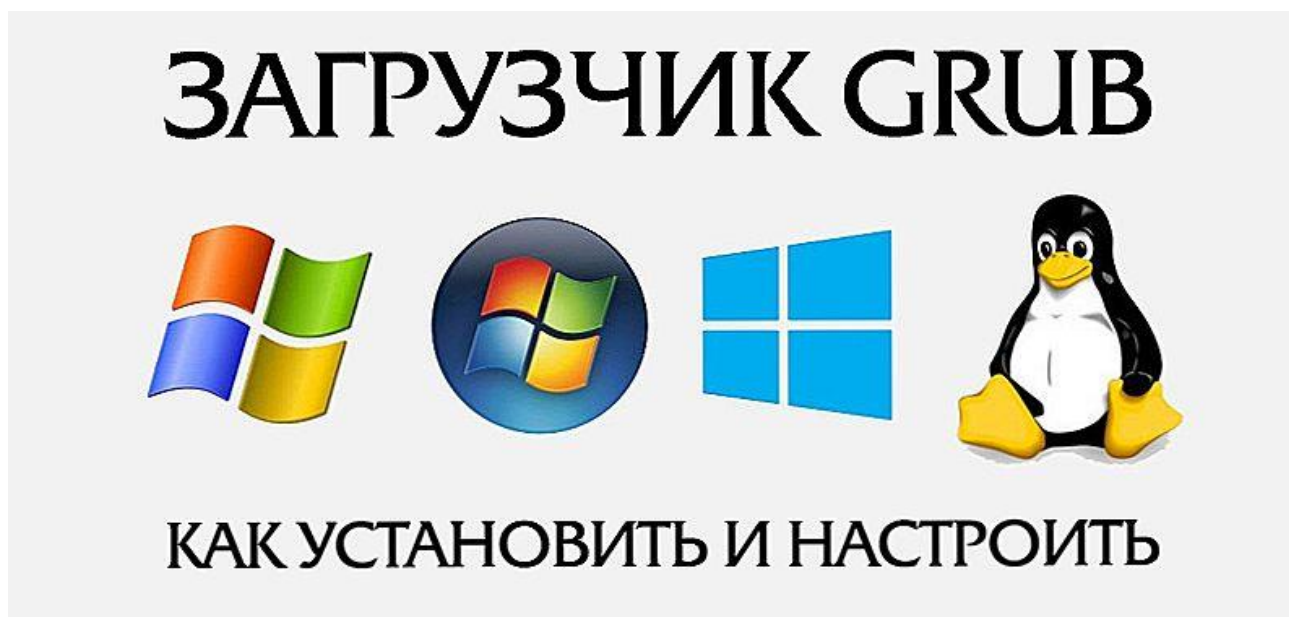
Установка і настройка завантажувача GRUB ОС Linux

зміст

- 1 Вступ
- 2 Що являє собою завантажувальний сектор?
- 3 Особливості завантажувального сектора
- 4 включення пристрою
- 5 Для чого важливо створити копію MBR?
- 6 Інсталяція в MBR жорсткого диска
- 7 Способи інсталяції завантажувача
- 8 Інсталяція на USB-накопичувач
- 9 висновок

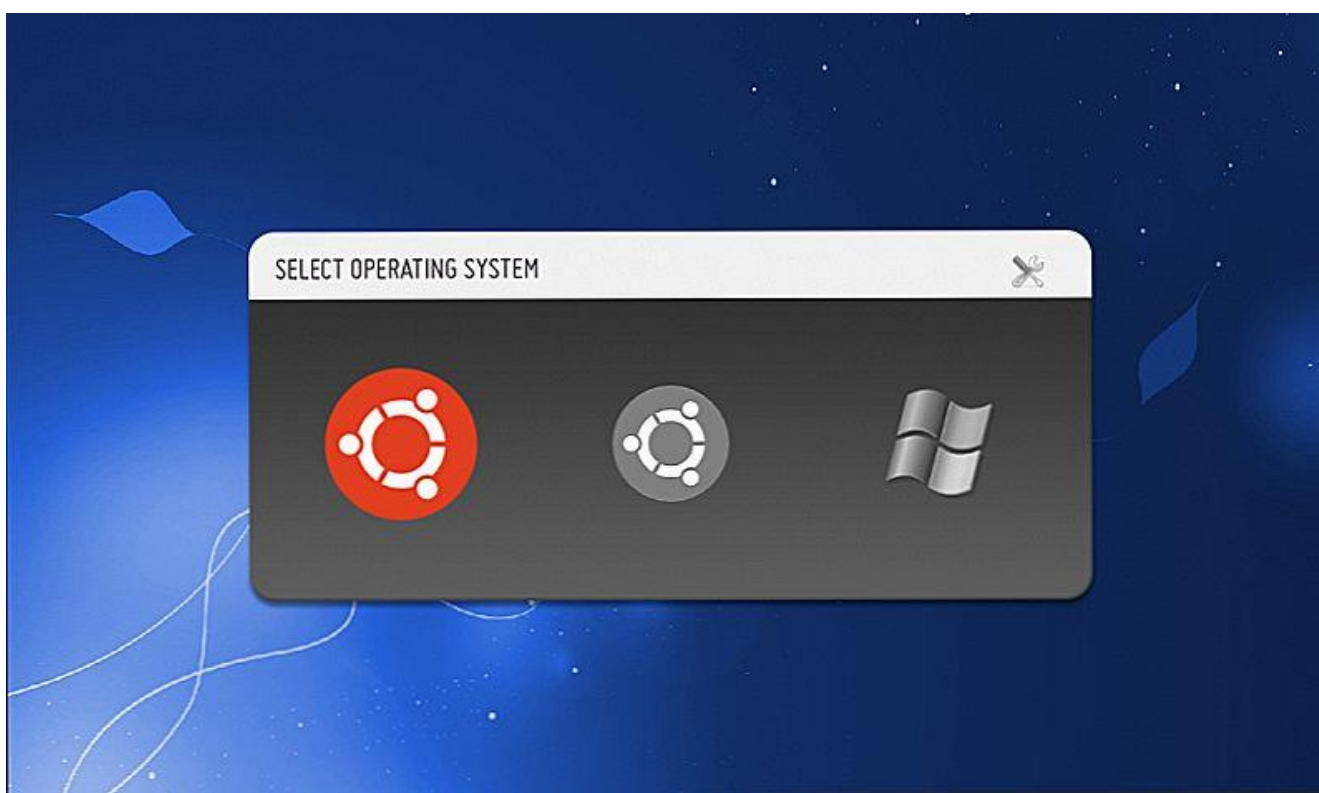
Вступ

Як правило, додаткова настройка GRUB не потрібно, всі необхідні файли встановлюються на комп'ютер самостійно під час інсталяції ОС Linux. Немає необхідності заново встановлювати завантажувач, якщо, наприклад, користувачеві потрібно провести конфігурацію, так як в цьому випадку слід тільки редагувати вміст / boot / grub / menu.lst.



Як зробити правильну установку і настройку – читайте нижче

Установка GRUB доцільна в тому випадку, якщо його фізично не існує на пристрої, іншими словами, він може бути стертий після інсталяції іншої системи (найчастіше Windows). Іноді трапляється й таке, що копії завантажувача не функціонують належним чином або GRUB, за бажанням власника ПК, замінюється іншим типовим загрузчиком. Якими б не були причини, установка GRUB, в будь-якому випадку, повинна проводитися виключно з "живою" системи (Knoppix, наприклад). Справа в тому, що без функціонуючого завантажувача, запустити на ПК дистрибутив Linux, який розташовується на жорсткому диску, просто не вийде. У статті розглянемо способи установки і настройки завантажувача GRUB.



Завантажувач дозволяє вибрати, яку саме операційну систему запустити

Що являє собою завантажувальний сектор?

Якщо власник ПК впевнений в своїх силах і знаннях, йому буде достатньо прочитати лише розділ, в якому покроково розписаний процес установки завантажувача GRUB, однак тим користувачам, які вперше стикаються з подібною проблемою, краще попередньо дізнатися про особливості BIOS і MS DOS.

Жорсткий диск розділений на кілька секторів, у першому з яких здійснюється основна запис диска (користувачі звикли називати її як MBR).

Для цього потрібно всього 512 байт, в яких "захована" невелика програмка. Наступний сектор займає таблиця розбиття диска, яка, за звичаєм, складається з чотирьох основних розділів, обсягом 64 байта, і цифрового підпису (тільки 2 байта).

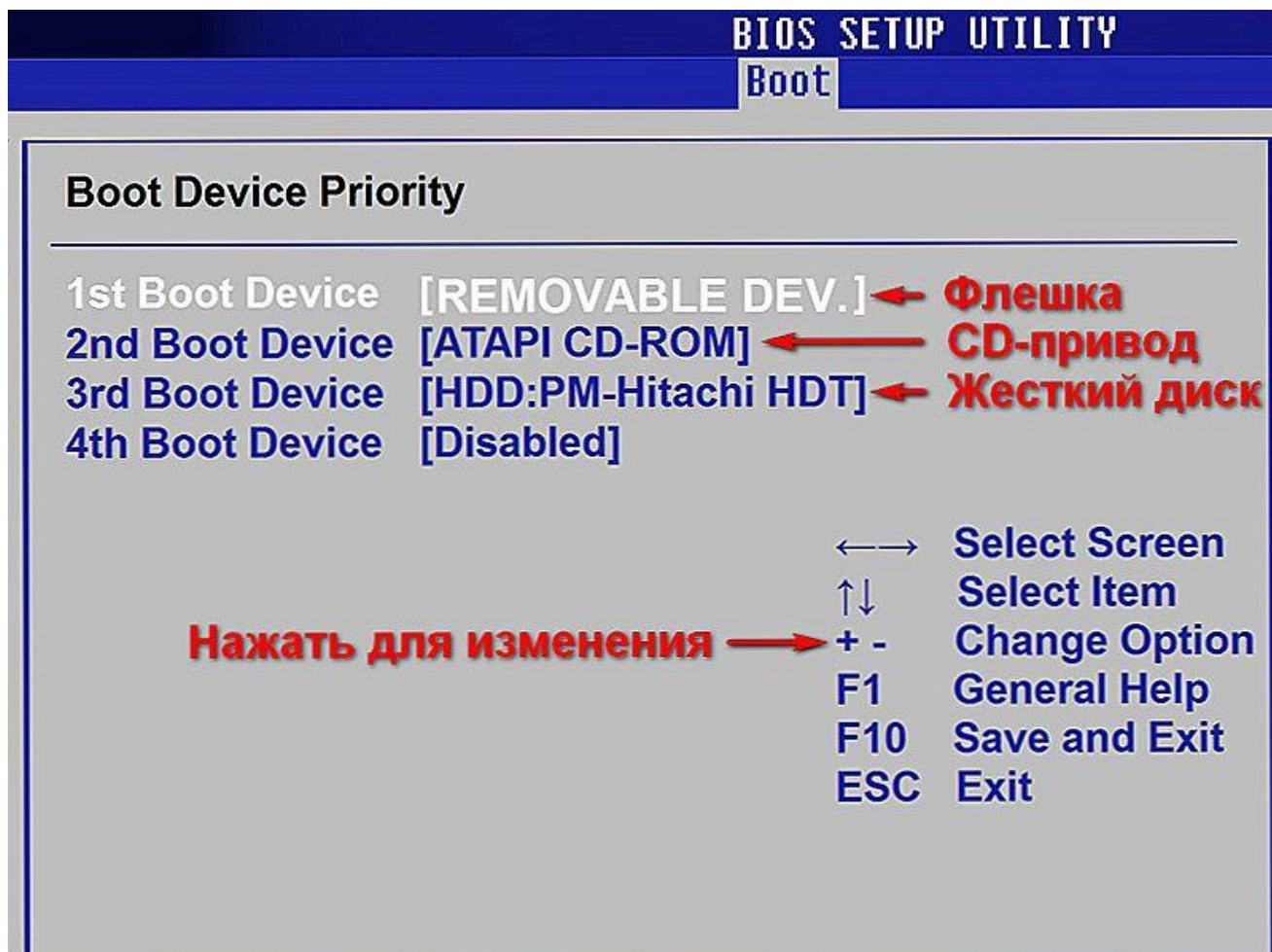
Особливості завантажувального сектора

Завантажувальний сектор можна назвати основою основ, так як він знаходиться в MBR і у всіх інших розділах. Як показує практика, його укладають в усі 16 секторів, на які розділений будь жорсткий диск. Незважаючи на той факт, що в більшості сучасних файлових систем, дані, які включає завантажувач, розташовуються не в першому секторі: система XFS, наприклад, має потребу у всіх працездатних секторах диска. Тому, в разі, коли дані завантажувача стираються, файлова система повністю руйнується.

включення пристрою

У момент запуску комп'ютера, в першу чергу починає запускатися безпосередньо сам BIOS. У цей момент відбувається зчитування даних, що зберігаються в MBR першого диска. Через кілька секунд, вміст "йде" до оперативної пам'яті, де відбувається перевірка на наявність в останніх байтах шістнадцятирічних кодів 55 AA. Така перевірка полягає в необхідності ідентифікувати носії для їх подальшого завантаження. Невелика програмка, що зберігається в MBR, необхідна при збігу представлених кодів. Перша програма "чіпляє" наступну, розташовану в завантажувальному секторі активного розділу, щоб згодом активувати її, і, як наслідок, запустити ОС Windows.

У разі наявності не одного жорсткого диска, користувач може налаштувати в BIOS порядок, згідно з яким буде здійснюватися запуск конкретних жорстких дисків в момент завантаження системи. Завдяки цій особливості, нові ПК мають можливість завантажувати систему як із зовнішнього диска, так і з USB-накопичувача. Природно, змінивши порядок завантаження в BIOS, користувач може завантажувати систему за допомогою CD і DVD-дисків.



За допомогою BIOS ви можете відрегулювати порядок завантаження носіїв

Щоб в майбутньому не доводилося розбиратися з завантажувачами і думати про те, яким чином повинна здійснюватися настройка GRUB, користувач повинен знати, як встановити дві операційні системи від різних виробників таким чином, щоб вони не конфліктували між собою. Як показує практика, найрозумніше встановити спочатку завантажувач системи Linux, щоб згодом мати можливість вибору щодо запускається системи.

Однак, якщо на пристрої варто одна з останніх версій ОС від компанії Microsoft (пізніше 9x / ME), то в засобі завантаження системи можна зробити так, щоб він самостійно "турбувався" про GRUB. Величезною перевагою такої можливості є відсутність необхідності зайвий раз розбиратися з особливостями MBR. Правда, далеко не кожен сучасний користувач зможе самостійно здійснити таку маніпуляцію, так як процес цей досить складний. Все доведеться робити вручну.

Загрузчик GRUB



Загружает операционные системы



Поддержка файловых систем.

NTFS, ISO, UFS, UFS2, FFS, FAT16, FAT32, Minix, ext2/ext3/ext4, ReiserFS, JFS и XFS

Серед інших способів, налаштувати нормальну працездатність двох систем, можна вдавшись до додаткової установки GRUB. Завантажувач необхідно розташувати безпосередньо в завантажувальному сектору головного розділу і згодом відзначити його в якості активного. Цей метод також виключає необхідність використовувати MBR, проте він може спрацювати тільки з основними розділами та файловими системами, що не впливають на завантажувальний сектор розділу.

Для чого важливо створити копію MBR?

Після того, як буде прийнято рішення зробити GRUB install, користувачеві необхідно створити резервну копію MBR. Справа в тому, що неправильно внесені зміни "допоможуть покласти" як Windows, так і Linux. У цьому випадку неминуче знадобиться система Knoppix або інсталяційний DVD-диск для Windows, які допоможуть відредагувати завантажувальний сектор. Виправити свої дії можна буде в кілька разів швидше, якщо в наявність буде резервна копія MBR. Для цього слід виконати наведені нижче команди:

```
root # dd if = / dev / sda of = / boot / bootsektor.scsi bs = 1 count = 446
```

Відновлення завантажувального сектора може бути здійснено тільки за допомогою команди:

```
root # dd if = / boot / bootsektor.scsi of = / dev / sda bs = 1 count = 446
```

Однак, слід врахувати, що за допомогою цих команд користувачеві вдасться змінити лише 446 байт MBR.

Інсталяція в MBR жорсткого диска

GRUB install можна робити відразу ж, як тільки буде створено відповідний конфігураційний файл (задається за допомогою команди `/boot/grub/menu.lst`). В каталозі `grub` в обов'язковому порядку повинні бути присутні такі файли, як `stage1`, `stage2` і `*_stage1_5`. У разі їх відсутності, в цей розділ слід записати файли, що містяться в GRUB.

Подальші маніпуляції полягають в запуску завантажувача GRUB і виконанні команди `setup`. Значення `hd1, 12` повинні бути замінені на ім'я пристрою GRUB розділу диска, що містить `/boot`. Слід бути передбачливим, так як розділ `/boot` може розташовуватися не в системній папці, а в інший. `Hd0` відповідає за область зберігання завантажувального сектора основного жорсткого диска.

SUSE під час інсталяції поповнюється файлом `/etc/grub.conf.`, В якому містяться команди, які стосуються безпосередньо до GRUB install. Якщо щось пішло не так, користувач ПК може в будь-який момент процедури заново запустити установку. Допомогти в цьому може команда `grub </etc/grub.conf`.

Способи інсталяції завантажувача

Установка GRUB може бути проведена не тільки в MBR. За бажанням, завантажувач можна розташувати в будь-якому завантажувального сектору абсолютно будь-якого жорсткого диска. Незважаючи на те, що бувають ситуації, коли таке рішення виявляється невиправданим, в перерахованих нижче випадках, установка GRUB в інший розділ вельми спрощує завдання користувача.

Вдатися до GRUB install можна в тому випадку, якщо запуск завантажувача здійснюється за допомогою аналогічного пристрою Windows. Також рішення буде доцільним, якщо на комп'ютері встановлено кілька варіантів ОС Linux і раніше інстальований GRUB користувач не збирається міняти на інший завантажувач.

Переважно для оптимізації процесу вибрати системний розділ Linux. Іншими словами, якщо операційна система знаходиться в / dev / sda7, для правильної установки, користувач повинен використовувати наведені нижче команди. Основною відмінністю пива установки GRUB в MBR або системний розділ є необхідність вказати в setup замість hd0 вибраний розділ.

```
root # grub grub> root (hd1,12)
grub> setup (hd0,6) (Установка в завантажувальний сектор / dev / sda7)
grub> quit
```

Перш ніж здійснити GRUB install, слід врахувати одну особливість деяких файлових систем, так як іноді завантажувальні сектора розділу не можуть бути використані загрузчиком або іншим типовим додатком. Зокрема, однією з таких систем є XFS - інстальований в її завантажувальний сектор GRUB, просто зруйнує всю файлову систему.

Інсталяція на USB-накопичувач

Завдяки сучасним технологіям, кожен бажаючий може завантажити операційну систему за допомогою USB-накопичувача. Якщо GRUB install буде проводитися безпосередньо з зовнішнього носія інформації, а не за допомогою Windows або Linux, користувач повинен перевірити правильність розпізнавання накопичувача материнською платою.

Перш ніж почати роботу, слід відформатувати флешку одним з двох відомих способів. За допомогою першого методу вона очиститься як superfloppy, іншим - як жорсткий диск. Вибирати варіант очищення слід виходячи з особливостей BIOS.

Наступним кроком повинна стати активізація підтримки USB-накопичувача, для чого в BIOS передбачений окремий параметр. При розпізнаванні USB-накопичувача як окремого завантажувального диска, в BIOS буде змінений список, згідно з яким GRUB черзі списує дані з носіїв. В якості першого диска стане USB-накопичувач, інші жорсткі диски стануть доступні за допомогою hdl і hd2. Якщо виникне необхідність, перед GRUB install можна змінити /boot/grub/devices.map.

Якщо все зробити правильно, установка GRUB з зовнішнього носія буде більш ніж успішною. GRUB розпізнає флеш-накопичувач як `hdn + 1`, де `n` є числом, відповідним останньому внутрішньому жорсткому диску. Подальша установка GRUB в MBR за допомогою зовнішнього накопичувача інформації потребує виконання наведеної нижче команди:

```
root # grub grub> root (hd1,12)
```

```
grub> setup (hd2) (інсталяція в MBR флеш-накопичувача) grub> quit
```

Подальша перезавантаження ПК повинна допомогти GRUB з'явитися в MBR і запустити всі наявні системи, які містяться в `menu.lst`. У разі виникнення помилок, слід зайти через меню завантажувача в інтерактивний режим, натискаючи кнопку `C`. Активація команди, що складається з слова `cat` і натискання на клавішу табуляції, допомагає дізнатися користувачеві про тих назвах, які завантажувач присвоїв всім жорстких дисків. За допомогою клавіші `Esc` можна вийти назад в меню, а кнопка `E` допоможе змінити раніше написані для завантаження команди і знову їх виконати.

Якщо ОС Linux зберігається на флеш-накопичувачі або зовнішньому жорсткому диску, який можна підключити до комп'ютера виключно через USB-порт, користувачеві слід врахувати кілька важливих проблем. Справа в тому, що GRUB `install` може бути здійснений не з першого разу, найімовірніше, доведеться кілька разів змінювати налаштування BIOS і позначення приводів. Буде краще, якщо всі властивості, що зберігаються в файлі `menu.lst`, будуть налаштовуватися тільки вручну.

Крім того, у файлі `Initrd` слід зберігати всі важливі USB-модулі. При роботі з Ubuntu в розділі `/ etc / fstab` і рядку `kernel`, розташованої в `menu.lst`, потрібно вказувати не самі назви пристроїв, так як після перезавантаження системи вони можуть бути змінені, а номери `UUID`. Користувач може самостійно в `menu.lst` допомогою `uuid` вказати саме той каталог, в який згодом будуть поміщені ядро і файл `Initrd`.

Висновок

Незважаючи на складність, зробити GRUB install може кожен. Сприяння з новомодними материнськими платами передбачає просту і безпроблемну роботу будь-якої операційної системи, в тому числі і Linux, яка може бути інстальована безпосередньо на самому USB-накопичувачі. Налаштувати правильно всі параметри і створити всі умови для безперебійного існування двох різних систем нелегко, крім вільного часу і знань, користувач повинен володіти чималим досвідом, якого немає у тих, хто тільки починає знайомство з середовищем Linux.