

FlashBoot User Manual

© 2011 Mikhail Kupchik

Table of Contents

	Foreword	0
Part I	Introduction	5
1	Product Overview	5
2	Major Changes in V2	5
3	Why USB Flash Disks?	6
4	Why FlashBoot?	6
5	System Requirements	7
6	Limitations of Demo Version	8
7	Demo Version -> Full Version	8
8	Support & Feedback	8
Part II	CD to USB conversions	9
1	WinXP setup CD -> USB	9
2	WinVista setup CD -> USB	16
3	XP Recovery Console -> USB	22
4	BartPE -> USB	27
5	Generic boot CD -> USB	34
6	IsoLinux/GRUB4DOS boot CD -> USB	40
7	HDD/Floppy emulation boot CD -> USB	46
Part III	Floppy/USB to USB conversions	52
1	DOS boot disk -> USB	53
2	Syslinux/GRUB4DOS boot disk -> USB	58
3	Windows NT boot disk -> USB	64
4	Generic boot disk -> USB	70
Part IV	Builtins	76
1	Windows password editor	76
2	FreeDOS	81
3	Minimal SysLinux/GRUB4DOS	86
Part V	Mini DOS on USB	91
1	Installed DOS -> Mini DOS on USB	92
2	Windows 9x setup files -> Mini DOS on USB	98
Part VI	Setting up BIOS	103

Part VII	Decoding Status Codes	104
Part VIII	VIII Understanding Loader Messages	
Part IX	Reverting Multiformat to Normal	106
Part X	Using command-line version	113
Part XI	FlashBoot as a Portable Software	120
Part XII	License Agreement	120
	Index	0

1 Introduction

Welcome, you're browsing FlashBoot user manual for version 2.1e of the software.

This topic provides general information about FlashBoot features and ordering information.

1.1 **Product Overview**

FlashBoot is a tool to make USB disks bootable. Its primary focus is USB Flash disks, but other types of USB devices are supported as well. Making disk bootable involves formatting and copying operating system files to it. Different operating systems are supported: Windows 7/Vista, Windows XP, SysLinux-based disks, GRUB4DOS disks, Linux kernel etc.

You may create blank bootable USB flash with minimal set of system files and then manually tune it for your needs, or convert a full-featured bootable CD-ROM or floppy disk to bootable USB Flash keeping all functionality.

FlashBoot can either format physical disk or write an image file. So you may create customized USB disk manually or with another tool and use FlashBoot to create image out of it and redistribute it in local network or online.

1.2 Major Changes in V2

FlashBoot 2.0 is a major step forward towards improvement of the software and satisfying user needs. Its code was rewritten from the ground up since old version 1.4 released in 2006.

Here is a list of improvements:

- Full compatibility with Windows Vista and Windows 7, both x86 and x64 editions
- Support for USB disks larger than 4 GB (up to 2 TB)
- Support for conversion of Windows XP/Vista/7 installation CD to bootable USB disk
- Support for extraction of Windows XP recovery console to bootable USB disk
- Support for conversion of generic CD discs to bootable USB disks (operating system or boot loader of such generic CD disc must access boot device exclusively via BIOS API)
- Support for multiformat USB disks (FlashBoot 2 does not shift to user the burden of choice between "USB-ZIP" and "USB-HDD" anymore. There is one unified format, "Multiformat", and every USB disk is formatted by FlashBoot will work in all conditions: both USB-ZIP and USB-HDD. This is for sake of maximum compatibility.)
- Support for boot time mapping to either A: or C:, chosen by user at format time, not by machinespecific BIOS at boot time
- Support for explicit specification of USB disk CHS geometry by user at format time, not by machine-specific BIOS at boot time
- Greater BIOS and USB disk compatibility (explicit measures taken in order to support BIOSes which do not map USB boot disk to INT 13h devices 0 or 0x80; emulation of MBR track if it's stripped off by BIOS; FlashBoot loader always provides LBA and CHS disk access to USB disk for OS code).
- Autodetection of input CD/DVD/floppy/USB disk type. User does not have to make choice from long list of all possible disk types, it is detected automatically. But experienced user can override autodetection results and make choice from the full list.
- Less restrictive limitations of demo version (30-days bootability in V2 vs. one-time bootability in V1, 16 tries per physical USB disk in V2 vs. 10 tries per physical USB disk in V1)
- Full support for Unicode file names
- Command-line interface as alternative to GUI for users who need automation
- Improved old features (support for new SYSLINUX/ISOLINUX versions, BartPE, GRUB4DOS)

- · Support for copying RAW, fixed-size image files to/from USB disks
- Support for full erase format (wipe) of USB disks
- Improved explanation for busy USB disks (reporting the full list of conflicting processes/windows)
- Improved error handling and reporting (especially for broken "silent-forgetter" flash memory cells)
- Better UI (more details, explicit warnings) to prevent unintended formatting of wrong USB disk from multiple plugged ones or USB HDD instead of USB Flash disk.
- NT password editor is now integrated to FlashBoot, end user does not have to download any
 additional third-party files to use this feature.

All users who ordered FlashBoot V1 can upgrade to FlashBoot V2 free of charge, just by coping license key file to installation folder of FlashBoot V2.

1.3 Why USB Flash Disks?

Why do you might want to use bootable USB flash disks?

Unlike the most bootable medias, bootable USB Flash keys are very handy: compared to floppies, they have much bigger size, speed and reliability, compared to CD/DVD discs, they are random write access devices, so you can backup your data to the same media where you booted from, without need to reformat (reburn) the entire media. Again, the cost per gigabyte for them continues to cut down, unlike CD/DVD discs.

Bootable USB flash disks are especially useful with netbooks like ASUS Eee PC which does not have builtin CD/DVD drive or an opportunity to install one. On the other hand, buying external CD/DVD drive for netbook is not a truly wise choice because it will be shifted out of use just after Windows is installed, thanks to widespread use of DVD image files and modern hard disk capacities.

Bootable USB flash disks are useful as boot devices on the "big" desktop PCs too, unlike CD/DVD discs they do not have sensible surface you could scratch, thus more reliable (especially when holding your backup data). If your sysadmin at work restricts PC to not to have CD/DVD drives, you still can boot from USB flash disk. Or if your home PC has CD/DVD drive failed, you can do it too.

There are some mobility considerations as well. If your laptop has a bootable CD/DVD drive, you can't work with it for a long time: boot device is accessed quite often, and battery power is obviously not enough to supply laser for a long time.

With bootable USB Flash disk, you don't have to obey a CD/DVD size limit of 700 or 4700 MB. You can buy a big or a small USB disk depending on your needs. Just after boot, on every PC, you may save your files to the same boot device, or restore them back. There's no need to reformat (reburn) the boot disk, you just copy files and folders, and there's no need for extra hardware for such operations. Of course you may do some things you can't do under your OS: copy/modify system files (they are busy when OS is running), reinstall OS, repartition your main hard disk etc.

1.4 Why FlashBoot?

FlashBoot is designed to be compatible with all brands. It is not binded to Transend, Kingston, HP or to any other particular manufacturer of USB Flash or other types of USB disks.

FlashBoot is designed to be compatible with all types of media. It supports USB Flash, USB HDD and every possible future type of USB device conforming to "USB Mass Storage Device" requirements.

FlashBoot is a tool with wide feature list, able to install bootable builtins to USB disk, convert some bootable flopppy/CD/DVD disks to USB disks, and duplicate USB disks as well. You get all the features "in one box" if you use FlashBoot.

FlashBoot is compatible with all known BIOS bugs and weird features.

Some BIOSes does have an option to boot from USB disk as a USB-ZIP, USB-HDD, USB-Floppy or Auto. When this option is not set properly, USB disk is not bootable.	FlashBoot does not shift to user the burden of choice between "USB-ZIP" and "USB-HDD" at format time and proper BIOS setup at boot time. There is one unified format, "Multiformat", and every USB disk is formatted in such way that it will work in every environment properly, regardless of current BIOS setting
When BIOS boots from ordinary, non-FlashBoot formatted USB disk, this disk is mapped to A: or C: at BIOS discretion, quite randomly. In the majority of real world cases USB-ZIP formatted disks are mapped to A: and USB-HDD disks are mapped to C:.	All this diversity is no more a problem for FlashBoot user. When formatting USB disk, you'll be able to specify target drive letter, e.g. A: or C:, and stage2 loader of FlashBoot will take care of this problem at run time.
But there are some exceptions.	
For example, ASUS P5GDC-V BIOS in Auto mode maps 0-512Mb USB disks to A: and 512+ Mb disks to C:.	
ASUS P6T BIOS in Auto mode maps 0-1024Mb USB disks to A: and 1024+ Mb disks to C:.	
ASUS netbooks show similar behavior, but unfortunately there is no setting in their BIOS setups to override such "Auto mode" when choice between A: / C: is made by BIOS depending on disk size	
Sometimes disk CHS geometry is different from BIOS to BIOS. E.g., when formatted on the workstation, the USB disk sometimes is not bootable on the embedded hardware because of different CHS geometry on Windows and under embedded BIOS.	FlashBoot 2 allows to specify disk CHS geometry explicitly at format time, and stage2 loader will force it to predefined values at run time.
Some BIOSes cut off MBR track from USB disk when booting (especially for A:-mapped boots). E.g., they map only partition 1 of USB disk via int 13h.	FlashBoot stage2 loader emulates MBR track in such cases, thus hiding firmware diversity and avoiding OS confusion when it switches to native hardware drivers to access USB disk.
Some BIOSes provide int 13h extended API for USB disks, some do not. Some BIOSes do not provide these services in USB-ZIP mode, but provide ones in USB-HDD mode.	To unify runtime environment, FlashBoot 2 stage2 loader always provides LBA and CHS disk access to USB device it boots from.

1.5 System Requirements

Minimal system requirements for FlashBoot 2.0:

FlashBoot User Manual

8

- BIOS which supports booting from USB devices
- Operating system: Windows XP, Windows Vista, Windows 7, Windows Server 2003, Windows Server 2008, either x86 or x64 edition
- 256 Mbytes of RAM
- 30 Mbytes of disk space

You can verify conformance to these requirements with demo version of FlashBoot.

Windows 95/98/Me/NT4/2000 are not supported and not planned to be supported, as well as Itanium editions of Windows.

1.6 Limitations of Demo Version

FlashBoot is a shareware, it means that there exists a demo version (with limited functionality) available for public evaluation for free, and full version available to users who ordered it for \in 29.95.

Demo version has the following limitations:

- USB disk or image file is bootable only for 30 days since the date of formatting
- Each distinct USB disk can be formatted by FlashBoot demo version no more than 16 times

Maximum time of use of demo version is not limited.

It is recommended to try the demo version of FlashBoot before buying the full version to ensure that there are no hardware incompatibilities and product meets your particular needs.

1.7 Demo Version -> Full Version

Full version of FlashBoot can be ordered via website: http://www.prime-expert.com/flashboot/buy.php

Shortly after your order (within 2-3 minutes), you should receive e-mail message with attached file **license_key.xml**. Do not open this attacment (it's just an XML file with registration data and digital signature - you won't find it useful). Instead, right-click it and choose "Save As..." in the context menu. Choose a folder where demo version of FlashBoot is installed (usually "C:\Program Files\FlashBoot") and save **license_key.xml** there.

Demo version can be converted to full version just by adding **license_key.xml** to FlashBoot installation folder, there's no need to download anything else or reinstall FlashBoot.

If you are registered user of FlashBoot 1.x, then you are eligible for free upgrade. Find **license.xml** file in your archives, copy it to FlashBoot installation folder (usually "C:\Program Files\FlashBoot") and FlashBoot 2.0 demo version will turn into full version.

1.8 Support & Feedback

If you have any technical questions about FlashBoot, or if you need a technical support, or if you have any suggestions for further development of the project (feature request, etc), mail here: flashboot@prime-expert.com

If you have questions about orders, payment and credit card processing, please refer to share-it Customer Care Center:

http://www.shareit.com/ccc/index.html?publisherid=200000171

2 CD to USB conversions

2.1 WinXP setup CD -> USB

1) Run FlashBoot, click Next

FlashBoot 2.0	×
	Welcome to FlashBoot version 2.0
	This wizard will guide you through formatting of USB flash or USB hard disk as bootable.
	For more information, refer to http://www.prime-expert.com/flashboot/
	FlashBoot is running in DEMO mode (license keyfile not found)
	Click Next to continue
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

2) Choose CD -> USB in Main Menu



3) Choose CD/DVD drive or image file and click Next

FlashBoot 2.0					×
	$CD \rightarrow USB$: Specify s	source			
	Please specify source	CD/DVD disk or ima	age file		
	CD/DVD disk:	F:			•
	🔘 Image file:				
		,			
			< <u>B</u> ack	<u>N</u> ext >	ancel

4) Choose "Convert Windows XP/2000 installation CD" from the scenario list and click Next

FlashBoot 2.0	
	CD (F:) \rightarrow USB: Choose scenario
	What should FlashBoot do with this CD/DVD?
	Convert Windows XP/2000 installation CD Extract recovery console from Windows XP/2000 installation CD Wrap bootable CD/DVD
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

5) Choose target USB disk (flash, HDD) or image file and click Next

FlashBoot 2.0		
	CD (F:) \rightarrow USB: Sp	pecify target
	Please specify targe	et USB device or image file
	OSB device:	G: Generic Flash Disk
	🔘 Image file:	
		Size:
		< <u>Back</u> <u>N</u> ext > <u>C</u> ancel

6) Enter volume label or leave it as is and click Next

FlashBoot 2.0		×
	CD (F:) \rightarrow USB (G): Formatting options
	Formatting options	:
	Volume label:	XP_SETUP
XXII	Filesystem:	Auto 🔹
	Set advance	d options
	Allowed cluste	r sizes: 🗸 0.5K 🗸 1K 🗸 2K 🗸 4K 🗸 8K 🗸 16K 🗸 32K
	Disk geometry	: Heads: 255 Sectors per track: 63
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

7) Check summary information and click Next

FlashBoot 2.0	
	Summary
	Scenario: Convert Windows XP installation CD/DVD Data source: CD/DVD disk F: Target: USB storage device G: Generic Flash Disk Filesystem: Auto Volume label: XP_SETUP Disk CHS geometry: Heads=255 SectorsPerTrack=63 Allowed cluster sizes: 0.5k 1k 2k 4k 8k 16k 32k
	< <u>B</u> ack <u>Format Now!</u> <u>Cancel</u>

8) Wait for process completion

(24%) FlashBoot is creating bootable USB disk	
Formatting now	
	Cancel
Log:	
Mounting ISO9660 filesystem on disk F: Verifying read-only ISO 9660 file system in disk F: Formatting VFAT filesystem on target USB disk Creating helper image file (wntsetup.img)	•
	~

9) USB disk is ready for use. Click OK to exit



WARNING: Installation of Windows XP may look like halted at this point:



If you see that screen does not update and installation looks like halted for a long time at this point, **please be patient and don't panic**. Wait at least for a 45 minutes before aborting Windows XP installation.

Click here to view gallery of screenshots made during installation of Windows XP from FlashBootformatted USB disk to netbook. Windows XP was installed to the newly-created disk C: (which took 20 Gb out of 64 Gb total SSD space). Final filesystem of disk C: is NTFS.

2.2 WinVista setup CD -> USB

1) Run FlashBoot, click Next



2) Choose CD -> USB in Main Menu



3) Choose CD/DVD drive or image file and click Next

FlashBoot 2.0			×
	$CD \rightarrow USB$: Specify source		
	Please specify source CD/DVD disk or in	nage file	
	CD/DVD disk: F:		
	© Image file:]
		< <u>B</u> ack <u>N</u> ext > <u>C</u> a	incel

4) Choose "Convert Windows Vista installation CD" from the scenario list and click Next

FlashBoot 2.0	
	CD (F:) \rightarrow USB: Choose scenario
	What should FlashBoot do with this CD/DVD?
	Convert Windows Vista installation DVD Wrap bootable CD/DVD Ignore CD/DVD autodetection results (show all options)
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

5) Choose target USB disk (flash, HDD) or image file and click Next

FlashBoot 2.0		
	CD (F:) → USB: Sp	ecify target
	Please specify targe	et USB device or image file
	USB device:	G: JetFlash TS8GJFV60
	🔘 Image file:	
		Size:
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

6) Enter volume label or leave it as is and click Next

FlashBoot 2.0		
	CD (F:) \rightarrow USB (G	:): Formatting options
	Formatting options	Σ
	Volume label:	VISTA_SETUP
XNN	Filesystem:	Auto 🔹
	Set advance	d options
	Allowed cluste	er sizes: 🗹 0.5K 🖤 1K 🖤 2K 🖤 4K 🖤 8K 🖤 16K 🖤 32K
	Disk geometry	: Heads: 255 Sectors per track: 63
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

7) Check summary information and click Next

FlashBoot 2.0	
	Summary
	Scenario: Convert Windows Vista installation CD/DVD Data source: CD/DVD disk F:
	Target: USB storage device G: JetFlash TS8GJFV60 Filesystem: Auto Volume label: VISTA_SETUP Disk CHS geometry: Heads=255 SectorsPerTrack=63 Allowed cluster sizes: 0.5k 1k 2k 4k 8k 16k 32k
	< <u>B</u> ack <u>Format Now!</u> <u>C</u> ancel

8) Wait for process completion



9) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully	×
Completed successfully. Click OK to exit	
Log:	
SUPPORT\MIGWIZ\EN-US\MIGSETUP.EXE.MUI Copying file SUPPORT\MIGWIZ\EN-US\MIGUIRES.DLL.MUI to SUPPORT\MIGWIZ\EN-US\MIGUIRES.DLL.MUI Copying file SUPPORT\MIGWIZ\EN-US\MIGWIZ.EXE.MUI to SUPPORT\MIGWIZ\EN-US\MIGWIZ.EXE.MUI Copying file SUPPORT\MIGWIZ\EN-US\SPWIZRES.DLL.MUI to SUPPORT\MIGWIZ\EN-US\SPWIZRES.DLL.MUI Copying file SUPPORT\TOOLS\GBUNICNV.EXE to SUPPORT\TOOLS\GBUNICNV.EXE Copying file UPGRADE\NETFX to UPGRADE\NETFX Copying file UPGRADE\NETFX\NETFX.MSI to UPGRADE\NETFX\NETFX.MSI Copying file UPGRADE\NETFX\NETFX.MSF to UPGRADE\NETFX\NETFX.MSF Copying file UPGRADE\NETFX\NETFX1.CAB to UPGRADE\NETFX\NETFX1.CAB Copying file UPGRADE\NETFX\NETFXUPDATE.EXE to UPGRADE\NETFX\NETFXUPDATE.EXE Unmounting VFAT filesystem in partition #1 of physical disk #5 (G:) Done	^
Save log OK	

2.3 XP Recovery Console -> USB

1) Run FlashBoot, click Next



2) Choose CD -> USB in Main Menu



3) Choose CD/DVD drive or image file and click Next

FlashBoot 2.0		
	$CD \rightarrow USB: Specify$	source
	Please specify source	CD/DVD disk or image file
	CD/DVD disk:	F:
	Mage file:	
	<u> </u>	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

4) Choose "Extract recovery console from Windows XP/2000 installation CD" from the scenario list and

click Next

FlashBoot 2.0	
	CD (F:) \rightarrow USB: Choose scenario
	What should FlashBoot do with this CD/DVD?
	Convert Windows XP/2000 installation CD
	Extract recovery console from Windows XP/2000 installation CD
	Ignore CD/DVD autodetection results (show all options)
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

5) Choose target USB disk (flash, HDD) or image file and click Next

FlashBoot 2.0		×
	CD (F:) \rightarrow USB: Sp	ecify target
	Please specify targe	et USB device or image file
	OSB device:	G: Generic Flash Disk
X	🔘 Image file:	
		Size:
		< Back Next > Cancel

FlashBoot 2.0		×
	CD (F:) \rightarrow USB (G	:): Formatting options
	Formatting option	s:
	Volume label:	XP_CONSOLE
	Filesystem:	Auto
	Set advance	ed options
	Disk geometry	r: Heads: 255 Sectors per track: 63
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

6) Enter volume label or leave it as is and click Next

7) Check summary information and click Next

FlashBoot 2.0	
	Summary
	Scenario: Extract Recovery Console from Windows XP installation CD/DVD Data source: CD/DVD disk F: Target: USB storage device G: Generic Flash Disk Filesystem: Auto Volume label: XP_CONSOLE Disk CHS geometry: Heads=255 SectorsPerTrack=63 Allowed cluster sizes: 0.5k 1k 2k 4k 8k 16k 32k
	< <u>B</u> ack <u>Format Now!</u> <u>C</u> ancel

8) Wait for process completion

(84%) FlashBoot is creating bootable USB disk	
Formatting now	
	Cancei
Log:	
Mounting ISO9660 filesystem on disk F: Verifying read-only ISO 9660 file system in disk F: Scanning read-only ISO 9660 file system in disk F: Formatting VFAT filesystem on target USB disk Copying files Generating ISO9660 filesystem (xprc.iso)	
	~

9) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully		×
Completed successfully. Click OK to exit		
Log:		
Mounting ISO9660 filesystem on disk F: Verifying read-only ISO 9660 file system in disk F: Scanning read-only ISO 9660 file system in disk F: Formatting VFAT filesystem on target USB disk Copying files Generating ISO9660 filesystem (xprc.iso) Unmounting VFAT filesystem in partition #1 of physical disk #5 (G:) Done		*
Save log	ОК	

2.4 BartPE -> USB

BartPE is a third-party tool which allows functionally-reduced Windows XP to be run from CD/DVD disc without installation to HDD. BartPE is a freeware and it can be downloaded from here.

FlashBoot can convert ISO file produced by BartPE into bootable USB disk.

1) Run FlashBoot, click Next



2) Choose CD -> USB in Main Menu



3) Choose CD/DVD drive or image file and click Next

FlashBoot 2.0		
	$CD \rightarrow USB: Specify states the state of the state of the states of the s$	source
	Please specify source	e CD/DVD disk or image file
	CD/DVD disk:	F : ▼
	Image file:	
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

4) Choose "Convert Windows PE 1.x or Windows-XP derived BartPE" from the scenario list and click Next



FlashBoot 2.0		×
	CD (F:) → USB: Sp	ecify target
	Please specify targe	et USB device or image file
	OSB device:	G: Generic Flash Disk
	🔘 Image file:	
		Size;
MAT		
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

5) Choose target USB disk (flash, HDD) or image file and click Next

6) Enter volume label or leave it as is and click Next

FlashBoot 2.0		
	CD (F:) → USB (G	:): Formatting options
	Formatting option	s:
	Volume label:	BARTPE
XNN	Filesystem:	Auto
4	Set advance	ed options
	Allowed cluste	er sizes: 📝 0.5K 📝 1K 📝 2K 📝 4K 📝 8K 📝 16K 📝 32K
	Disk geometry	y: Heads: 255 Sectors per track: 63
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

FlashBoot 2.0	
	Summary
	Scenario: Convert BartPE or Windows PE installation CD/DVD Data source: CD/DVD disk F:
	Target: USB storage device G: Generic Flash Disk Filesystem: Auto Volume label: BARTPE Disk CHS geometry: Heads=255 SectorsPerTrack=63 Allowed cluster sizes: 0.5k 1k 2k 4k 8k 16k 32k
	< <u>B</u> ack <u>Format Now!</u> <u>C</u> ancel

7) Check summary information and click Next

8) Wait for process completion



9) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully	
Completed successfully. Click OK to exit	
Log:	
Mounting ISO9660 filesystem on disk F: Verifying read-only ISO 9660 file system in disk F: Scanning read-only ISO 9660 file system in disk F: Formatting VFAT filesystem on target USB disk Copying files Generating ISO9660 filesystem (winpe.iso) Unmounting VFAT filesystem in partition #1 of physical disk #5 (G:) Done	
Save log	ОК

WARNING: BartPE will not boot from USB if size of source ISO image file is larger than 500 Mb



2.5 Generic boot CD -> USB

FlashBoot can convert generic bootable CD to bootable USB disk.

This won't work with every possible CD, because FlashBoot loader will only remap INT 13h CD access to USB. If CD contains an operating system which uses custom driver and direct I/O to access boot CD, FlashBoot won't be able to redirect such I/O to USB.

Let's take Acronis TrueImage as an example of generic boot CD that can be converted to bootable USB disk by FlashBoot.

1) Run FlashBoot, click Next

FlashBoot 2.0	
	Welcome to FlashBoot version 2.0
	This wizard will guide you through formatting of USB flash or USB hard disk as bootable.
	For more information, refer to http://www.prime-expert.com/flashboot/
	FlashBoot is running in DEMO mode (license keyfile not found)
724	
	Click Next to continue
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

2) Choose CD -> USB in Main Menu



3) Choose CD/DVD drive or image file and click Next:

FlashBoot 2.0		×
	$CD \rightarrow USB$: Specify s	ource
	Please specify source	CD/DVD disk or image file
	CD/DVD disk:	· · · · · · · · · · · · · · · · · · ·
	Image file:	C:\Downloads\True Image BootCD.iso
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

4) Choose "Wrap bootable CD/DVD" from the scenario list and click Next

FlashBoot 2.0	
	CD (image file) $ ightarrow$ USB: Choose scenario
	What should FlashBoot do with this CD/DVD?
	Wrap bootable CD/DVD
	Ignore CD/DVD autodetection results (show all options)
MA	
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

5) Choose target USB disk (flash, HDD) or image file and click Next
| FlashBoot 2.0 | | × |
|---------------|----------------------|--|
| | CD (image file) → | USB: Specify target |
| | Please specify targe | et USB device or image file |
| | OSB device: | G: Generic Flash Disk |
| | 🔘 Image file: | |
| 724 | | Size: |
| | | |
| | | |
| | | |
| | | |
| | | < <u>B</u> ack <u>N</u> ext > <u>C</u> ancel |

6) Enter volume label or leave it as is and click Next

FlashBoot 2.0		×
	CD (image file) —	• USB (G:): Formatting options
	Formatting option	IS:
	Volume label:	TRUEIMAGE
XVI	Filesystem:	Auto Map to (at boot time): A:
\mathcal{A}	Set advance	ed options
	Allowed clust	er sizes: 🔽 0.5K 🗹 1K 🗹 2K 🗹 4K 🗹 8K 🗹 16K ✔ 32K
	Disk geometr	y: Heads: 255 Sectors per track: 63
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

7) Check summary information and click Next



8) Wait for process completion

(81%) FlashBoot is creating bootable USB disk	
Formatting now	
	Cancel
Log:	
Formatting VFAT filesystem on target USB disk Writing configuration file to target filesystem Writing image of CD/DVD (boot.iso) to target filesystem	

9) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully		×
Completed successfully. Click OK to exit		
Log:		
Formatting VFAT filesystem on target USB disk Writing configuration file to target filesystem Writing image of CD/DVD (boot.iso) to target filesystem Unmounting VFAT filesystem in partition #1 of physical disk #1 (G:) Done		
Save log	OK	

2.6 IsoLinux/GRUB4DOS boot CD -> USB

Let's take EBCD 1.x as an example to see how Isolinux-based boot CD can be converted to bootable USB disk by FlashBoot. EBCD is a CD-ROM for booting PC and recovering data in emergency situations, shareware.

1) Run FlashBoot, click Next

FlashBoot 2.0	
	Welcome to FlashBoot version 2.0
	This wizard will guide you through formatting of USB flash or USB hard disk as bootable.
	For more information, refer to http://www.prime-expert.com/flashboot/
	FlashBoot is running in DEMO mode (license keyfile not found)
	Click Next to continue.
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

2) Choose CD -> USB in Main Menu



3) Choose CD/DVD drive or image file and click Next

FlashBoot 2.0		
	$CD \rightarrow USB$: Specify s	ource
	Please specify source	CD/DVD disk or image file
	CD/DVD disk:	· · · · · · · · · · · · · · · · · · ·
	Image file:	C:\Downloads\ebcd-1.1k-demo.iso
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

4) Choose "Convert IsoLinux-based bootable CD/DVD" or "Convert GRUB4DOS-based bootable CD/ DVD" from the scenario list and click Next

FlashBoot 2.0	
	CD (image file) $ ightarrow$ USB: Choose scenario
	What should FlashBoot do with this CD/DVD?
	Convert IsoLinux-based bootable CD/DVD Wrap bootable CD/DVD
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

FlashBoot 2.0		×
	CD (image file) $ ightarrow$	USB: Specify target
	Please specify targe	et USB device or image file
	OSB device:	G: Generic Flash Disk
XDI	🔘 Image file:	
1		Size:
XAP		
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

5) Choose target USB disk (flash, HDD) or image file and click Next

6) Enter volume label or leave it as is and click Next

FlashBoot 2.0		
	CD (image file) —	→ USB (G:): Formatting options
	Formatting option	S:
	Volume label:	EBCD
XVI	Filesystem:	Auto Map to (at boot time): A:
\mathcal{A}	Set advance	ed options
	Allowed clust	er sizes: 🔽 0.5K 🔽 1K 🗹 2K 🗹 4K 📝 8K 📝 16K 🗹 32K
	Disk geometr	y: Heads: 255 Sectors per track: 63
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

FlashBoot 2.0	
	Summary
	Scenario: Convert ISOLinux-based bootable CD/DVD
X	Data source: CD/DVD image file C:\Downloads\ebcd-1.1k-demo.iso
	Target: USB storage device G: Generic Flash Disk Filesvstem: Auto
	Volume label: EBCD Map to (at boot time): A: Disk CHS geometry: Heads=255 SectorsPerTrack=63
	Allowed cluster sizes: 0.5k 1k 2k 4k 8k 16k 32k
	< <u>B</u> ack <u>F</u> ormat Now! <u>C</u> ancel

7) Check summary information and click Next

8) Wait for process completion

(13%) FlashBoot	is creating bootable USB disk	
	Formatting now	
		Cancel
Log:		
Mounting ISO9660 Verifying read-only Formatting VFAT fi	filesystem on file C:\Downloads\ebcd-1.1k-d ISO 9660 file system in file C:\Downloads\eb lesystem on target USB disk	emo.iso ocd-1.1k-demo.iso

9) USB disk is ready for use. Click OK to exit



2.7 HDD/Floppy emulation boot CD -> USB

Let's take MemTest86+ as an example to see how floppy-emulation boot CD can be converted to bootable USB disk by FlashBoot. MemTest86+ is a third-party memory diagnostic utility, freeware.

1) Run FlashBoot, click Next

FlashBoot 2.0	
	Welcome to FlashBoot version 2.0
	This wizard will guide you through formatting of USB flash or USB hard disk as bootable.
	For more information, refer to http://www.prime-expert.com/flashboot/
	FlashBoot is running in DEMO mode (license keyfile not found)
XA	Click Next to continue.
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

2) Choose CD -> USB in Main Menu



3) Choose CD/DVD drive or image file and click Next

FlashBoot 2.0			×
	$CD \rightarrow USB$: Specify s	ource	
	Please specify source	CD/DVD disk or image file	
	CD/DVD disk:		-
X	Image file:	C:\Downloads\memtest\memtest86+-4.00.iso	
		< <u>Back</u> Next > Cance	ei

4) Choose "Convert Floppy-emulation based bootable CD/DVD" from the scenario list and click Next



5) Choose target USB disk (flash, HDD) or image file and click Next

FlashBoot 2.0		×
	CD (image file) →	USB: Specify target
	Please specify targe	et USB device or image file
	OSB device:	G: Generic Flash Disk
	Image file:	
		Size:
		< Back Next > Cancel

6) Enter volume label or leave it as is and click Next

FlashBoot 2.0		×			
	CD (image file) \rightarrow USB (G:): Formatting options				
	Formatting options:				
	Volume label:	MEMTEST			
XVI	Filesystem:	Auto Map to (at boot time): A:			
	Set advance	ed options			
	Allowed clust	er sizes: 🗸 0.5K 🗸 1K 🗸 2K 🗸 4K 🗸 8K 🗸 16K 🗸 32K			
	Disk geometr	y: Heads: 255 Sectors per track: 63			
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel			

7) Check summary information and click Next



8) Wait for process completion

(91%) FlashBoot is creating bootable USB disk	
Formatting now	
	Cancel
Log:	
Mounting ISO9660 filesystem on file C:\Downloads\memtest\memtest8 Verifying read-only ISO 9660 file system in file C:\Downloads\memtest4 4.00.iso Formatting VFAT filesystem on target USB disk Copying files Writing boot image file (boot.img) Unmounting VFAT filesystem in partition #1 of physical disk #5 (G:)	i6+-4.00.iso ,memtest86+-

9) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully		×
Completed successfully. Click OK to exit		
Log:		
Mounting ISO9660 filesystem on file C:\Downloads\memtest\memtest86 Verifying read-only ISO 9660 file system in file C:\Downloads\memtest\m 4.00.iso Formatting VFAT filesystem on target USB disk Copying files Writing boot image file (boot.img) Unmounting VFAT filesystem in partition #1 of physical disk #5 (G:) Done	+-4.00.iso nemtest86+	
Save log	ОК	-

3 Floppy/USB to USB conversions

3.1 DOS boot disk -> USB

1) Run FlashBoot, click Next



2) Choose Floppy -> USB or USB -> USB in Main Menu



3) Choose floppy disk or image file and click Next

FlashBoot 2.0		
	Floppy \rightarrow USB: Spe	cify source
	Please specify sourc	e floppy disk or image file
	Floppy disk:	· · · · · · · · · · · · · · · · · · ·
	Image file:	C:\Downloads\Boot floppies\msdos622.img
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

4) Choose "Convert MS-DOS bootable floppy disk" from the scenario list and click Next

FlashBoot 2.0	
	Floppy (image file) $ ightarrow$ USB: Choose scenario
	What should FlashBoot do with this floppy disk?
	Convert MS-DOS bootable floppy disk Wrap bootable floppy disk
	Ignore floppy disk autodetection results (show all options)
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

5) Choose target USB disk (flash, HDD) or image file and click Next

FlashBoot 2.0		
	Floppy (image file)) \rightarrow USB: Specify target
	Please specify targe	et USB device or image file
	OSB device:	G: Generic Flash Disk
X	🔘 Image file:	
		Size:
		< Back Next > Cancel

6) Enter volume label or leave it as is and click Next

FlashBoot 2.0		×
	Floppy (image file	ile) \rightarrow USB (G:): Formatting options
	Formatting options	ins:
	Volume label:	MSDOS622
XVI	Filesystem:	Auto Map to (at boot time): A:
4	Set advance	ced options
	Allowed cluste	ster sizes; 🔽 0.5K 🔽 1K 📝 2K 📝 4K 📝 8K 📝 16K 📝 32K
	Disk geometry	try: Heads: 255 Sectors per track: 63
MAN T		
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

7) Check summary information and click Next

FlashBoot 2.0	
	Summary
	Scenario: Copy DOS-based bootable disk Data source: Floppy image file D:\SHARED\User software\BootCD\(Boot floppy)\msdos622.img Target: USB storage device G: Generic Flash Disk Filesystem: Auto Volume label: MSDOS622 Map to (at boot time): A: Disk CHS geometry: Heads=255 SectorsPerTrack=63 Allowed cluster sizes: 0.5k 1k 2k 4k 8k 16k 32k
NA P	
	< <u>B</u> ack <u>Format Now!</u> <u>C</u> ancel

8) Wait for process completion

(15%) FlashBoot is creating bootable USB disk	
Formatting now	
	Cancel
Log:	
Mounting VFAT filesystem on file D:\SHARED\User software\BootCD\(E \msdos622.img Verifying VFAT filesystem in file D:\SHARED\User software\BootCD\(Bo \msdos622.img Verifying compliance and formatting VFAT filesystem on target USB dis	Boot floppy)

9) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully	
Completed successfully. Click OK to exit	
Log:	
Copying file BOOT\ASPI4DOS.SYS to BOOT\ASPI4DOS.SYS Copying file BOOT\ECSCDIDE.SYS to BOOT\ECSCDIDE.SYS Copying file BOOT\BTCDROM.SYS to BOOT\ECSCDIDE.SYS Copying file BOOT\KEYRUS.COM to BOOT\KEYRUS.COM Copying file BOOT\ASPICD.SYS to BOOT\ASPICD.SYS Copying file SHELLS\HIEW.EXE to SHELLS\HIEW.EXE Copying file SHELLS\VCEDIT.EXT to SHELLS\VCEDIT.EXT Copying file SHELLS\VCEDIT.HIP to SHELLS\VCEDIT.EXT Copying file SHELLS\VC.COM to SHELLS\VC.COM Copying file SHELLS\VC.COM to SHELLS\VC.COM Copying file SHELLS\VC.VIEW.EXT to SHELLS\VC.VIEW.EXT Copying file SHELLS\VC.VIEW.HIP to SHELLS\VC.VIEW.EXT Copying file SHELLS\HIEW.VMM to SHELLS\HIEW.VMM Copying file SHELLS\HIEW.VMM to SHELLS\HIEW.VMM Copying file SHELLS\VC.INI to SHELLS\VC.INI Unmounting VFAT filesystem in partition #1 of physical disk #5 (G:) Done	
Save log	ОК

3.2 Syslinux/GRUB4DOS boot disk -> USB

Let's take recovery floppy disk for RedHat Linux as an example to see how Syslinux-based boot CD can be converted to bootable USB disk by FlashBoot.

1) Run FlashBoot, click Next

FlashBoot 2.0	
	Welcome to FlashBoot version 2.0
	This wizard will guide you through formatting of USB flash or USB hard disk as bootable.
	For more information, refer to http://www.prime-expert.com/flashboot/
	FlashBoot is running in DEMO mode (license keyfile not found)
	Click Next to continue.
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

2) Choose Floppy -> USB or USB -> USB in Main Menu



3) Choose floppy disk or image file and click Next

FlashBoot 2.0		×
	Floppy \rightarrow USB: Spe	cify source
	Please specify source	e floppy disk or image file
	C Floppy disk:	•
	Image file:	C:\Downloads\Boot floppies\syslinux_floppy.img
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

4) Choose "Convert Syslinux-based bootable floppy disk" or "Convert GRUB4DOS-based bootable floppy disk" from the scenario list and click Next



FlashBoot 2.0		×
	Floppy (image file)) $ ightarrow$ USB: Specify target
	Please specify targe	et USB device or image file
	OSB device:	G: Generic Flash Disk
	🔘 Image file:	
		Size:
MAT		
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

5) Choose target USB disk (flash, HDD) or image file and click Next

6) Enter volume label or leave it as is and click Next

FlashBoot 2.0	Floppy (image file	e) → USB (G:): Formatting options
	Formatting option Volume label: Filesystem:	s: SYSLINUX Auto Map to (at boot time): A:
	Set advance Allowed clust Disk geometr	er sizes: 🔽 0.5K 🔍 1K 🔍 2K 🔍 4K 🔍 8K 🔍 16K 🔍 32K y: Heads: 255 Sectors per track: 63
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

FlashBoot 2.0	
	Summary
	Scenario: Copy SYSLinux-based bootable disk
X	Data source: Floppy image file C:\Downloads\Boot floppies\syslinux_floppy.img
	Target: USB storage device G: Generic Flash Disk Filesystem: Auto
	Volume label: SYSLINUX Map to (at boot time): A: Disk CHS geometry: Heads=255 SectorsPerTrack=63 Allowed cluster sizes: 0.5k 1k 2k 4k 8k 16k 32k
XAT	
	< <u>B</u> ack <u>Format Now!</u> <u>C</u> ancel

7) Check summary information and click Next

8) Wait for process completion

Formatting now	Cancel
Log: Mounting VFAT filesystem on file C:\Downloads\Boot floppies\syslinux_flo	Cancel
Log: Mounting VFAT filesystem on file C:\Downloads\Boot floppies\syslinux_flo	Cancel
Log: Mounting VFAT filesystem on file C:\Downloads\Boot floppies\syslinux_flo	
Mounting VFAT filesystem on file C:\Downloads\Boot floppies\syslinux_flo	
Verifying VFAT filesystem in file C:\Downloads\Boot floppies\syslinux_flop Formatting VFAT filesystem on target USB disk	ppy.img

9) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully
Completed successfully. Click OK to exit
Log:
Mounting VFAT filesystem on file C:\Downloads\Boot floppies\syslinux_floppy.img Verifying VFAT filesystem in file C:\Downloads\Boot floppies\syslinux_floppy.img Formatting VFAT filesystem on target USB disk Copying files Copying file RESCUE.MSG to RESCUE.MSG Copying file SYSLINUX.CFG to SYSLINUX.CFG Copying file SYSLINUX.CFG to SYSLINUX.CFG Copying file PARAM.MSG to PARAM.MSG Copying file BOOT.MSG to BOOT.MSG Copying file BOOT.MSG to BOOT.MSG Copying file GENERAL.MSG to GENERAL.MSG Copying file GENERAL.MSG to GENERAL.MSG Copying file EXPERT.MSG to EXPERT.MSG Unmounting VFAT filesystem in partition #1 of physical disk #5 (G:) Done
Save log OK

3.3 Windows NT boot disk -> USB

A floppy disk containing NTLDR, NTDETECT.COM and BOOT.INI is very useful to boot Windows 2000/ XP when these files were accidently deleted from disk C: or some changes were made to C:\BOOT.INI which rendered system unbootable.

FlashBoot can convert such bootable floppy disk into bootable USB disk.

1) Run FlashBoot, click Next

FlashBoot 2.0	
	Welcome to FlashBoot version 2.0
	This wizard will guide you through formatting of USB flash or USB hard disk as bootable.
	For more information, refer to http://www.prime-expert.com/flashboot/
	FlashBoot is running in DEMO mode (license keyfile not found)
XA	Click Next to continue.
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

2) Choose Floppy -> USB or USB -> USB in Main Menu



3) Choose floppy disk or image file and click Next

FlashBoot 2.0			×
	Floppy \rightarrow USB: Spe	cify source	
	Please specify source	e floppy disk or image file	
	C Floppy disk:		-
	Image file:	C:\Downloads\Boot floppies\ntldr_floppy.img	
M			
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel	

4) Choose "Convert NT4.x or NT5.x (NTLDR-based) bootable floppy disk" and click Next



5) Choose target USB disk (flash, HDD) or image file and click Next

FlashBoot 2.0		×
	Floppy (image file) $ ightarrow$ USB: Specify target
	Please specify targe	et USB device or image file
	OSB device:	G: Generic Flash Disk
XDIN	🔘 Image file:	
		Size:
HA T		
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

6) Enter volume label or leave it as is and click Next

FlashBoot 2.0		×
	Floppy (image fil	le) $ ightarrow$ USB (G:): Formatting options
	Formatting option	ns:
	Volume label:	NTLDR
XNI	Filesystem:	Auto Map to (at boot time): A:
	Set advance	ed options
	Allowed clust	ter sizes: 🗸 0.5K 🗸 1K 🗸 2K 🗸 4K 🗸 8K 🗸 16K 🗸 32K
	Disk geometr	ry: Heads: 255 Sectors per track: 63
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

7) Check summary information and click Next



8) Wait for process completion

Formatting now	
	Cancel
Log:	
Mounting VFAT filesystem on file C:\Downloads\Boot floppies\ntldr_flo Verifying VFAT filesystem in file C:\Downloads\Boot floppies\ntldr_flop Formatting VFAT filesystem on target USB disk	oppy.img

9) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully	
Completed successfully. Click OK to exit	
Log:	
Mounting VFAT filesystem on file C:\Downloads\Boot floppies\ntldr_floppy Verifying VFAT filesystem in file C:\Downloads\Boot floppies\ntldr_floppy Formatting VFAT filesystem on target USB disk Copying files Copying file BOOT.INI to BOOT.INI Unmounting VFAT filesystem in partition #1 of physical disk #5 (G:) Done	y.img
Save log	ОК

3.4 Generic boot disk -> USB

FlashBoot can convert generic bootable floppy disk to USB boot disk.

At boot time, image of floppy disk is loaded into memory from USB boot disk, and all changes to this image are discarded at the next reboot.

Let's convert floppy disk image of Windows 98 EBD (Emergency Boot Disk) for example.

1) Run FlashBoot, click Next

FlashBoot 2.0	
	Welcome to FlashBoot version 2.0
	This wizard will guide you through formatting of USB flash or USB hard disk as bootable.
	For more information, refer to http://www.prime-expert.com/flashboot/
	FlashBoot is running in DEMO mode (license keyfile not found)
	Click Next to continue
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

2) Choose Floppy -> USB in Main Menu



3) Choose floppy disk or image file and click Next

FlashBoot 2.0		×
	Floppy \rightarrow USB: Spe	cify source
	Please specify source	e floppy disk or image file
	C Floppy disk:	· · · · · · · · · · · · · · · · · · ·
	Image file:	C:\Downloads\Boot floppies\win98ebd.img
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

4) Choose "Wrap bootable floppy disk" and click Next

FlashBoot 2.0	
	Floppy (image file) $ ightarrow$ USB: Choose scenario
	What should FlashBoot do with this floppy disk?
	Convert MS-DOS bootable floppy disk
	Wrap bootable floppy disk
	Ignore floppy disk autodetection results (show all options)
XAT	
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

5) Choose target USB disk (flash, HDD) or image file and click Next
| FlashBoot 2.0 | | × |
|---------------|----------------------|--|
| | Floppy (image file) |) \rightarrow USB: Specify target |
| | Please specify targe | et USB device or image file |
| | OSB device: | G: Generic Flash Disk |
| | 🔘 Image file: | |
| | | Size: |
| | | |
| | | |
| | | |
| MA | | |
| | | < <u>B</u> ack <u>N</u> ext > <u>C</u> ancel |

6) Enter volume label or leave it as is and click Next

FlashBoot 2.0		×
	Floppy (image file	e) \rightarrow USB (G:): Formatting options
	Formatting option	5:
	Volume label:	WIN98EBD
XNN	Filesystem:	Auto Map to (at boot time): A:
\mathcal{A}	Set advance	ed options
	Allowed clust	er sizes: 🗸 0.5K 🗸 1K 🗸 2K 🗸 4K 🗸 8K 🗸 16K 🗸 32K
	Disk geometry	r: Heads: 255 Sectors per track: 63
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

7) Check summary information and click Next



8) Wait for process completion

(87%) FlashBoot is creating bootable USB disk		X
Formatting now		
	Cancel	
Log:		
Formatting VFAT filesystem on target USB disk Copying files Writing image of floppy disk to target filesystem Unmounting VFAT filesystem in partition #1 of physical disk #5 (G:)		•

9) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully		×
Completed successfully. Click OK to exit		
Log:		
Formatting VFAT filesystem on target USB disk Copying files Writing image of floppy disk to target filesystem Unmounting VFAT filesystem in partition #1 of physical disk #5 (G:) Done		•
Save log	OK	

4 Builtins

4.1 Windows password editor

FlashBoot contains builtin password editor, extracted from EBCD. So FlashBoot users don't have to buy a copy of EBCD if they need to edit Windows NT/2000/XP/Vista/7 passwords.

1) Run FlashBoot, click Next



2) Choose MiniOS in Main Menu

FlashBoot 2.0	
	FlashBoot Main Menu
	• CD \rightarrow USB Convert bootable CD/DVD disk to bootable USB device
X	◆ Floppy → USB Convert bootable floppy disk to bootable USB device
	• USB \rightarrow USB Copy bootable USB device to bootable USB device
	MiniOS Install mini OS on bootable USB device
	Non-boot Format USB device as non-bootable
	< <u>B</u> ack <u>C</u> ancel

3) Choose "Windows NT password editor" in MiniOS Menu and click Next



4) Choose target USB disk (flash, HDD) or image file and click Next

FlashBoot 2.0				×
	NT Password Edito	or $ ightarrow$ USB: Specify target		
	Please specify targe	t USB device or image file		
	OSB device:	G: Generic Flash Disk		•
	🔘 Image file:			
		Size:		
MA				
		< <u>B</u> a	ck <u>N</u> ext >	<u>C</u> ancel

5) Enter volume label or leave it as is and click Next

FlashBoot 2.0				×
	NT Password Edit	tor $ ightarrow$ USB (G:): Format	ting options	
	Formatting option	s:		
	Volume label:	NTPWEDIT		
XNI	Filesystem:	Auto 🔹	Map to (at boot time):	A: •
	Set advance	ed options		
	Allowed clust	er sizes: 0.5K 1	LK 📝 2K 📝 4K 📝 8K	✓ 16K ✓ 32K
	Disk geometr	y: Heads: 255	Sectors per tr	ack: 63
Contraction of the local division of the loc				
			< <u>B</u> ack <u>N</u> ext >	Cancel

6) Check summary information and click Next

FlashBoot 2.0	
	Summary
	Scenario: Install NT password wizard Data source: FlashBoot builtin Target: USB storage device G: Generic Flash Disk Filesystem: Auto Volume label: NTPWEDIT Map to (at boot time): A: Disk CHS geometry: Heads=255 SectorsPerTrack=63 Allowed cluster sizes: 0.5k 1k 2k 4k 8k 16k 32k
	< <u>B</u> ack <u>Format Now!</u> <u>C</u> ancel

7) Wait for process completion

(59%) FlashBoot is creating bootable USB disk	
Formatting now	
	Cancel
Log:	
Formatting VFAT filesystem on target USB disk Copying files	
	~

8) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully	
Completed successfully. Click OK to exit	
Log:	
Formatting VFAT filesystem on target USB disk Copying files Unmounting VFAT filesystem in partition #1 of physical disk #5 (G:) Done	
Save log	ОК

More information about using password editor: http://www.prime-expert.com/ebcd/user_guide/ebcd_boot_passwd.pdf

4.2 FreeDOS

FlashBoot can install minimalistic DOS on USB disk and make it bootable.

There are no input disk(s) required, DOS is FlashBoot-builtin. We use FreeDOS because of licensing restrictions of MS-DOS and PC-DOS. Minimalistic DOS created by FlashBoot also contains Volkov Commander - a file manager for DOS, formerly shareware, now defunct.

You can use this option to quick-test FlashBoot, and also as a base for more complex custom bootable USB disk - for example when you need to upgrade firmware of your motherboard and have all necessary files except DOS.

1) Run FlashBoot, click Next



2) Choose MiniOS in Main Menu



3) Choose "Minimal DOS (FreeDOS)" in MiniOS Menu and click Next



4) Choose target USB disk (flash, HDD) or image file and click Next

FlashBoot 2.0		
	Mini DOS \rightarrow USB:	Specify target
	Please specify targe	et USB device or image file
	USB device:	G: Generic Flash Disk 🔹
	🔘 Image file:	
		Size:
M-		
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

5) Enter volume label or leave it as is and click Next

FlashBoot 2.0						×
	Mini DOS \rightarrow USB	(G:): Formatting	options			
	Formatting options	5:				
	Volume label:	FREEDOS				
XNN	Filesystem:	Auto	 Map t 	o (at boot time):	A:	•
4	Set advance	d options				
	Allowed cluste	er sizes: 🚺 0.5K	✓ 1K ✓ 2K	✓ 4K 8K	✓ 16K 🗸	32K
	Disk geometry	r; Heads:	255	Sectors per tra	ack: 63	
MA T						
			< <u>B</u> ack	<u>N</u> ext >	<u>C</u> ar	ncel

6) Check summary information and click Next

FlashBoot 2.0	
Ar	Summary
	Scenario: Install mini-DOS (FreeDOS) Data source: FlashBoot builtin Target: USB storage device G: Generic Flash Disk Filesystem: Auto Volume label: FREEDOS Map to (at boot time): A: Disk CHS geometry: Heads=255 SectorsPerTrack=63 Allowed cluster sizes: 0.5k 1k 2k 4k 8k 16k 32k
	< <u>B</u> ack <u>Format Now!</u> <u>C</u> ancel

7) Wait for process completion

(88%) FlashBoot is creating bootable USB disk		×
Formatting now		
	Cancel	
Log:		
Formatting VFAT filesystem on target USB disk Copying files Unmounting VFAT filesystem in partition #1 of physical disk #5 (G:)		•
		Ŧ

8) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully		×
Completed successfully. Click OK to exit		
Log:		
Formatting VFAT filesystem on target USB disk Copying files Unmounting VFAT filesystem in partition #1 of physical disk #5 (G:) Done		*
Save log	Ok	;

4.3 Minimal SysLinux/GRUB4DOS

FlashBoot can install minimalistic SysLinux or GRUB4DOS bootloader to USB disk. Such USB disk won't be ready for use out-of-the-box, it requires further customization (copying of configuration files, kernels, initramdisks etc). But this is quite useful scenario because FlashBoot takes care of compatibility (provides Multiformat, Boot Time Mapping and BIOS compatibility features), so experienced user can focus on actual work of boot disk preparation.

1) Run FlashBoot, click Next



2) Choose MiniOS in Main Menu



3) Choose "Minimal SysLinux" or "Minimal GRUB4DOS" in MiniOS Menu and click Next



4) Choose target USB disk (flash, HDD) or image file and click Next

FlashBoot 2.0		
	Mini Syslinux $ ightarrow$ U	SB: Specify target
	Please specify targe	et USB device or image file
	OSB device:	G: Generic Flash Disk
	🔘 Image file:	
		Size:
MAI		
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

5) Enter volume label or leave it as is and click Next

FlashBoot 2.0				×
	Mini Syslinux $ ightarrow$ (JSB (G:): Formatting o	ptions	
	Formatting option	S:		
	Volume label:	SYSLINUX		
XXI	Filesystem:	Auto 💌	Map to (at boot time):	: A: •
4	Set advance	ed options		
	Allowed clust	er sizes: V 0.5K V 1	1K 📝 2K 📝 4K 📝 8K	✓ 16K ✓ 32K
	Disk geometry	y: Heads: 255	Sectors per tr	rack: 63
			< <u>B</u> ack <u>N</u> ext >	<u>C</u> ancel

6) Check summary information and click Next

FlashBoot 2.0	
	Summary
	Scenario: Install mini-SYSLinux Data source: FlashBoot builtin Target: USB storage device G: Generic Flash Disk Filesystem: Auto Volume label: SYSLINUX Map to (at boot time): A: Disk CHS geometry: Heads=255 SectorsPerTrack=63 Allowed cluster sizes: 0.5k 1k 2k 4k 8k 16k 32k
	< <u>B</u> ack <u>Format Now!</u> <u>C</u> ancel

7) Wait for process completion

(73%) FlashBoot is creating bootable USB disk	
Formatting now	
	Cancel
Log:	
Formatting VFAT filesystem on target USB disk	•
	-

8) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully	
Completed successfully. Click OK to exit	
Log:	
Formatting VFAT filesystem on target USB disk Unmounting VFAT filesystem in partition #1 of physical disk #5 (G:) Done	
Save log	ОК

5 Mini DOS on USB

92

5.1 Installed DOS -> Mini DOS on USB



2) Choose MiniOS in Main Menu



3) Choose "DOS files from DOS bootable floppy, hard disk or USB flash disk"

FlashBoot 2.0	
	Install mini OS on USB device
	FlashBoot builtins:
	Windows NT password editor
XN	Minimal DOS (FreeDOS)
	O Minimal SysLinux
	O Minimal GRUB4DOS
	System files from external source:
	OOS files from DOS bootable floppy, hard disk or USB flash disk
	ODS 7.x files from Windows 9x setup
XAT	
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

4) Choose disk or image file containing DOS files

FlashBoot 2.0	
	Mini DOS \rightarrow USB: Open DOS files
	Please specify location to DOS files which are going to be installed to USB device:
	Disk: B: disk2part1 Generic USB SD Reader
	Mage file:
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

5) Choose target USB disk (flash, HDD) or image file and click Next

FlashBoot 2.0		×
	Installed DOS (B:)) $ ightarrow$ USB: Specify target
	Please specify targe	get USB device or image file
	USB device:	G: Generic Flash Disk
	🔘 Image file:	
		Size:
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

6) Enter volume label or leave it as is and click Next

FlashBoot 2.0				×
	Installed DOS (B:)) \rightarrow USB (G:): Formatting	options	
	Formatting option	s:		
	Volume label:	DOS		
XNN	Filesystem:	Auto 👻	Map to (at boot time): A:	•
4	Set advance	ed options		
	Allowed cluste	er sizes: 🔽 0.5K 📝 1K	✓ 2K ✓ 4K ✓ 8K ✓ 16K	√ 32K
	Disk geometry	y: Heads: 255	Sectors per track: 63	
			< <u>B</u> ack <u>N</u> ext >	<u>C</u> ancel

7) Check summary information and click Next

FlashBoot 2.0	
	Summary
	Scenario: Install mini-DOS from installed DOS or Windows 9x Data source:
	disk B: Target: USB storage device G: Generic Flash Disk
	Filesystem: Auto Volume label: DOS Map to (at boot time): A: Disk CHS geometry: Heads=255 SectorsPerTrack=63 Allowed cluster sizes: 0.5k 1k 2k 4k 8k 16k 32k
	< <u>B</u> ack <u>Format Now!</u> <u>C</u> ancel

8) Wait for process completion

Formatting now	
Formatting now	
	Cancel
Log:	
Mounting VFAT filesystem on disk B: Verifying VFAT filesystem in disk B: Verifying compliance and Formatting VFAT filesystem on target USB dis	sk

9) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully	
Completed successfully. Click OK to exit	
Log:	
Mounting VFAT filesystem on disk B: Verifying VFAT filesystem in disk B: Verifying compliance and Formatting VFAT filesystem on target USB disk Copying files Unmounting VFAT filesystem in partition #1 of physical disk #1 (G:) Done	
Save log	ОК

98

5.2 Windows 9x setup files -> Mini DOS on USB



2) Choose MiniOS in Main Menu



3) Choose "DOS 7.x files from Windows 9x setup"

FlashBoot 2.0				
	Install mini OS on USB device			
	FlashBoot builtins:			
	○ Windows NT password editor			
XN	Minimal DOS (FreeDOS)			
	Minimal SysLinux			
	Minimal GRUB4DOS			
	System files from external source:			
O DOS files from DOS bootable floppy, hard disk or USB flash disk				
	OOS 7.x files from Windows 9x setup			
KAT				
	< Back <u>N</u> ext > <u>C</u> ancel			

4) Specify path to Windows 9x setup files

FlashBoot 2.0	
	Mini DOS \rightarrow USB device: Open Windows 9x setup files
	Please specify location of Windows 95/98/Me setup files.
	Windows setup files will be used as a source of DOS system files. DOS system files will be transferred to target USB device in order to make it bootable.
	D:\Archives\Windows 95 Setup
	< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

5) Choose target USB disk (flash, HDD) or image file and click Next

FlashBoot 2.0		
	Win9x-derived DO	S \rightarrow USB: Specify target
	Please specify targe	et USB device or image file
	OSB device:	G: Generic Flash Disk
	🔘 Image file:	
		Size:
MA		
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

6) Enter volume label or leave it as is and click Next

FlashBoot 2.0				×
	Win9x-derived D0	OS $ ightarrow$ USB (G:): Formatting	g options	
	Formatting option	s:		
	Volume label:	WIN95_DOS		
XNN	Filesystem:	Auto 👻	Map to (at boot time): A:	•
4	Set advance	ed options		
	Allowed cluste	er sizes: 🔽 0.5K 📝 1K	✓ 2K ✓ 4K ✓ 8K ✓ 16F	К 📝 32К
	Disk geometry	y: Heads: 255	Sectors per track: 63	
M				
			< <u>B</u> ack <u>N</u> ext >	<u>C</u> ancel

7) Check summary information and click Next

FlashBoot 2.0	
	Summary
	Scenario: Install mini-DOS from Windows 9x setup files
X	Data source: Windows 9x setup files D:\Archives\Windows 95 Setup
	Target: USB storage device G: Generic Flash Disk Filesystem: Auto Volume label: WIN95_DOS Map to (at boot time): A: Disk CHS geometry: Heads=255 SectorsPerTrack=63
	Allowed cluster sizes: 0.5k 1k 2k 4k 8k 16k 32k
	< <u>B</u> ack <u>Format Now!</u> <u>C</u> ancel

8) Wait for process completion

(5%) FlashBoot is creating bootable USB disk	
Formatting now	
	Cancel
Log:	
Verifying Windows 9x setup files Formatting VEAT filesystem on target USB disk	^
Formatting VIAT mesystem on target osb disk	
	-

9) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully	
Completed successfully. Click OK to exit	
Log:	
Verifying Windows 9x setup files Formatting VFAT filesystem on target USB disk Copying files Unmounting VFAT filesystem in partition #1 of physical disk #1 (G:) Done	
Save log	ОК

6 Setting up BIOS

1. General approach

1.1. Change the BIOS boot order so the USB device option is listed first. The BIOS is rarely setup this way by default.

If the USB port is not first in the boot order, your PC will start "normally" (i.e. boot from your hard drive) without even looking at any boot information that might be on your USB device.

Note: After setting your USB device as the first boot device, your computer will check it for boot information each time your PC starts. Leaving your computer configured this way shouldn't cause problems unless you plan on leaving the USB device attached all the time.

1.2. Attach the USB device to your computer via any available USB port.

1.3. Restart your computer.

1.4. Watch for a *Press any key to boot from external device...* message.

On some bootable devices, you may be prompted with a message to press a key before the computer will boot to the flash drive or other USB device.

If you do nothing, your computer will check for boot information on the next boot device in the list in BIOS (see Step 1.1) which will probably be your hard drive.

Note: Most of the time when trying to boot to a USB device there is no key-press prompt. The boot process usually starts immediately.

1.5. Your computer should now boot from the USB flash drive or USB based external hard drive.

Note: What happens now depends on what the bootable USB device was intended for. If you're booting to an old version of MS-DOS on a flash drive, MS-DOS will load. If you're booting to the DSL version of Linux, it will start. You get the idea.

2. Newer BIOS Boot Menu

Many newer computers detect the USB device as a hard drive (USB-HDD0). In which case, you can press a specific key (F2, F10, F11 or ESC) during system post to access the "Boot Menu". Select your USB DISK from the Boot Menu and resume startup.

3. Troubleshooting

If you tried the above steps but your computer did not boot from the USB device, check out some of the tips below.

3.1. Recheck the boot order in BIOS (Step 1.1). The number one reason a bootable flash drive or other USB device won't boot is because BIOS is not configured to check the USB port first.

3.2. Didn't find a "USB Device" boot order listing in BIOS? If your computer was manufactured around 2001 or before, it may not have this ability. If your computer is newer, check for some other ways that the USB option might be worded. In some BIOS versions, it's called "Removable Devices" or "External Devices".

3.3. Switch to another USB port. The BIOS on some motherboards only check the first few USB ports. Switch to another USB port and restart your computer.

3.4. Be sure to keep an eye for BIOS updates from your board manufacturer.

7 Decoding Status Codes

FlashBoot stage1 loader in some rare cases can encounter an error, which results in error message like this:

Boot from USB disk failed, status <00000000>. Press Ctrl+Alt+Del to reboot.

Stage1 loader can't print more verbose and self-explaining error messages, because there is no enough free space in 512 byte sector to contain such messages. That's why it prints status codes.

This section explains how to decode such status codes.

Each status digit corresponds to certain BIOS disk.

Meaning of each status digit:

- 0 No single sector of stage2 loader was read from disk, neither in CHS nor in LBA mode
- 1 Should not happen (internal error in stage1 loader)
- 2 Some sectors of stage2 loader were read from disk in LBA mode, but not all. No single sector could be read from disk in CHS mode.
- 3 All sectors of stage2 loader were read from disk in LBA mode, but checksum or signature does not mach the expected one.
- 4 Some sectors of stage2 loader were read from disk in CHS mode, but not all. No single sector could be read from disk in LBA mode.
- 5 All sectors of stage2 loader were read from disk in CHS mode, but checksum or signature does not mach the expected one.
- 6 Some sectors of stage2 loader were read from disk in LBA mode, but not all. Some sectors of stage2 loader were read from disk in CHS mode, but not all.
- 7 Some sectors of stage2 loader were read from disk in LBA mode, some other sectors of stage2 loader were read from disk in CHS mode, but checksum or signature does not mach the expected one.

8 Understanding Loader Messages

FlashBoot stage2 Loader prints messages like this during boot:

```
FlashBoot Loader Version 2.1
Copyright (C) Mikhail Kupchik 2005-2011
```

Backend (BIOS) boot disk: 128 [C=260, H=255, S=63] Frontend (emulated) boot disk: 0 [C=260, H=255, S=63]

Front end (emulated) boot disk is 0 for A:-mapped disks and 128 for C:-mapped disks. Backend (BIOS) boot disk could be any number from 0 to 255 but usually it's 0 for USB-ZIP boot mode and 128 for USB-HDD boot mode.

C, H and S values are geometry specification for USB boot disk. For more information, refer to Wikipedia.

Also BIOS could hide MBR track (we have seen this behavior on ASUS P5GDC-V in USB-ZIP boot mode for USB removable devices with valid MBR), and then FlashBoot will print such message:

BIOS hides MBR track of USB boot disk (emulating)

Ideally, all BIOSes must support CHS interface for accessing boot disks, and all modern BIOS should support LBA interface too. Unfortunately, many real world BIOSes, especially on notebooks, have bugs in implementation of CHS and LBA disk I/O. Common cases are API which always gives an error, API which reads data from somehow shifted location on the disk, API which operates normally at the beginning of the disk but shifts randomly in the middle, and API which availability depends on BIOS settings (e.g. LBA inaccessible in USB-ZIP boot mode).

FlashBoot tries its best to provide compatibility with all BIOSes and employs the following algorithm.

1) First, stage1 loader loads stage2 loader via CHS or LBA interface in slow (single-sector transfer) mode and verifies its checksum.

2) FlashBoot stage2 loader tries to read first 128 sectors of the File Allocation Table via CHS and LBA interfaces. If either fails, FlashBoot loader switches to single-sector transfer mode and retries. If that interface fails again, FlashBoot prints "LBA not available" or "CHS not available". If LBA was unavailable in stage1, it is not tried here again, then only CHS is tried.

3) If both interfaces have failed, FlashBoot prints "Press Ctrl+Alt+Del to reboot" and softly halts machine.

4) Once FAT#1 and FAT#2 is loaded into the memory via CHS and/or LBA, FlashBoot compares FAT#1 versus FAT#2 to verify each BIOS interface. If FAT#1 does not match FAT#2 for LBA-originated data, FlashBoot prints "FAT#1 != FAT#2 for LBA on this BIOS". If FAT#1 does not match FAT#2 for CHS-originated data, FlashBoot prints "FAT#1 != FAT#2 for CHS on this BIOS".

5) If both interfaces have failed, FlashBoot prints "Press Ctrl+Alt+Del to reboot" and softly halts machine.

6) FlashBoot chooses LBA if available, otherwise it falls back to CHS. Then it prints either "LBA mode" or "CHS mode". If single-sector transfer mode is active, it adds " (slow)".

Please note than CHS interface won't work correctly with USB disks larger than 8 Gb (255 * 63 * 1024 * 512 = 8,422,686,720 bytes). It will simply cut off higher part of the disk from BIOS.

9 **Reverting Multiformat to Normal**

FlashBoot formats target USB disks as Multiformat, for the sake of compatibility with different BIOSes.

Windows 2000, XP, Vista and 7 perfectly recognizes Multiformat USB disks, even when they are reformatted with non-FlashBoot software.

But under Linux, in certain configurations you may encounter some unexpected behavior.

1. The problem

Let's consider the following example.

USB flash disk is formatted with FlashBoot as Mini-FreeDOS (Chick here for more info). Then it is plugged to first machine running Linux and recognized by kernel as /dev/sdd. It is manually mounted as /dev/sdd1 and then existence of all files is verified. Then it's unmounted and new filesystem on the /dev/sdd1 is recreated with mkdosfs. This new filesystem is mounted and example.gz is copied into it.



Then USB flash disk is unmounted and taken to another Linux machine, where is automatically recognized and mounted as /dev/sdd.

root@slax:~# df			Shell - Konsole	_ 8 ×
Filesystem 1K-blocks Used Available Use% Mounted on aufs 613292 273580 339712 45% / tmpfs 511072 0 511072 0% /dev/shm /dev/sda1 3931528 2688544 1242984 69% /mnt/sda1 /dev/sdb1 11804232 9996032 1808200 85% /mnt/sdb1 /dev/sdd 1011792 128 1011664 1% /mnt/sdd root@slax:~# ls /mnt/sdd root@slax:~# Image: Construction of the state of the	root@slax:~# df			
aufs 613292 273580 339712 45% / tmpfs 511072 0 511072 0% /dev/shm /dev/sdal 3931528 2688544 1242984 69% /mnt/sdal /dev/sdbl 11804232 9996032 1808200 85% /mnt/sdbl /dev/sdd 1011792 128 1011664 1% /mnt/sdd root@slax:~# ls /mnt/sdd root@slax:~#	Filesystem	1K-blocks	Used Available Use% Mounted on	
tmpfs 511072 0 511072 0% /dev/shm /dev/sdal 3931528 2688544 1242984 69% /mnt/sdal /dev/sdbl 11804232 9996032 1808200 85% /mnt/sdbl /dev/sdd 1011792 128 1011664 1% /mnt/sdd root@slax:~# ls /mnt/sdd root@slax:~# ■ 7.	aufs	613292	273580 339712 45% /	
/dev/sdal 3931528 2688544 1242984 69% /mnt/sdal /dev/sdbl 11804232 9996032 1808200 85% /mnt/sdbl /dev/sdd 1011792 128 1011664 1% /mnt/sdd root@slax:~# ▮s /mnt/sdd root@slax:~# ▮	tmpfs	511072	0 511072 0%/dev/shm	
/dev/sdb1 11804232 9996032 1808200 85% /mnt/sdb1 /dev/sdd 1011792 128 1011664 1% /mnt/sdd root@slax:~# ■	/dev/sdal	3931528	2688544 1242984 69% /mnt/sda1	
/dev/sdd 1011792 128 1011664 1% /mnt/sdd root@slax:~# ■ ************************************	/dev/sdb1	11804232	9996032 1808200 85% /mnt/sdb1	
root@slax:~# ls /mnt/sdd root@slax:~#	/dev/sdd	1011792	128 1011664 1% /mnt/sdd	
root@slax:~#	root@slax:~# ls	/mnt/sdd		
Image: Shell - Konsole Image: Shell - Konsole	root@slax:~#			
▼ • ▼ • ▼ • ● • ● • ● • ● • ● • ● • ● •				
T T T T T T T T T T T T T T T T T T T				
🗘 🕞 🍋 🏹 🖳 2 Shell - Konsole				
Image: Shell - Konsole Image: Shell - Konsole				
T T T T T T T T T T T T T T T T T T T				
T T T T T T T T T T T T T T T T T T T				
T T T T T T T T T T T T T T T T T T T				
T T T T T T T T T T T T T T T T T T T				
T T T T T T T T T T T T T T T T T T T				
Image: Shell - Konsole Image: Shell - Konsole				
🔽 🔚 🏟 🦓 🏲 🍋 🛄 🖳 💌 2 🖷 Shell - Konsole 👘 📾 🗠 🖕 🖞 🖓				
🜃 🗃 🚳 🦓 🦵 🍋 🛄 💶 2 🖷 Shell - Konsole 🗰 🖬 🖓 🖓 🖓 🖓				
	🕅 📰 🚳 🦀	r 🛆 🔜 💷	2 🖷 Shell - Konsole	man 🗠 🔊 (5:22

But example.gz is not there!

This is because USB flash disk was used as /dev/sdd1 for the first time and as /dev/sdd for the second time. On FlashBoot-formatted MultiFormat USB disk both filesystems are syncronized and always map to the same files; but once the inner filesystem is destroyed and overwritten by mkdosfs, the outer filesystem no longer matches inner one.

2. The solution

Apply mkdosfs to entire /dev/sdd instead of /dev/sdd1.

Or use FlashBoot QuickFormat/Wipe feature if you are not familiar with Linux. FlashBoot QuickFormat/ Wipe feature will always work in demo version, even if the number of tries for target USB disk is exhaused. Thus, it allows you to restore all FlashBoot-processed MultiFormat USB disks if you decided not to buy FlashBoot after trying.

1) Run FlashBoot, click Next
| FlashBoot 2.0 | | |
|---------------|--|--|
| | Welcome to FlashBoot version 2.0 | |
| | This wizard will guide you through formatting of USB flash or USB hard disk as bootable. | |
| | For more information, refer to http://www.prime-expert.com/flashboot/ | |
| | FlashBoot is running in DEMO mode (license keyfile not found) | |
| | | |
| | | |
| | | |
| | Click Next to continue. | |
| | < Back Next > Cancel | |
| | | |

2) Choose Non-boot in Main Menu

FlashBoot 2.0	
	FlashBoot Main Menu
	 CD → USB Convert bootable CD/DVD disk to bootable USB device Floppy → USB Convert bootable floppy disk to bootable USB device USB → USB Copy bootable USB device to bootable USB device MiniOS
	Install mini OS on bootable USB device
	Non-boot Format USB device as non-bootable
	< <u>B</u> ack <u>C</u> ancel

3) Choose either "Quick format" or "Wipe"

FlashBoot 2.0		×
	Format USB device as non-bootable	
	Non-bootable formatting options:	
	Quick format	
	© Wipe	
4		
	< <u>B</u> ack <u>N</u> ext >	<u>C</u> ancel

Wipe will erase all data from USB disk, but will take longer to proceed.

4) Choose target USB disk (flash, HDD) or image file and click Next

FlashBoot 2.0		×
	Quick Format: Spe	cify target
	Please specify targ	et USB device or image file
	OSB device:	G: Generic Flash Disk
	🔘 Image file:	
		Size:
		< <u>B</u> ack <u>N</u> ext > <u>C</u> ancel

FlashBoot 2.0					×
	Quick Format (G:)	: Formatting options			
	Formatting option	5:			
	Volume label:	CLEAR			
XNI	Filesystem:	Auto	• Layo	ut: partitioned	•
	Set advance	ed options			
	Allowed cluste	er sizes: 📝 0.5K 📝 1	К 📝 2К 📝 4	К 📝 8К 📝 16К	✓ 32K
The					
			< <u>B</u> ack	<u>N</u> ext >	<u>C</u> ancel

5) Enter volume label or leave it as is and click Next

6) Check summary information and click Next

FlashBoot 2.0	
	Summary
	Scenario: Quick format Data source: FlashBoot builtin Target: USB storage device G: Generic Flash Disk Filesystem: Auto Volume label: CLEAR Layout: partitioned Allowed cluster sizes: 0.5k 1k 2k 4k 8k 16k 32k
	< <u>B</u> ack <u>Format Now!</u> <u>Cancel</u>

112 FlashBoot User Manual

7) Wait for process completion

(50%) FlashBoot is creating bootable USB disk	
Formatting now	
romating now	
	Cancel
Log:	
Opening output USB disk Deducing VFAT filesystem parameters for physical disk #5 (G:) Creating empty VFAT filesystem on physical disk #5 (G:)	
	-

8) USB disk is ready for use. Click OK to exit

FlashBoot completed successfully	•	×
Completed successfully. Click OK to exit		
Log:		
Opening output USB disk Deducing VFAT filesystem parameters for physical disk #5 (G:) Creating empty VFAT filesystem on physical disk #5 (G:) Done		~
Save log	OK	

10 Using command-line version

FlashBoot can be invoked from command line. This is particularly useful when you need some automation or unattended execution of FlashBoot.

Installation folder contains fb-cli.exe file (CLI stands for "Command Line Interface") alongside fb-gui.exe (GUI = "Graphical User Interface"). Installer creates shortcut only for fb-gui.exe.

Command-line version of FlashBoot has the following options:

```
FlashBoot -scenario ConvertWinXpInstallCDorDVD
-input -disk <CD/DVD drive letter>:
| -disk <CD/DVD mount path>
| -image ImageFile.iso
| -path <Local path to CD/DVD files>
-fstype <Filesystem type>
-label <Volume label>
-geometry <Geometry options>
-output -disk <USB drive letter>:
| -disk <USB mount path>
| -physicaldisk <Number>
```

-image <Image file name> <Image file size> -reserve <Size of unallocated space at the end of disk, 0 if not required> Convert Microsoft Windows XP installation CD to bootable USB disk FlashBoot -scenario ConvertWinVistaOr7InstallCDorDVD -input -disk <CD/DVD drive letter>: | -disk <CD/DVD mount path> | -image ImageFile.iso | -path <Local path to CD/DVD files> <Filesystem type> -fstype <Volume label> -label -geometry <Geometry options> -output -disk <USB drive letter>: | -disk <USB mount path> | -physicaldisk <Number> | -image <Image file name> <Image file size> <Size of unallocated space at the end of disk, 0 if not -reserve required> Convert Microsoft Windows Vista or Windows 7 installation CD/DVD to bootable USB disk FlashBoot -scenario ExtractRecConFromWinInstallCDorDVD -disk <CD/DVD drive letter>: -input -disk <CD/DVD mount path> | -image ImageFile.iso
| -path <Local path to CD/DVD files> <Filesystem type> -fstype -label <Volume label> -geometry <Geometry options> -disk <USB drive letter>: -output | -disk <USB mount path> | -physicaldisk <Number> | -image <Image file name> <Image file size> <Size of unallocated space at the end of disk, 0 if not -reserve required> Extract Microsoft Windows Recovery Console from Microsoft Windows installation CD and transform it to bootable USB disk FlashBoot -scenario ConvertBartPEorWindowsPE -disk <CD/DVD drive letter>: -input | -disk <CD/DVD mount path> | -image ImageFile.iso
| -path <Local path to CD/DVD files> <Filesystem type> -fstype <Volume label> -label -geometry <Geometry options> -disk <USB drive letter>: -output | -disk <USB mount path> | -image <Image file name> <Image file size> | -physicaldisk <Number> <Size of unallocated space at the end of disk, 0 if not -reserve required> Convert BartPE or Microsoft Windows PE (Preinstallation Environment) to bootable USB disk FlashBoot -scenario ConvertIsolinuxBootCDorDVD -disk <CD/DVD drive letter>: -input | -disk <CD/DVD mount path> | -image ImageFile.iso <Local path to CD/DVD files> | -path

-mapto c | a -fstype <Filesystem type> -label <Volume label> -geometry <Geometry options> -disk <USB drive letter>: -output | -disk <USB mount path> | -physicaldisk <Number> -image <Image file name> <Image file size> <Size of unallocated space at the end of disk, 0 if not -reserve required> Convert ISOLinux-based bootable CD/DVD disk to bootable USB disk FlashBoot -scenario ConvertHddFlopEmulBootCDorDVD -input -disk <CD/DVD drive letter>: | -disk <CD/DVD mount path> | -image ImageFile.iso -mapto c | a <Filesystem type> -fstype -label <Volume label> -geometry <Geometry options> -output -disk <USB drive letter>: | -disk <USB mount path> | -physicaldisk <Number> -image <Image file name> <Image file size> <Size of unallocated space at the end of disk, 0 if not -reserve required> Convert HDD/floppy-emulation-based bootable CD/DVD disk to bootable USB disk FlashBoot -scenario ConvertGrub4DosBootCDorDVD -input -disk <CD/DVD drive letter>: | -disk <CD/DVD mount path> | -image ImageFile.iso | -path <Local path to CD/DVD files> c | a -mapto <Filesystem type> -fstype -label <Volume label> -geometry <Geometry options> -disk <USB drive letter>: -output | -disk <USB mount path> | -physicaldisk <Number> | -image <Image file name> <Image file size> <Size of unallocated space at the end of disk, 0 if not -reserve required> Convert Grub4Dos-based bootable CD/DVD disk to bootable USB disk FlashBoot -scenario WrapAnyCDorDVD -disk <CD/DVD drive letter>: -input | -disk <CD/DVD mount path> | -image ImageFile.iso -mapto c | a -fstype <Filesystem type> -label <Volume label> -geometry <Geometry options> -disk <USB drive letter>: -output | -disk <USB mount path> | -physicaldisk <Number> | -image <Image file name> <Image file size> <Size of unallocated space at the end of disk, 0 if not -reserve required> Wrap image file of any CD/DVD disk to Grub4Dos-based bootable USB disk FlashBoot -scenario CopyDosDisk

-input -disk <Floppy/HDD/USB drive letter>: | -disk <Floppy/HDD/USB mount path>
| -image ImageFile.img c | a -mapto <Filesystem options> -fstype -label <Volume label> -geometry <Geometry options> -disk <USB drive letter>: -output | -disk <USB mount path> | -physicaldisk <Number> | -image <Image file name> <Image file size> -reserve <Size of unallocated space at the end of disk, 0 if not required> Make DOS-based bootable USB disk from some floppy, HDD or USB disk. FlashBoot -scenario CopySyslinuxDisk -disk <Floppy/HDD/USB drive letter>: -input | -disk <Floppy/HDD/USB mount path> | -image ImageFile.img -mapto c | a -fstype <Filesystem type> -label <Volume label> -geometry <Geometry options> -disk <USB drive letter>: -output | -disk <USB mount path> | -physicaldisk <Number> -image <Image file name> <Image file size> -reserve <Size of unallocated space at the end of disk, 0 if not required> Make Syslinux-based bootable USB disk from some floppy, HDD or USB disk. FlashBoot -scenario CopyGrub4DosDisk -input -disk <Floppy/HDD/USB drive letter>: | -disk <Floppy/HDD/USB mount path> | -image ImageFile.img -mapto c | a -fstype <Filesystem type> -label <Volume label> -geometry <Geometry options> -output -disk <USB drive letter>: | -disk <USB mount path> | -physicaldisk <Number> | -image <Image file name> <Image file size> -reserve <Size of unallocated space at the end of disk, 0 if not required> Make Grub4Dos-based bootable USB disk from some floppy, HDD or USB disk. FlashBoot -scenario CopyWindowsNt4or5Disk -disk <Floppy/HDD/USB drive letter>: -input | -disk <Floppy/HDD/USB mount path> | -image ImageFile.img -mapto c | a -fstype <Filesystem type> <Volume label> -label -geometry <Geometry options> -disk <USB drive letter>: -output | -disk <USB mount path> | -physicaldisk <Number> | -image <Image file name> <Image file size> <Size of unallocated space at the end of disk, 0 if not -reserve required> Make Windows NT 4 or Windows NT 5 based bootable USB disk from some floppy, HDD

```
or USB disk.
FlashBoot -scenario CopyWindowsNt6Disk
                    -disk <Floppy/HDD/USB drive letter>:
          -input
                     | -disk <Floppy/HDD/USB mount path>
                    | -image ImageFile.img
          -mapto
                    c | a
          -fstype
                    <Filesystem type>
                    <Volume label>
          -label
          -geometry <Geometry options>
                    -disk <USB drive letter>:
          -output
                     | -disk <USB mount path>
                     | -physicaldisk <Number>
                     -image <Image file name> <Image file size>
          -reserve
                    <Size of unallocated space at the end of disk, 0 if not
required>
   Make Windows NT 6 based bootable USB disk from some floppy, HDD or USB disk.
FlashBoot -scenario WrapAnyFloppy
                    -disk <Floppy drive letter>:
          -input
                     | -disk <Floppy mount path>
                    | -image FloppyImageFile.img
         -mapto
                    c | a
                    <Filesystem type>
          -fstype
          -label
                    <Volume label>
          -geometry <Geometry options>
                    -disk <USB drive letter>:
          -output
                     | -disk <USB mount path>
                     | -physicaldisk <Number>
                     | -image <Image file name> <Image file size>
                    <Size of unallocated space at the end of disk, 0 if not
          -reserve
required>
    Wrap image file of any floppy disk to Syslinux-based bootable USB disk
FlashBoot -scenario NtPasswordWizard
          -mapto
                   c | a
                    <Filesystem type>
          -fstype
         -label
                    <Volume label>
         -geometry <Geometry options>
          -output
                    -disk <USB drive letter>:
                     | -disk <USB mount path>
                     | -physicaldisk <Number>
                     | -image <Image file name> <Image file size>
                    <Size of unallocated space at the end of disk, 0 if not
          -reserve
required>
   Make Syslinux-based bootable UFD with NT password wizard from EBCD
FlashBoot -scenario MiniDosFromBuiltinFreeDos
                    c | a
          -mapto
         -fstype
                    <Filesystem type>
         -label
                    <Volume label>
          -geometry <Geometry options>
          -output
                    -disk <USB drive letter>:
                     | -disk <USB mount path>
                     | -physicaldisk <Number>
                    | -image <Image file name> <Image file size>
                    <Size of unallocated space at the end of disk, 0 if not
          -reserve
required>
   Make minimal DOS-based bootable USB disk with Volkov Commander from builtin
FreeDOS
    Perform necessary A:/C: substitution in \CONFIG.SYS and *.BAT files.
```

```
FlashBoot -scenario MiniDosFromWin9xSetupFiles
                    -path <Path to folder with Windows 95/98/Me setup files>
          -input
          -mapto
                    c | a
          -fstype
                    <Filesystem type>
          -label
                     <Volume label>
          -geometry <Geometry options>
                     -disk <USB drive letter>:
          -output
                     | -disk <USB mount path>
                     | -physicaldisk <Number>
                     -image <Image file name> <Image file size>
                     <Size of unallocated space at the end of disk, 0 if not
          -reserve
required>
   Make minimal DOS-based bootable USB disk with Volkov Commander from Windows
95/98/Me setup files.
   Perform necessary A:/C: substitution in \CONFIG.SYS and *.BAT files.
FlashBoot -scenario MiniDosFromInstalledDosOrWin9x
                    -disk <Floppy/HDD/USB drive letter>:
          -input
                     | -disk <Floppy/HDD/USB mount path>
| -image ImageFile.img
          -mapto
                    c | a
                    <Filesystem type>
          -fstype
          -label
                     <Volume label>
          -geometry <Geometry options>
                     -disk <USB drive letter>:
          -output
                     | -disk <USB mount path>
                     | -physicaldisk <Number>
                     -image <Image file name> <Image file size>
          -reserve
                     <Size of unallocated space at the end of disk, 0 if not
required>
   Make minimal DOS-based bootable USB disk with Volkov Commander from installed
copy of DOS
    or MS Windows 95/98/Me. Perform necessary A:/C: substitution in \CONFIG.SYS and
*.BAT files.
FlashBoot -scenario MiniSyslinux
                   c | a
          -mapto
          -fstvpe
                    <Filesystem type>
                    <Volume label>
          -label
          -geometry <Geometry options>
                     -disk <USB drive letter>:
          -output
                     | -disk <USB mount path>
                     | -physicaldisk <Number>
                     | -image <Image file name> <Image file size>
                    <Size of unallocated space at the end of disk, 0 if not
          -reserve
required>
   Make minimal Syslinux-based bootable USB disk
FlashBoot -scenario MiniGrub4Dos
          -mapto c | a
          -fstype
                    <Filesystem type>
          -label
                    <Volume label>
          -geometry <Geometry options>
                     -disk <USB drive letter>:
          -output
                     | -disk <USB mount path>
                     | -physicaldisk <Number>
                     | -image <Image file name> <Image file size>
                     <Size of unallocated space at the end of disk, 0 if not
          -reserve
required>
   Make minimal Grub4Dos-based bootable USB disk
FlashBoot -scenario RawCopy
```

-input -disk <USB drive letter>: | -disk <USB mount path> | -physicaldisk <Number> | -image ImageFile.img -disk <USB drive letter>: -output | -disk <USB mount path> | -physicaldisk <Number> | -image <Image file name> Do raw sector-to-sector copy from one USB flash disk or image file to another one FlashBoot -scenario Wipe -layout partitioned | superfloppy -fstype <Filesystem type> -label <Volume label> -disk <USB drive letter>: -output | -disk <USB mount path> | -physicaldisk <Number> Wipe all files and free space on USB disk, then create empty nonbootable filesystem FlashBoot -scenario QuickFormat -layout partitioned | superfloppy <Filesystem type> -fstype -label <Volume label> -disk <USB drive letter>: -output | -disk <USB mount path> | -physicaldisk <Number> | -image <Image file name> <Image file size> -reserve <Size of unallocated space at the end of disk, 0 if not required> Create empty nonbootable filesystem on USB disk or in image file FlashBoot -detect CD -disk <CD/DVD drive letter>: -input | -disk <CD/DVD mount path> | -image ImageFile.iso Detect type of CD/DVD to help end user choose CD/DVD conversion scenario FlashBoot -detect VFatDisk -disk <Floppy/HDD/USB drive letter>: -input | -disk <Floppy/HDD/USB mount path> | -image ImageFile.img Detect type of VFAT disk to help end user choose VFAT conversion scenario <Filesystem type> explained in examples: Automatically choose between FAT12, FAT16 and auto FAT32 with default cluster sizes fat16 Allow FAT16 only fat16(1024) Allow FAT16 only with cluster size of 1K, disallow other cluster sizes, FAT12 and FAT32 fat16(1024,2048) Allow FAT16 only with cluster size of 1K or 2K, disallow other cluster sizes, FAT12 and FAT32 fat16(1024,2048),fat32(4096) Allow FAT16 with cluster size 1K or 2K, or FAT32 with cluster size 4K. Disallow all other cluster sizes and FAT12. fat12(512), fat32(4096, 8192) Allow FAT12 with cluster size 512, or FAT32 with cluster size 4K or 8K. Disallow all other cluster sizes and FAT16.

```
<Geometry options> explained in examples:
                                 Use default geometry (255 heads, 63 sectors per
   auto
track)
  h:255,s:63
                                 Same as above
  h:127,s:32
                                 127 heads, 32 sectors per track
   s:32
                                 Override sectors per track (255 heads, 32 sectors
per track)
  h:127
                                 Override heads (127 heads, 63 sectors per track)
Minimum value for sectors per track is 15 for fat32 formatted disks and 11 for
fat12/fat16 formatted disks.
Maximum value for sectors per track is 63.
Head count value must be between 2 and 255.
Cylinder count is detected automatically by FlashBoot.
```

11 FlashBoot as a Portable Software

Portable software is a type of software which does not require installation and can be run from USB Disk or other type of removable media on any computer this disk is plugged to. For more information about portable software, refer to PortableApps.com or other similar websites.

Portable FlashBoot can be downloaded from http://www.prime.expert.com/flashboot/ along with installable FlashBoot.

There is no difference in binary files between Portable FlashBoot and installable FlashBoot. FlashBoot detects its "portability" by presence of FlashBoot.ini. If this file does not exist, then FlashBoot stores its settings in Windows Registry (HKEY_CURRENT_USER\Software\FlashBoot\2.0 registry key), otherwise it stores its settings in FlashBoot.ini.

12 License Agreement

FlashBoot version 2.x End User License Agreement

This License Agreement is a legal agreement between you ("LICENSEE") (either an individual or a single entity) and Mikhail N. Kupchik ("PRIMEEXPERT") for the software product FlashBoot 2.x which includes computer software and electronic documentation (collectively the "SOFTWARE"). Read it carefully before using the SOFTWARE. If you did not obtain this copy of the SOFTWARE legally, please destroy the copy immediately.

By installing, copying, or otherwise using the SOFTWARE, you agree to be bound by the terms of this License Agreement.

If you do not agree to the terms of this License Agreement, PRIMEEXPERT is unwilling to license the SOFTWARE. In such event, you may not install, copy or otherwise use the SOFTWARE. YOU AGREE THAT YOUR USE OF THE SOFTWARE ACKNOWLEDGES THAT YOU HAVE READ THIS LICENSE, UNDERSTAND IT, AND AGREE TO BE BOUND BY ITS TERMS AND CONDITIONS.

Versions of the SOFTWARE

The SOFTWARE is available in two versions: (a) the Demo Version and (b) the Full Featured Version. Each of the above versions of the SOFTWARE is subject to the terms and conditions of this Agreement and may be used only in accordance with the terms and conditions of this Agreement.

1. Grant of License (general terms)

PRIMEEXPERT grants you the following rights provided you comply with all terms and conditions of this EULA:

(a) Use. You may use the Software Product on one or more computers. You may not separate component parts of the Software Product for use on more than one computer. You do not have the right to distribute the Software Product. You may load the Software Product into Your Computer's temporary memory (RAM) for purposes of using the Software Product.

(b) Storage. You may copy the Software Product into the local memory or storage device.

(c) Copying. You may make archival or back-up copies of the Software Product, provided the copy contains all of the original Software Product's proprietary notices and that it is used only for back-up purposes.

(d) Reservation of Rights. PRIMEEXPERT and its suppliers reserve all rights not expressly granted to you in this EULA.

2. Grant of License

(a) Demo Version

If you have licensed the Demo Version of the SOFTWARE, subject to the terms and conditions of this Agreement, PRIMEEXPERT grants you, LICENSEE, a limited, non-exclusive, non-sublicenseable license to install one (1) copy of the Demo Version of the SOFTWARE and use it on one or more physical workstation(s) or server(s) by one (1) person. You muse use Demo Version of the SOFTWARE only in accordance with this Agreement solely for internal evaluation use of the SOFTWARE. The Demo Version of the SOFTWARE may not be used for any commercial development or for any deployment or ultimate production purposes.

(b) Full Featured Version

If you have purchased a license for the Full Featured Version of the SOFTWARE, subject to the terms and conditions of this Agreement, PRIMEEXPERT grants you, LICENSEE, a limited, non-exclusive, non-transferable, non-sublicenseable license to install either:

a) one (1) the Full Featured Version of the SOFTWARE only on a single physical workstation or server for use by one or several persons, but not simultaneously; OR

b) one or more copies of the Full Featured Version of the SOFTWARE on one or more physical workstation(s) or server(s) for use by one (1) person; but not (a) and (b) at the same time.

LICENSEE may use that copy (copies) of the SOFTWARE only in binary object code form and only in accordance with this Agreement.

LICENSEE may make one or more copies of the SOFTWARE solely for backup or archival purposes, provided that LICENSEE reproduces and includes all copyright and other proprietary notice(s) on the copy.

3. Upgrades

To use a Software Product identified as an upgrade, you must first be licensed for the original Software Product identified by PRIMEEXPERT as eligible for the upgrade.

4. Transfer

(a) Third Party. The initial user of the Software Product may make a one-time transfer of the Software Product to another end user. Any transfer must include all component parts, media, printed materials, this EULA. The transfer may not be an indirect transfer, such as a consignment. Prior to the transfer, the end user receiving the transferred product must agree to all the EULA terms. Upon transfer of the Software Product, your license is automatically terminated.

(b) Restrictions. You may not rent, lease or lend the Software Product or use the Software Product for commercial timesharing or bureau use. You may not sublicense, assign or transfer the license or Software Product except as expressly provided in this EULA.

5. Proprietary Rights

All intellectual property rights in the Software Product and user documentation are owned by PRIMEEXPERT or its suppliers and are protected by law, including but not limited to international copyright, trade secret, and trademark law, as well as other applicable laws and international treaty provisions. You shall not remove any product identification, copyright notices or proprietary restrictions from the Software Product.

6. Restrictions

LICENSEE may not:

(i) Decompile, reverse engineer, disassemble or otherwise reduce the SOFTWARE to a human perceivable form.

(ii) Rent, lease, lend, transfer or otherwise transfer rights to the SOFTWARE.

(iii) Translate, adapt, modify the SOFTWARE or create derivative works based upon the SOFTWARE or any part thereof.

(iv) Distribute copies of the SOFTWARE.

(v) Electronically transfer the SOFTWARE over a network, a telephone circuit or the Internet.

(vi) Remove any proprietary notices or labels on the SOFTWARE.

(vii) Use the SOFTWARE to encode, reproduce or copy any material or intellectual property you do not have the right to encode, reproduce or copy.

7. Termination

This License Agreement is effective until terminated. PRIMEEXPERT may terminate this License Agreement if LICENSEE fails to comply with the terms and conditions of this License Agreement. In

such event, LICENSEE must destroy all copies of the SOFTWARE and all of its component parts. All provisions relating the ownership, disclaimer of warranties, limitation of liability and the general provisions shall survive the termination of this License Agreement.

8. Ownership

The SOFTWARE is licensed, NOT sold. The foregoing License Agreement gives LICENSEE limited rights to use the SOFTWARE. All rights, title and interest, including all copyrights, in and to the SOFTWARE and any copies LICENSEE is permitted to make herein are exclusively owned by PRIMEEXPERT and/or its suppliers and are protected by copyright, trademark and trade secret law and international treaties. PRIMEEXPERT retains all rights not expressly granted to LICENSEE in this License Agreement.

9. Disclaimer of Warranties

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, PRIMEEXPERT PROVIDE THE SOFTWARE PRODUCT AS IS AND WITH ALL FAULTS, AND HEREBY DISCLAIM ALL OTHER WARRANTIES AND CONDITIONS, EITHER EXPRESS, IMPLIED, OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF TITLE AND NON-INFRINGEMENT, ANY IMPLIED WARRANTIES, DUTIES OR CONDITIONS OF MERCHANTABILITY, OF FITNESS FOR A PARTICULAR PURPOSE, AND OF LACK OF VIRUSES ALL WITH REGARD TO THE SOFTWARE PRODUCT. Some states/jurisdictions do not allow exclusion of implied warranties or limitations on the duration of implied warranties, so the above disclaimer may not apply to you in its entirety.

10. Limitation of Liability

Notwithstanding any damages that you might incur, the entire liability of PRIMEEXPERT and any of its suppliers under any provision of this EULA and your exclusive remedy for all of the foregoing shall be limited to U.S. \$5.00.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL PRIMEEXPERT AND BE LIABLE FOR ANY CONSEQUENTIAL, INCIDENTAL, DIRECT, INDIRECT, SPECIAL, PUNITIVE OR OTHER DAMAGES WATHSOEVER (INCLUDING WITHOUT LIMITATION DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION OR OTHER PECUNIARY LOSS) ARISING OUT OF THIS AGREEMENT OR CAUSED BY THE USE OR INABILITY TO USE THE SOFTWARE, EVEN IF PRIMEEXPERT HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Some states/jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you.

11. No other warranties

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, MANUFACTURER AND ITS SUPPLIERS DISCLAIM WITH RESPECT TO THE SOFTWARE ALL OTHER WARRANTIES AND CONDITIONS, EITHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING, BUT NOT LIMITED TO ANY (IF ANY) WARRANTIES OR CONDITIONS OF OR RELATED TO: TITLE, NON-INFRINGEMENT, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, LACK OF VIRUSES, ACCURACY OR COMPLETENESS OF RESPONSES, RESULTS, LACK OF NEGLIGENCE OR LACK OF WORKMANLIKE EFFORT, QUIET ENJOYMENT, QUIET POSSESSION, AND CORRESPONDENCE TO DESCRIPTION. THE ENTIRE RISK ARISING OUT OF USE OR PERFORMANCE OF THE SOFTWARE AND ANY SUPPORT SERVICES REMAINS WITH YOU. THIS LIMITED WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY HAVE OTHERS WHICH VARY FROM STATE/JURISDICTION TO STATE/JURISDICTION.

12. Capacity and authority to contract

You represent that you are of the legal age of majority in your state of residence and, if applicable, you are duly authorized by your employer to enter into this contract.

13. Entire Agreement

This License Agreement constitutes the entire agreement between LICENSEE and PRIMEEXPERT and supersedes any other prior agreements or understandings, whether oral or written, regarding the SOFTWARE. If a provision of this agreement is deemed null and void, invalid or without effect, the remainder of this agreement shall remain in effect.