

- **AptioMemoryFix.efi** ist ein komplett neu entwickelter Treiber aus dieser Serie, Autor vit9696, mit zusätzlicher automatischer Suche nach dem besten Wert für slide=xxx. Im Moment wird empfohlen, es an erster Stelle zu verwenden

- **OsxLowMemFix.efi** - vereinfachte Version von AptioFix, geeignet für einige fremde UEFI-BIOS-Varianten (Insyde H2O).

- Diese fünf Aptio-Optionen werden nicht gleichzeitig verwendet, auch wenn alle vorhanden sind, wählt Clover einen von ihnen nach Priorität aus: **AptioMemoryFix**, **OsxAptioFix3Drv.efi**, **OsxAptioFix2Drv.efi**, **OsxAptioFixDrv.efi** und zu letzt **OsxLowMemFix.efi**.

- Für die neue Revision des Clover 5120 wird empfohlen, alle diese Treiber zu vergessen *Fix*, und die Menge der OcQuirks und OpenRuntime zu verwenden, die jetzt im Paket enthalten sind Clover, und ab 5122 wird OcQuirks nicht mehr benötigt, es ist in Clover enthalten;

- **NvmExpressDxe** - NvmExpress-Controller-Treiber, der als SATA-Ersatz für SSD-Laufwerke dient. Sie sollten bedenken, dass, wenn der Controller auf Motherboard ist, wird dieser Treiber höchstwahrscheinlich bereits im UEFI-BIOS vorhanden sein. Sie benötigen es für zusätzlicher Controller.

- **Usb*.efi**, **UHCl.efi**, **EHCl.efi**, **XHCl.efi**, **OHCl.efi** - USB-Treibersatz, für die Option B Fälle, in denen die eingebauten Treiber aus irgendeinem Grund nicht gut funktionieren. Warum sollte Ich dennoch auf einmal benutzen? Vielleicht gibt es eine Art Verbindung zu anderen Funktionen, die zu deaktivieren sind.

- **PS2Mouse...**, **PS2MouseAbsolute...**, **UsbMouse...** - Satz von Treibern zur Unterstützung der Maus/Trackpad/Touchpad-Zeiger in CloverGUI. Bei der Bedienung des Systems, haben diese Treiber keine Wirkung. Das UEFI-BIOS sollt ebenfalls einen solchen Treiber haben, Sie sollten ihm nicht in die Quere kommen.

- **AptioInputFix** - auch für Maus und Tastatur, wenn die UEFI-BIOS-Option richtig funktioniert .
Autor vit9696.

- **DataHubDxe.efi** - dieser Treiber ist bereits in Option A vorhanden, und zwar möglicherweise, ist auch sie in UEFI verfügbar. Aber falls es nicht da ist, lohnt es sich, sie extern zu booten. Es wird keinen Konflikt entstehen, aber es wird sicher sein, dass es einen gibt. Mehr und mehr sehe ich Beschwerden von Benutzern, dass etwas nicht geladen wird oder für sie nicht funktioniert. Und sie haben keine DataHubDxe! Woher haben Sie all diese Gerissenheit? Hat man Sie auf tonimac davon überzeugt, dass dieser Treiber extra ist? Dann kommen Sie auch nicht damit zu mir.

- **CsmVideoDxe.efi** - Videotreiber, der mehr Reichweite bietet als die eingebaute UEFI-Bildschirmgröße wird für Option B benötigt, wenn die Grafikkarte kein UEFI VideoBIOS hat. Es gibt Varianten, bei denen dieser Treiber nicht kompatibel damit ist mit. Wenn Sie eine Videokarte, z. B. eine eingebettete Intel-Grafikkarte, verwenden, seien Sie vorsichtig. Der Treiber benötigt die Aktivierung des Kontrollkästchens CSM Enable im BIOS.

- **GrubNTFS**, **GrubEXFAT**, **GrubHFSPLUS**, **GrubUDF...** - Satz von Treibern für die meisten verschiedene Dateisysteme, die von GRUB-Quellen angepasst wurden, um in als Teil von Clover zu arbeiten. Es ist sehr ermutigend, dass wir jetzt Unterstützung für all diese dateibasierten Systeme, und lizenztechnisch sauber, haben. Dank an AnV, der die folgenden Informationen beschafft und angepasst hat Quellen. Sie haben einige Vorteile gegenüber unseren Drivern, zum Beispiel GrubHFS unterstützt komprimierte Volumes, aber der Status des Ganzen ist eher "Beta". Sie sind langsam und fehlerhaft. Und der Funktionsumfang ist nicht beeindruckend. HFS - keine Unterstützung, UDF - nicht alle Hider sind lesbar, EXFAT - kein Schreiben (ich wünschte, es wäre so!).

- **AppleImageCodec.efi** - Decoder für PNG- und BMP-Dateien, benötigt für FileVault2.

- **AppleKeyAggregator.efi** - erstellt ein spezielles Protokoll für die Eingabe eines Passworts in der FileVault2-Schnittstelle.

- **AppleUITheme.efi** - erstellt das Protokoll, mit dem FileVault2 den Hintergrund zeichnet auf dem Bildschirmbild (in Sierra sind es z. B. unscharfe Berge).

- **AppleKeyFeeder.efi** - erforderlich für die Eingabe eines Kennworts in der FileVault2-Schnittstelle, wenn eine PS2-Tastatur verfügbar ist. Es gibt eine Alternative dazu **AptioInputFix.efi** von vit9696.

- **FirmwareVolume.efi** - erzeugt das FirmwareVolume-Protokoll, das jedoch nur in FileVault2 verwendet wird, das erwartet, dort ein Bild zu finden den Cursor. Ja. Es ist da.

- **SMCHelper.efi** - erzeugt das AppleSMCProtocol, das folgende Eigenschaften haben sollte Der SMC-Chip kommuniziert mit dem SMC-Chip und erhält von dort Peripherie-Statusinformationen. Wir haben einen Emulator erstellt, der die Werte zurückgibt, die von ihm erwartet werden. Gefüllt mit echten Inhalten ist sowohl Clover selbst als auch FakeSMC, das speichert Tasten im NVRAM, wenn FakeSMC mit dieser Funktion installiert ist. Alle diese wird vor allem für FileVault2 benötigt, hat aber auch Auswirkungen auf das System und ohne sie ist keine Tastenrückgabe an das System vorgesehen, da es sich um Zweifel an der Durchführbarkeit dieser Maßnahme. Es gibt eine externe Alternative zu VirtualSMC von Mit Ihren Gesetzen.

- **HashServiceFix.efi** - Erzeugt HashService-Gruppenprotokolle, wenn UEFI sie nicht hat. FileVault2 interagiert mit ihnen, wie Sie in den Protokollen sehen können. Aber den visuellen Effekt habe ich nicht sägen.

- **APFS.efi** ist ein neuer Dateisystemtreiber, der in 10.13 eingeführt wurde. hat kein Open-Source-Analogon, daher verwenden wir das, was in der installierten Apple-System unter /usr/standalone/i386/apfs.efi, oder → - ApfsDriverLoader ist eine Open-Source-Alternative zum APFS-Treiber. Dank an savvas und andere zu Mitgliedern der Acidanthera-Gruppe. Seine Essenz ist, den richtigen APFS-Treiber zu laden einfach aus dem Container und hat somit immer die richtige Version. Wenn jedoch wir booten das alte System, wir bekommen den alten Treiber und können nicht lesen APFS-Partition mit dem neuen System. Hier geht es mehr um apfs.kext, aber das Problem ist ähnlich.

- **AudioDxe** ist ein von Goldfish64 entwickelter Soundtreiber. Es kann verwendet werden für um den Ton beim Hochfahren des Computers abzuspielen. Es ist jedoch eine Verschönerung, wenn ein solcher Ton nicht benötigt wird, können Sie ohne den Treiber leben. Es wird ein Fehler bemerkt, dass danach in Wenn der Ton auf dem System selbst fehlt, dann deinstallieren Sie einfach diesen Treiber.

Herunterladen der Kexte

In der Ordnerstruktur sehen wir einen gemeinsamen Ordner /EFI/CLOVER/kexts/ mit den Ordnern 10.7/, 10.11/ und so weiter, sowie Ordner mit den Namen OFF/ und OTHER/. Schauen wir uns das Thema einmal genauer an. "Kernel Extensions" für das Betriebssystem MacOSX, die meisten davon sind Treiber für irgendeine Art von Gerät. Oder ein Pseudo-Treiber eines nicht existierenden Geräte wie FakeSMC. Wir laden sie herunter, weil ein sauberes System von Apple einfach nicht unsere Geräte unterstützt, oder wir wollen neue Funktionen einführen. Wir bezieht uns hier speziell auf MacOSX, für Linux und Windows werden keine Kexte geladen.

Ab der Revision 3281 werden zuerst alle Kexte aus dem Ordner Others/ geladen und dann aus Ordner mit einer bestimmten Betriebssystemnummer. Dies ist sinnvoll, weil die meisten nicht an die Systemversion gebunden sind und sie funktionieren mit jeder Version gleichermaßen, FakeSMC im Besonderen. Wenn der Treiber geladen ist, werden möglicherweise bestimmte Ordner benötigt nur für ein bestimmtes System zu unterstützen. AppleHDA, zum Beispiel, ist auf jedem System anders (Ich weiß es nicht, tatsächlich, laden sie es mit Clover). Für mich macht also nur FakeSMC Sinn, den Rest kopieren Sie in den System-Ordner, und dann auch nur zum Zeitpunkt der Installation, denn im Installationsprogramm werden alle Caches gepackt sein, und es ist unmöglich, etwas Eigenes hineinzuschieben (obwohl... es scheint, dass OpenCore kann es). Aber da die Leute alle Kext-Sets mit Clover laden wollen, soll es eine solche Möglichkeit geben.

Optionen:

Optionen:

Erstens, wenn Sie das Menü manuell schreiben (Benutzerdefiniert → Einträge), können Sie für die einzelnen Elemente InjectKexts=Yes/No/Detect beschreiben.

Zweitens: Es gibt eine gemeinsame Einrichtung für alle:

```
<key>SystemParameter</key>
<dict>
    <key>InjectKexts</key>
    <string>yes</string>.
```

Detect ist veraltet und wird nicht mehr verwendet, und wir wollen immer das Kexte injizieren werden.

Ein weiterer Punkt. Im Captain gibt es eine zusätzliche Prüfung für unsigned Kexte. Um sie zu überwinden, muss der SIP (System Integrity Protection) ausgeschaltet werden. Das erfolgt in der Config im Schlüssel:

```
CsrActiveConfig
<key>RtVariablen</key>
<dict>
    <key>CsrActiveConfig</key>
    <string>0x3F7</string>
```

Nur Bit 0x1 (ALLOW_UNSIGNED_KEXTS) spielt eine Rolle, wenn Sie unsigned die Kexte haben. Ich habe alle Kexte signiert und kann sie mit Null laden.

Ab Revision 4233 ist es im Clover-Menü nun möglich, das Laden einzelne Kexts aus den Ordnern /EFI/CLOVER/kexts/* zu blockieren. Auf diese Weise können Sie dort neue, fragwürdige Kexte ablegen, und versuchen Sie zu booten. Wenn es nicht klappt, können sie ihn beim Neustart mit diesem Kontrollkästchen einfach sperren. Und ab der Revision 5052 können Sie den fraglichen Kext in den Ordner /EFI/CLOVER/kexts/Off/ legen, z. B. VoodooPS2 von Vasya Pupkin, der verspricht, dass sein Kext besser ist als Ihrer. Und Ihr Kext befindet sich im Ordner OTHER. Nachdem Sie das Clover-Menü geladen haben, drücken Sie die Leertaste, und Sie sehen das -Menü um den Kext zu deaktivieren.

Der Standardwert, der für den Ordner "OFF" nicht zulässig ist, und der Standardwert, der für den Ordner "OTHER" zulässig ist, lautet standardmäßig: Sie aktivieren dieses neue VoodooPS2 testweise und deaktivieren das alte. Wenn das nichts funktioniert, dann funktioniert der alte wieder, wenn Sie neu starten. Wenn alles in Ordnung ist, und Vasya richtig liegt, dann ändern Sie manuell, wo der Cache liegen soll.

Entwicklung

Das Projekt hat aus lizenzrechtlichen Gründen keine kommerzielle Relevanz, und auch ist es zu groß, um es alleine zu machen, daher ist die sinnvollste Lösung es mit open source zu machen, und lassen Sie jeden, der etwas beitragen möchte, mitmachen. Bis 2019 Jahr basierte das Projekt auf dem sf.net-Server im Repository <http://cloverefiboot.sourceforge.net/>

Mit Version 2.5k sind wir in das neue Repository unter <https://github.com/CloverHackyColor/CloverBootloader> Es gibt auch kompilierte Versionen, die sofort einsatzbereit sind:

<https://github.com/CloverHackyColor/CloverBootloader/releases>

Eine lyrische Abschweifung. Die Erstellung eines Projekts, und sogar eines so großen, erfordert harte Arbeit zu den folgenden Punkten:

- Sammlung von Dokumentationen, Datenblättern, Spezifikationen, Softwaremustern für das gelieferte die Aufgabe, und weitere Informationen über die Hacker finden. Ich hatte eine guten Start Punkt - **Chameleon**, bei dem "alles" funktioniert (allerdings in Anführungszeichen!). Stimmt, die Zeit geht weiter, und neue Prozessoren, neue Grafikkarten, neue Anforderungen für neue Versionen von OSX, und wir müssen erneut nach Beschreibungen suchen oder neue Tests durchführen.

Der Ausgangspunkt wurde von Kabyl systematisiert, verbeugen Sie sich zunächst vor ihm.

4 Open-Source-Projekte haben dennoch einige Einschränkungen in der Nutzung, um es so auszudrücken "Lizenzierung". Die GPL-Lizenz ist lästig, Intel empfiehlt ihre Verwendung nicht. Clover wird als BSD geliefert.

- Wenn es eine Problemstellung gibt, Eingabedaten und was man am Ausgang erhalten möchte, ist es erforderlich, einen Algorithmus zu schreiben, möglichst kompakt und schnell, vorzugsweise fehlerfrei und sicher. Dies ist die Art von Aufgabe, die Programmierer lieben, und die meisten Bögen in diesem Projekt werden an solche Assistenten vergeben. Ich möchte vor allem um Dmazar und Geek hervorzuheben, sie haben in den frühen Tagen einige wirklich harte Sachen gemacht des Bootloaders.
- Es ist Zeit für die mühsame und anstrengende, aber unkomplizierte (?) Arbeit des Schreibens von tausenden von Codezeilen, bei denen man nicht viel nachdenken muss, Copy-Paste mit Korrekturen. Und hier ist die Sache, Es tut mir leid, ich habe seit drei Jahren fast keine Hilfe mehr. Ihr Projekt sind Sie und Sie arbeiten sich den Arsch ab. Nur, dass Apianti es irgendwie mit einem Unterschied gemacht hat, ein besonderer Dank an ihn. Solche rhetorische Frage: wenn ich die Quellen von jemand anderem für mein Projekt adaptiere, Wie hoch ist der Beitrag, den ich leiste? Ich meine, wir schreiben sowieso von Proben...
- Dann kommt die Arbeit des Testens und der Fehlererkennung. Ich habe eine Menge von Assistenten, ich kann mich gar nicht an alle erinnern. 20.000 Beiträge allein auf applelife.ru. Testen kommt in vielen Formen vor, vom einfachen Jammern, dass es nicht funktioniert, bis hin zu spezifische Anweisungen, was im Code geändert werden muss. All dies ist nützlich, sogar wenn jemand jammernd, bringt es Sie dazu, darüber nachzudenken, wie Sie das Problem beseitigen können (ob Sie das Problem lösen oder nicht, das Gejammer muss aufhören!)
- Die schlimmste Art von Fehler, ist ein prinzipielles Problem. Die ersten drei Probleme, die ich über ein halbes Jahr lang allein gelöst: 1. KP in Junior-Systemen, 2. Schlafmangel, 3. Kein Start am Laptop. Die folgenden Probleme wurden gemeinsam mit Dmazar und Pene gelöst: 4. iCloud, 5. iMessage. Und dann gibt es noch das Problem, dass Legacyboot herumhängt. Ich sehe keinen Freiwilligen, der daran arbeiten will, obwohl es diejenigen gibt, die das Ergebnis sehen wollen. 2018: über die Legacy-Boot-Systeme sind zu vergessen, sie werden nicht mehr gebraucht.
- Weitere erwähnenswerte Dinge: Kompilierungsskripte, Installer, System-Skripte und Anwendungen, die direkt mit dem Projekt zusammenhängen, obwohl sie nicht Teil des Laders sind. All diese riesige Arbeit liegt praktisch außerhalb meiner Kompetenz, auch wenn ich damit angefangen habe. Die Hauptakteure hier sind Jadran, Crasybirdy, JrCs, yeah. Und Apianti hat sich darauf eingelassen. Jetzt liegt es in der Kompetenz von vector-sigma.
- Jief Machak hat eine großartige Arbeit geleistet, indem er es in C++ übersetzt und ich habe mit und von ihm gelernt, wie man C++ richtig einsetzt.
- Nun und das Programm muss dokumentiert werden. Wiederum akribisch und nicht ein zu interessantes Werk. An dieser Stelle besonderen Dank an xsmile für die Übersetzung dieses Buches in Englisch, und als solches gelangte es in das WIKI, wo verschiedene Leute von Zeit zu Zeit mitwirken.

Ein separates Wort zu Dmazar. Es ist nicht nur ein Beitrag, es ist ein zweiter Autor. Clover ist So wie es jetzt steht, nicht meine Autorenschaft. Ohne Dmazar wäre es nur ein Vermächtnis.
Bootloader.

Dmazar hat einen UEFI-Bootloader erstellt. Er verfügt über die folgenden Technologien:

- OsxAptioFixDrv mit allen Optionen, sonst lädt UEFI macOS nicht. Beachten Sie, dass die Arbeit dieses Treibers ist für jeden Programmierer noch unverständlich, nur der Autor verstand es; dann kam vit9696, und er fand es heraus und machte eine neue Version von AptioMemoryFix, mit NVRAM-Unterstützung. Und jetzt ist es noch einmal umbenannt in OpenRuntime;
- DmpUefiCalls, ohne die wir nicht in der Lage wären, herauszufinden, was das MacOS braucht;
- EmuVariableUefi - für diejenigen, die kein echtes NVRAM haben, und war die Mehrheit. Ohne diesen Treiber auf einigen Konfigurationen wäre das Booten unmöglich.
- sowie das Arbeiten mit NVRAM innerhalb Clovers;
- ccj und Kernel-Patches. Ursprüngliche Quellen von pcj, der sie platziert hat und weggelaufen ist, und Dmazar hat es zu Ende gebracht, ohne sein Copyright darauf zu setzen;

- FSInject wird zum Patchen der Kexts benötigt. Auch hier weiß niemand mehr, wie es funktioniert;
- OsxFatBinaryDxe, wird benötigt für Systeme vor 10.8, da auf diesen Systemen boot.efi FatBinary gibt, was UEFI nicht versteht;
- Tiefschlaf. Wir arbeiteten alle, um es zu ermöglichen, aber es war Dmazar, der darauf hinwies, wie erreicht werden kann;
- und er hat mir tatsächlich geholfen, semantische Fehler in meinem Code zu korrigieren. Das ist richtig, Ein Mann ist dem anderen nicht gewachsen.

Die anderen Entwickler und Mitwirkenden werden auf die eine oder andere Weise in den Quellen und im installer erwähnt.

Ich schreibe dieses Kapitel in der Erwartung, dass es andere gibt, die bereit sind, an dem Projekt zu arbeiten, und müssen Sie zumindest lernen, wie man kompiliert.

Details

Um das Projekt zu kompilieren, benötigen Sie außerdem einen Compiler und Bibliotheken, das ist eine Binsenweisheit. Was braucht man in unserem Fall? Die Bibliotheken sind die Quellmodule von EDK2. Wie lautet der korrekte Name dafür? Rahmen? Umwelt? Auf Russisch sollte ich wahrscheinlich sagen

MITTEL.

Von demselben Server heruntergeladen. <http://sourceforge.net/projects/edk2/>

Da das gesamte Projekt für Hackintosh und für Hackintosh erstellt wurde, wird an erster Stelle die Kompilierung des Projekts in der MacOSX-Umgebung betrachtet. Dies ist jedoch nicht die einzige Fähigkeit. EDK2 selbst sieht die Kompilierung auch in Windows, Linux und in einige andere Systeme und Umgebungen vor. Für Windows benötigen Sie das Visual Studio 20xx, für Unter MacOSX müssen die Xcode Command Line Tools installiert sein und unter Linux ist der gcc-Compiler ist standardmäßig enthalten. Die eingebauten MacOSX-Werkzeuge reichen jedoch nicht aus, um Kompilierung des Projekts durch zu führen, daher wird, wie bei Linux, vorgeschlagen, eine neue Version von gcc, unter Verwendung des Skripts builgcc.sh im Ordner Clover. Warum nicht Klan? Denn Klan gibt immer noch keine brauchbaren Codes aus. Wir warten.

Wartet! Dank Zenith432 er hat endlich eine funktionsfähige Zusammenstellung mit Klan gemacht, in Verbindung mit der dieser Text entsteht Änderungen. Weitere Änderungen aufgrund der Tatsache, dass ich vorschlage, nicht permanent EDK2 ändern, und stabiler Brunch UDK2018. Ja, es ist genug. Allerdings ist die Option mit EDK2 ist ebenfalls vorhanden.

Ab der modernen 2.5k-Version hat sich auch hier eine dramatische Veränderung ergeben, daher sollte der vorherige Absatz als überholt betrachtet werden. Alle notwendigen Umgebungsbibliotheken heutzutage sind in Clover enthalten, und das Repository ist vollständig für die Kompilierung, keine externe Links werden benötigt. Und auch der gcc-Compiler wird wieder als die beste Option erkannt, jetzt Version 10. Die EDK2-Quellen existieren jedoch noch und werden verbessert, wir sollten sie ab und zu anzuschauen, sie mit den eigenen zu vergleichen.

Code schreiben

Wenn Sie Interesse haben, können Sie dem Programmiererteam beitreten und Ihren Teil beitragen. Wir gehen immer einen Schritt weiter, obwohl wir bestimmte Anforderungen an die Codes haben. Ursprünglich wurde das Projekt in der Sprache C erstellt und nach speziellen Regeln formatiert Tiano Code Style. <https://github.com/tianocore/tianocore.github.io/wiki/Code-Style-C>

Mit dem Übergang zu C++ wollen wir jedoch nicht so viele Makros verwenden, und lieber einen Standard-Programmierstil verwenden. Es sollte verstanden werden, dass Tianos Anforderung ist die Fähigkeit, Quellen auf jeder Plattform zu kompilieren, um mit den schwächsten C-Compilern arbeiten zu können. Und in der Tat ist EDK2 kein funktionierendes Projekt, es ist nur

einen Satz von Quellen, um Ihr Projekt zu erstellen. Aber wir machen ein echtes Projekt, und wir setzen unsere eigenen Regeln, für leistungsfähige Compiler und moderne Hardware.

Über C++.

1. Indem wir Strukturen und Funktionen in logisch vereinheitlichten Klassen zusammenfassen, vereinfachen wir den Code, und dadurch wird sie besser sichtbar. Außerdem stellt sich heraus, dass viele der Techniken, die verwendet werden, wiederholen, so dass es eine unkomplizierte Möglichkeit ist, Codes zu reduzieren. Code wiederverwenden.

2. Es ist nicht notwendig, Variablen im Voraus zu deklarieren, wie es die Sprache C verlangt. Eine Variable muss angekündigt werden, wo es verwendet wird. Erstens ist es visueller, es ist nicht nötig, durch den Text zu scrollen zurück, um die Deklaration zu sehen, zweitens kann der Compiler zusätzliche Optimierung, drittens werden Programmierfehler bei wiederholten Anzeigen reduziert, die nicht durch die Sprache verboten sind. Ein Beispiel ist die Schleife `for (int i=0; i<0; ++i)`. Im alten C war es unmöglich und daher in C verboten, aber für uns ist es besser, den Schleifenindex auf diese Weise zu deklarieren.

3. Allocate/Free-Alpträume können durch die Verwendung von Funktionen beseitigt werden Compiler auf automatische Konstruktor/Destruktor-Speicherverwaltung.

4. Verzichten Sie auf die Verwendung von Arrays zu Gunsten von Vektoren. Arrays steuern nicht außerhalb der Grenzen, während Vektoren einfach sind.

5. Verweigern Sie das Kopieren von Zeigern - die größte Quelle für Fehler im Projekt. Obwohl ermöglicht es Ihnen, etwas zu speichern. Verwenden Sie Links oder kopieren Sie Strukturen als Ganzes. Es scheint teuer zu sein, aber es zahlt sich durch sündhafte Arbeit aus.

6. Vermeiden Sie Übertreibungen. Das Schreiben von `(INTN)N` bedeutet einen logischen Fehler. Wenn die Variable `N` in einem vorzeichenbehafteten Ausdruck verwendet wird, d. h. der kleiner als Null werden kann, warum es als vorzeichenlos deklariert wurde? Und ähnlich verhält es sich mit Zeigertypen. C++ fördert nicht solche Typ-Substitution, und wir fordern, dass Typ-Substitution so selten wie möglich verwendet wird.

7. Verwenden Sie mehr konstante Variablen und Funktionen, d. h. geben Sie dem Compiler mitteilen, dass sich ihr Wert nicht ändern wird, der Compiler dann und der Code entsprechend das andere gilt. Und es wird eine Warnung geben, wenn ein Programmierer gegen das Verbot verstößt. Dies ist als Sicherheit bezeichnet und wird vom Compiler unterstützt.

8. Wir haben keine STL-Bibliothek, aber wir erstellen unsere eigenen Klassen und übersetzen alle Quellen in ihre verwenden.

9. C-Code Style impliziert die Verwendung von `CONST`, `VOID`, `STATIC` geschrieben in Großbuchstaben. Warum? Um die Sprache mehr wie Fortran oder so zu machen?! Wir sind neigen zu mehr Standard-Schreibweise `const`, `void`, `static`, und so weiter.

Compilation

Nun zur Sache. Der Leser, der in dieses Kapitel schaut, kann per Definition nicht ein Benutzer sein, dem man erklären muss, wie man das Terminal verwenden soll.

1. Laden Sie die Quellen herunter.

```
cd ~
mkdir src
git clone --recurse-submodule https://github.com/CloverHackyColor/CloverBootloader.git cd
CloverBootloader
```

2. Kompilierwerkzeuge zusammenstellen

```
./buildgettext.sh
```

```
./build_gcc10.sh
./buildnasm.sh
```

Es ist jedoch möglich, NASM nicht selbst zu bauen, sondern einfach die neueste Version von der offiziellen Website <http://www.nasm.us/pub/nasm/releasebuilds/>, derzeit die Version 2.15.05, und legen Sie es im Ordner `~/src/opt/local/bin/` ab. Wenn Sie etwas davon vergessen, erledigt das Clover-Kompilierungsskript dies für Sie.

3. Das Compilieren von Clover wird entweder teilweise oder vollständig durch skripte erledigt:

```
./buildme
```

Diese werden Sie genau fragen, was Sie wollen. Vector-sigma hat dort ein Menü mit Auswahlmöglichkeiten aus dem Test bereitgestellt

Builds zu einer Vollversion.

Obsolete!

1. Laden Sie die Quellen herunter und bereiten Sie die Umgebung vor:

```
cd ~
mkdir src
cd src
git clone https://github.com/tianocore/edk2 -b UDK2018 --depth 1 UDK2018 cd UDK2018 svn co
svn://svn.code.sf.net/p/cloverefiboot/code/ Clover make -C BaseTools/Source/C cd Clover
```

Oder verwenden Sie den traditionellen EDK2. Sie entwickelt sich auf ihre eigene Weise weiter, während Clover verwendet Funktionen, die schon seit langem entwickelt und abgestimmt wurden, also ist es ausreichend, UDK2018 zu verwenden, aber wenn Sie möchten, können Sie dies auch mit einer bestimmten Revision tun, oder mit letzterem, wenn bekannt ist, dass es mit Clover funktioniert:

```
svn co -r 18198 https://svn.code.sf.net/p/edk2/code/trunk/edk2 edk2 svn co
https://svn.code.sf.net/p/edk2/code/trunk/edk2 edk2
```

2. Kompilierwerkzeuge zusammenstellen

```
./buildgettext.sh
./buildgcc-4.9.sh
./buildnasm.sh
```

Jetzt wird gcc nicht besonders benötigt, Clover kompiliert gut mit eingebautem Klang, und Sie müssen Xcode ohnehin installiert haben, daher ist der Compile-Schalter `-t XCODE8` ist jetzt die Standardtaste. NASM benötigt mindestens die Version 2.13.03, am besten die neueste, sie scheint bereits im System vorhanden zu sein, aber eine alte Version, so dass es entweder mit dem obigen Skript gebaut werden muss, oder einfach Laden Sie die fertige Version hier herunter: <http://www.nasm.us/pub/nasm/releasebuilds/>

Die korrekte Version sollte hier stehen

```
~/src/opt/local/bin/nasm
oder einfach /opt/local/bin/
mkdir -p /opt/local/bin/
sudo cp ~/Downloads/nasm-2.13.03/nasm /opt/local/bin
```

Um mit dem Klang zu kompilieren, benötigen Sie auch das MTOC-Dienstprogramm, das Sie selbst aus dem aktuellen den Quellcode,

```
./buildmtoc.sh
oder nehmen Sie einfach die vorgefertigte Datei, die mit den Clover-Quellen im
Clover/BuildTools/usr/local/bin/mtoc.NEW.zip entpacken, und in /usr/local/bin/mtoc.NEW einfügen
```

Wird nicht mehr geliefert, wenn Sie es noch nicht haben, bauen Sie eine Version mit dem Skript, oder eine vorgefertigte Version aus dem Internet ausleihen. Sie muss nicht aktualisiert werden. Niemand hat jemals beobachtet, dass die eine Version anders funktioniert als die andere.

3. die Anpassung der EDK2-Umgebung an unsere Bedürfnisse durch Auswahl der entsprechenden Patches

```
cd ...
./edksetup.sh
cd Clover
./buildgettext.sh
cp -R /Patches_für_UDK2018/* ../
```

4. jetzt können Sie CloverEFI selbst bauen. Zum Beispiel so:

```
./ebuild.sh
./ebuild.sh -mc --no-usb
./ebuild.sh -ia32
Standardmäßig ist es -x64 und --xcode8, wenn Sie mit gcc kompilieren wollen, dann
./ebuild.sh -gcc49
oder auch -gcc53, oder -t XCODE5
```

Es wurden weitere Kompilierungs-skripte erstellt, die in der Regel selbstdokumentiert sind. Siehe, wählen, verwenden.

5. Eine Spitzfindigkeit. Das Repository enthält nicht die HFSPlus.efi-Dateien für 32 und 64 Bit, da sie Privatsphäre. Die beiden Möglichkeiten sind, diese Dateien aus anderen Quellen zu beziehen, oder mit frei VboxHfs. Dieser Treiber ist funktionsfähig, aber etwas langsamer. Es war Unterstützung fremder Dateisysteme

```
#INF Clover/VBoxFsDxe/VBoxHfs.inf
INF RuleOverride=BINARY Clover/HFSPlus/HFSPlus.inf
Make (dies ist die Voreinstellung)
Unterstützung fremder Dateisysteme
INF Clover/VBoxFsDxe/VBoxHfs.inf
#INF RuleOverride=BINARY Clover/HFSPlus/HFSPlus.inf
```

6. Ebenso fehlt apfs.efi, aber unser Gegenstück existiert nicht, Sie müssen vom System 10.13+ von Ordner `usr/stanadlone/i386/apfs.efi`, oder `ApfsDriverLoader.efi` verwenden

7. Das Projekt steht nicht still, daher kann dieses Handbuch irgendwann wegen einer kleinen Änderung unbrauchbar. Dieses Projekt ist für diejenigen, die denken, die kann herausfinden, was falsch ist und was zu tun ist.

8. Jetzt den Installer zusammenbauen

```
cd ~/src/edk2/Clover/CloverPackage/
./makepkg
./makeiso
```

Alles erledigt! Einige Schritte können weggelassen werden, wenn Sie es für sich selbst und nicht zum ersten Mal machen Zeiten. Anstatt das gesamte Projekt herunterzuladen, aktualisieren Sie zum Beispiel nur `svn up` oder besser `./update.sh` (einschließlich EDK2-Update, aber nicht benötigt für UDK2018), ausschließen 32-Bit-Kompilierung und nicht zum Erstellen eines Pakets und eines Disk-Images.

Für Dummies sind vollautomatische Skripte vorbereitet CloverGrower und seine Pro-Version. Das ist lächerlich. Ein Skript, mit dem ein Dummkopf das Paket selbst kompilieren kann. Dummie ist es besser, das fertige Installationsprogramm herunterzuladen.

NB: Mit den Revisionen 3000+ wurde eine neue Kompiliermethode LinkTimeOptimization eingeführt, die wesentlich Reduzierung der Codegröße. Compiler gcc-4.8 ohne es, 4.9 mit es.

Die Größe ohne ist 675520, mit Optimierung ist 633536, und es funktioniert auf die gleiche Weise, nur schneller.

Das ist für den Klan nicht notwendig.

Ab Revision 3059 war es möglich, die Größe der Boot-(CloverEFI)-Datei durch Verschieben von Translation Page Table von Adresse 0x90000 auf Adresse 0x88000, Dateigröße war 483Kb, wurde 450Kb, und dennoch arbeitet es auf einigen Chipsätzen sicherer. In der Tat, es hat Zenith432 hatte das Problem, dass Clover auf seiner Buche nicht funktioniert, er fand eine Überschneidung Adressen. Und da mit LTO die Codegröße erheblich gesunken ist, ist es möglich geworden, TRT zu bewegen. Jetzt lässt er auch Clover arbeiten!

Compilation mit dem Befehl:

```
./ebuild.sh -low-ebda
```

Dieser Schlüssel `undo` ist jetzt der Standard, der Schlüssel `-std-ebda` ist erforderlich, um ihn zu überschreiben. Für Entwickler ist es auch interessant zu wissen, wo der Buchstabe "B" oder "6" gebildet wird, der wird angezeigt, wenn CloverEFI startet. Verwenden Sie einen Hexadezimal-Editor, um den Offset zu suchen 0x02a9, dort sehen wir Byte 0x42 bzw. 0x36. Übrigens, um diesen Brief zu stornieren kann nicht, da es ein Trick ist, Clover überhaupt zum Starten zu bringen. Das BIOS erzeugt Initialisierung des Videoadapters an dieser Stelle. Sie können es auf Leerzeichen 0x20 ändern.

Erstellen einer DEBUG-Version des Clover

[Making a DEBUG version of Clover](#)

Wenn sich Leute darüber beschweren, dass Clover nicht startet und daher nicht funktioniert, kann ich nicht helfen, aber ein paar Standardtipps zu geben. Ich bin ein Mann, der kann einfach den Code bearbeiten und ein wenig kompilieren, kann zu Punkt X kommen, und dann werden wir zusammen festzustellen, was los ist.

Option 1: Wenn der Legacy-Boot bei Meldung 6_ hängen bleibt. Dies bedeutet, dass Sie im Inneren von Duet selbst hängen. Es ist nicht möglich, ein Protokoll zu erstellen. Es ist nur möglich, die Anzeige von Meldungen auf dem Bildschirm zu beobachten. Der Ablauf des Duettbetriebs ist wie folgt:

1. Startsektor `st32_64.s`. Genau an dieser Stelle wird 6 ausgegeben, lesen BIOS-ROM, A20-Umschaltung und CPU-Umschaltung von 16 Bit auf 32 und 64.

Meine Version des Sektors unterscheidet sich von der Vanilla-Version darin, dass sie auf einem Laptop läuft. Das ist nicht ausgeschlossen, dass es auch hier für jemanden ein Problem geben wird. Es ist notwendig, den Ausgang anderer Nummern an anderen Stellen einzufügen, und beobachten.

2. Beginn der C-Codes. Datei `efi32.s` oder `efi64.s`. Es ist unwahrscheinlich, dass sich hier etwas verlangsamen kann.

3. Datei `Efiloader.c`. Hier können Sie die Ausgabe auf den Bildschirm einfügen mit der Prozedur `PrintHeader('A')`; die derzeit auskommentiert ist.

4. Datei `Dxelpl/Dxelnit.c`. Sie können auch `PrintHeader` einfügen, aber trotzdem `Debug.c` in um dieses Modul zu kompilieren.

5. `DxeCore`. Die Ausführung ist hier schon ausufernd und es ist schwieriger nachzuvollziehen, was und wo passiert.

Fazit
zum Bildschirm kann auf die gleiche Weise erfolgen.

6. `CLOVERX64.EFI` selbst wird in der Prozedur `BdsBoot.c/BdsLibBootViaBootOption()` geladen.

An dieser Stelle im Programm können Sie bereits die Standard mit der Prozedur `AsciiPrint("\nasty!\n");`

Option 2. Duet selbst funktioniert, was Sie überprüfen können, indem Sie die Leertaste unmittelbar nach Pin 6 auf Bildschirm. Entweder haben wir eine UEFI>Last, und es gibt kein Duet, dann sollten wir CloverGUI, und es gibt keine, oder es gibt eine, aber sie bleibt hängen.

Die Standard-Methode `Boot->Debug=true` funktioniert nicht, da Sie den Speicherort von mehr Details.

Entfernen Sie in diesem Fall in der Datei `Platform.h` den Kommentar in Zeile 11

```
//#define DEBUG_ALL 2
```

oder in den Dateien, die Sie interessieren, in den oberen Zeilen `DEBUG_xxx 2` ein. In diesem Fall wird der gesamte Ausgang von `DBG("Nightmare #3\n");` wird auf den Bildschirm übertragen. Und auf diese Weise wird es möglich sein interaktiv zu beobachten, wohin die Programmausführung gelangt, bevor sie hängen wird.

ZS. Verwenden Sie keine russischen Buchstaben, wie ich hier illustriert habe! Es funktioniert nicht. C durch Umschalten auf C++ können russische Buchstaben im Debug verwendet werden

Option 3. Kompilieren der Debug-Version mit Haltepunkten (stop points), Ausführen unter Kontrolle von QEMU mit spezieller Version von gdb installiert. Dmazar hat diesen Weg einmal ausprobiert. Doch meiner Meinung nach sind diese Bemühungen nicht den Aufwand wert. Ein einfacher Trace ist immer ausreichend.

Option 4. Geben Sie beim Kompilieren `./ebuild.sh -D DEBUG_ON_SERIAL_PORT` und schließen Sie an die serielle Schnittstelle einen anderen Computer an, der in der Lage ist, Briefe in serielle Schnittstelle zu empfangen (Terminal in Windows). Die Variante funktioniert unter QEMU, wenn Sie beim Starten angeben

"-serial stdio"-Flag

Installation

Verwendung des Installationsprogramms

Wofür ist das Installationsprogramm?

So installieren Sie das Programm! Warum sollten Sie es manuell tun? Der Installer wird alles genauer machen als Sie! Die einzige Voraussetzung ist, dass Sie auf dieser Computer bereits MacOSX haben. Eine Möglichkeit ist, dass Sie die Installation DVD mit einem anderen Bootloader und führte das Installationsprogramm über die MacOSX-Installationsoberfläche aus.

B

Abhängig von der Sprache des Betriebssystems läuft das Installationsprogramm in Russisch, in Englisch oder sogar in Chinesisch. Hier sind die Anweisungen für die englische Version, da auf Russisch wie es ist. Sie können es herausfinden, und ich kann auch kein Chinesisch. Die aktuelle Version umfasst 20 Sprachen, darunter Indonesisch, falls jemand es braucht.

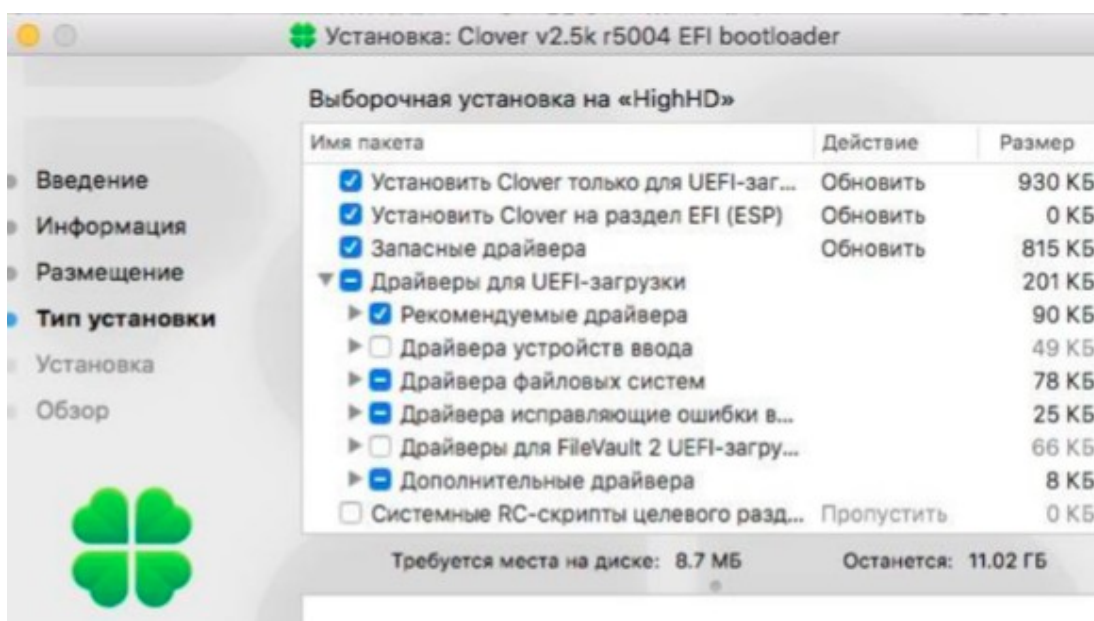


Ok.,

Folgen Sie der Taste Weiter (Continue) und OK und lesen und akzeptieren Sie die Lizenzvereinbarungen (hmm, gibt es die?), und kommen zu der Entscheidung, was wir wo und warum installieren.



Installationsort ändern - wählen Sie aus, wo der Bootloader installiert werden soll. Wenn Sie beabsichtigen auf einer EFI-Partition zu installieren und wählen Sie dann nur die Partition mit dem aktuellen System aus. MacHDD in diesem Beispiel. Kontrollkästchen auf "Clover im ESP installieren". Das Installationsprogramm findet die EFI-Partition auf demselben Laufwerk.



Аangepasste oder Standardinstallation - Wählen Sie die Download-Optionen. Wenn Sie den Cursor auf einer der Zeilen platzieren, erscheint im unteren Feld eine kurze Beschreibung dieser Option. Clover nur für UEFI-Boot installieren - diese Option übersteuert die Installation von Dateien Stiefel. Die englische Version bedeutet "Install for UEFI motherboard", und natürlich werden die Benutzer, das Kästchen ankreuzen. Das Ergebnis ist, wie immer, dass Clover nicht geladen wird. Ich bedauere sehr, dass die Entwickler des Installers diesen Punkt für ihre eigenen Zwecke erfunden haben.

Leute, um Himmels willen, aktivieren Sie dieses Kästchen nicht, auch wenn Sie einen Computer mit UEFI-BIOS haben. Der gewünschte UEFI-Boot wird trotzdem funktionieren!

Die Installation von Clover im ESP ist die beste Option, wenn es eine solche Partition gibt (GPT-Partitionsschema). Das Installationsprogramm kann diese Partition nicht sehen, Zeigen Sie also im Festplattenauswahlmenü auf die Partition, die sich auf derselben Festplatte befindet, auf ESP auf den wir den Bootloader legen wollen. Wir nehmen an, dass sich auf dieser Partition MacOSX befindet, wo die Skripte, das Control Panel und der Updater installiert werden (zu lang auf Russisch) "Automatisches Update-Programm").

Die ESP-Partition sollte vor der Installation ausgehängt werden. Auch für Catalina kann Sie müssen den Befehl eingeben, bevor Sie das Installationsprogramm ausführen

```
sudo mount -uw /
```

die es Skripten ermöglicht, Dateien in Systemordner zu schreiben. Wenn das System schreibt, dass der Installer nicht geöffnet werden kann, weil sein Autor deinstallierten Entwickler, können Sie in den Systemeinstellungen, Sicherheit, und drücken Sie die Schaltfläche "Trotzdem öffnen". Oder Sie können den Befehl im Terminal ausführen

```
sudo spctl --master-disable
```

Es ist jedoch nicht meine Aufgabe, Ihnen die Tricks der macOS-Nutzung beizubringen.

Bootloader ist eine Variante mit dem BIOS (Variante A), die CloverEFI, oder mit UEFI (Option B).

- MBR- und PBR-Sektoren nicht aktualisieren - Sektoren nicht aktualisieren, da sie bereits gibt, oder nur für Option B;

- boot0af im MBR installieren - Booten mit boot0af, d. h. Suchen des aktiven Teilung. Das Installationsprogramm macht die ausgewählte Partition aktiv. Die Ausnahme ist die Installation von auf eine EFI-Partition, wird diese nicht aktiv gemacht, und boot0af findet keine aktive Partition die Boot-Datei von der EFI-Partition laden, was wir für das Vermächtnis benötigen Booten Sie von der GPT-Platte, von der ESP-Partition.

- Installieren Sie boot0ss im MBR - booten Sie mit boot0ss, d.h. suchen Sie nach einer HFS+-Partition, auch wenn sie inaktiv ist. Das Installationsprogramm ändert die aktuell aktive Partition nicht. Dies ist ist für eine Konfiguration mit einer aktiven Windows-Partition gemacht - er braucht sie.

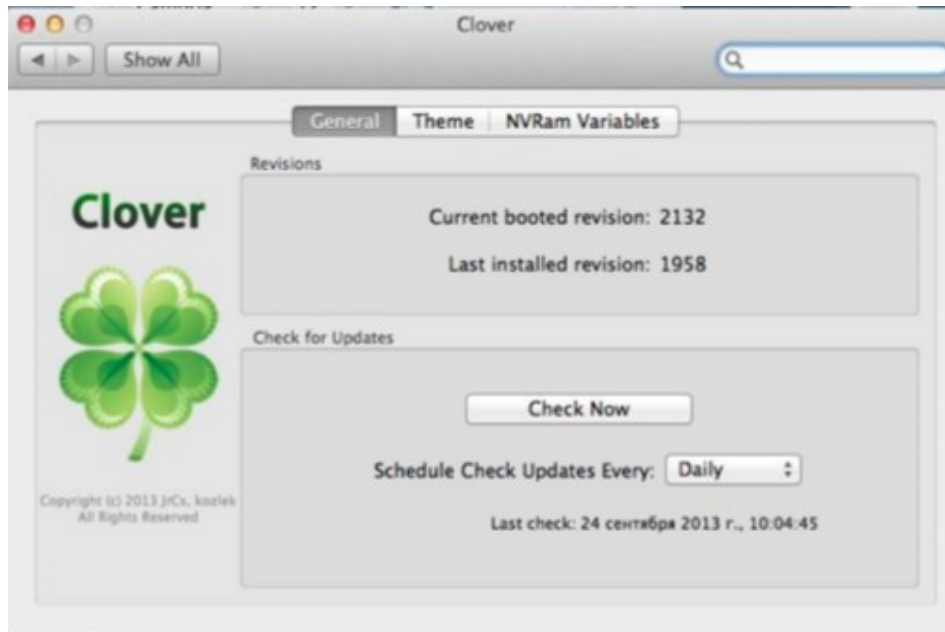
Alternatives Booten PBR verwenden - wie in dem Kapitel "Was ist was", kann der PBR-Sektor für 2 Sekunden angehalten werden, um die Tasten 1-9 zu drücken, oder ohne sie. Mit dieser Option wird der Sektor mit Pause eingestellt. Diese Option ist nicht mehr verfügbar in der Installateur ;)

CloverEFI ist, wie Sie aus der Liste ersehen können, die Bitratenauswahl des Bootloaders. Entweder 32 Bit oder 64 Bit. Auch hier gibt es eine spezielle Variante von BiosBlockIO. Dies ist eine solche Variante von CloverEFI-64, das einen speziellen Namen hat, nämlich boot7, und für Computer gedacht ist, die einen nicht standardisierten SATA-Controller. Dieser Treiber läuft durch das BIOS, und normalerweise, funktioniert mit jedem Controller (BIOS muss mit ihnen arbeiten!). Aber es gibt auch Fehlzündungen, wie zum Beispiel das Dell Inspiron 1525. Eine weitere spezielle Variante ist boot7-MCP79. Es ist so ähnlich wie Die einzige praktikable Option für den MCP79-Chipsatz, obwohl es dort nichts Spezifisches gibt für diesen Chipsatz, kann auch in anderen Fällen funktionieren. Diese Option wurde von Oscar09 gefunden, auch der einen USB-Patch für diesen Chipsatz vorgeschlagen hat.

Treiber. Die Auswahl der Treiber wird im obigen Kapitel "Was ist was" erläutert. Installieren Sie die RC-Systemskripte auf der Hauptpartition - dies sind die rc.local und rc.shutdown.local, die vom OSX-System bei der Anmeldung und Abmeldung ausgeführt werden - eine notwendige Teil des gesamten Legacy-Clover-Konzepts. Für diejenigen, die ein funktionierendes NVRAM für UEFI haben Downloads, sind diese Skripte nicht notwendig.

Installieren Sie die Skripte auf allen anderen Partitionen - wenn es mehr als eine MacOSX-Partition auf dem Computer gibt. Das Installationsprogramm ist schlau genug, sie nicht auf Windows-Partitionen oder Linux Partitionen abzulegen.

Installieren Sie das Clover-Bedienfeld - dieses Bedienfeld hilft bei der Aktualisierung von Clover, der Themenauswahl und das Setzen von NVRAM-Variablen:



Manuelles Installieren des Bootloaders

Wird in zwei Fällen benötigt: bei Flohbefall und bei Durchfall. **Erstens**, wenn eine Person gut bescheid weiß, was er tut und er jeden Schritt kontrollieren will, dem Installer nicht glauben (und das ist vergebens!), und, **zweitens**, wenn die Installation von einem anderen Betriebssystem aus erfolgt, auf dem das Installationsprogramm nicht ausgeführt werden kann.

OSX

Es wird dringend davon abgeraten, dies bei jemandem zu tun, der nicht weiß, was ein Terminal ist.

Installation auf einer HFS+-Partition in MBR oder hybrider Partitionierung. Warum MBR? Es ist eine oft zu findende Standardsituation, wenn ein Computer bereits vorhanden ist, und bereits mit Informationen darauf sind, die sie nicht zu verlieren wollen, dann können Sie nur einen neuen Bootloader installieren.

MBR-Sektor einrichten

```
cd BootSektoren
```

```
sudo fdisk440 -f boot0 -u -y /dev/rdisk0
```

Was bedeutet das?

fdisk440 ist eine spezielle Version des Dienstprogramms fdisk, die so korrigiert wurde, dass sie nur 440 Byte Null-Sektor korrigiert, damit es für die Windows-Kompatibilität funktioniert (Aufwachproblem), um das sich Apple nicht gekümmert hat.

boot0 - die oben im Kapitel "Was ist was" beschriebene Datei

rdisk0 ist das physikalische Gerät, auf dem Sie den Bootloader ablegen werden. Vergewissern Sie sich, dass es tatsächlich die Zahl 0 hat.

Diese Dateien werden mit Clover mitgeliefert. In der neuen Version ist fdisk440 nicht mehr dabei, in erster Linie, weil sie in der neuen Umgebung nicht kompiliert werden kann. Verwenden Sie nun dd stattdessen, wie für Linux gezeigt. Und Sie können die Partition aktiv machen mit dem Standard fdisk. Für die GPT-Partitionierung wird sie jedoch nicht benötigt.

Installation des PBR-Bereichs:

```
sudo dd if=boot1h2 of=/dev/rdisk0s9
```

boot1h2 - PBR-Sektordatei für HFS+-Dateisystem, anders als bei ähnlichen Unterstützung für große Boot-Dateien und die Möglichkeit, boot1,3,6 per Hotkey auszuwählen.

Einzelheiten finden Sie im Kapitel "Was ist was".

rdisk0s9 ist die neunte Partition auf dem ausgewählten Gerät... Warum die neunte? So dass Narren nichts durch stupides Wiederholen von geschriebenen Befehlen vermessen können. (Es gibt es wahrscheinlich keinen solchen Abschnitt.) Aber Sie sollten die reale Zahl angeben, z. B. die erste Partition.

Nun, nachdem die MBR- und PBR-Sektoren erfolgreich in den ausgewählten Gerät/ausgewählte Partition sind, sollten Sie diese Partition aktiv machen

```
fdisk440 -e /dev/rdisk0
```

```
>f 9
```

```
>w
```

```
>q
```

Die Neun in der zweiten Zeile ist wieder die Abschnittsnummer (es gibt nur vier davon!) - leiten Sie sie ab. Sie können nun die Boot-Datei und den EFI-Ordner in das Stammverzeichnis der Partition kopieren.

Installieren Sie auf einer FAT32-Partition.

Im Gegensatz zur vorherigen Methode gibt es hier eine Feinabstimmung. Der PBR-Bereich muss enthalten Trenngeometrie. Diese Informationen werden dort während der Partitionierung eingetragen, so dass der Verlust solcher Informationen ist mit Konsequenzen behaftet ist. Die Methode der Installation des Sektors selbst ist kompliziert

```
dd if=/dev/rdisk1s9 count=1 bs=512 of=origbs cp boot1f32alt newbs
```

```
dd if=origbs of=newbs skip=3 seek=3 bs=1 count=87 conv=notrunc dd if=newbs of=/dev/rdisk1s9 count=1 bs=512
```

boot1f32alt - bereits im Kapitel "Was ist was" erwähnt - ist ein Sektor zur Installation auf einer FAT32-Partition. Aber das ist nicht FAT16! Seien Sie vorsichtig!

rdisk1s9 - wiederum die neunte Partition auf dem ersten Gerät. Geben Sie Ihre konkrete Zahlen ein. Der Rest sind die Buchstaben und Zahlen, die in dieser Vorschrift sind nicht verändert werden dürfen.

Der Rest der Schritte ist ähnlich wie bei der Installation auf HFS+.

Für Besitzer von Festplatten mit 4k-Sektorgröße gibt es eine Warnung!

Im ersten und vierten Befehl müssen Sie anstelle von bs=512 bs=4096 schreiben.

ZS. Allerdings... das ist keine Tatsache! Bei mir hat eine 512er-Installation mit einem 4k-Laufwerk problemlos funktioniert.

Installieren auf einer exFAT-Partition

Ab Revision 3040 ist es nun möglich, Clover auf einer exFAT Format-Partition zu installieren. Ich habe zum Beispiel eine externe Festplatte mit diesem Format. Da ich sie nicht mehr neu partitionieren kann, und auf dem alten Weg gibt es keinen Platz für die Installation von Clover gibt. Die Installation eines PBR-Sektors erfordert ein spezielles boot1-Installationsprogramm, geschrieben von Zenith432. Der Befehl ist einfach

```
./boot1-install -u -y /dev/disk2s1
```

Grundsätzlich hat Zenith mit diesem Dienstprogramm sowohl HFS+ als auch FAT32 installierbar gemacht. Wichtig ist, dass die entsprechenden Sektoren (boot1h, boot1f32, boot1x) im gleichen Ordner liegen. Bei exFAT heißt der Sektor boot1x. Die Variante boot1xalt ist veraltet.

Es gibt jedoch noch eine weitere Überlegung bei der Installation dieser Sektoren. Das Volume, auf dem Sie den Sektor installieren möchten, müssen Sie ihn zuerst aushängen! Und dies ist nur möglich, wenn die Platte nicht die System-Platte ist. Andernfalls booten Sie von einem anderen Medium, um einen solchen Vorgang durchzuführen.

Linux

Unter Linux gibt es auch ein Terminal, und fast die gleichen Befehle, aber die Installation ist nur auf FAT32 möglich. Die Unterschiede sind wie folgt.

- anstelle von rdisk1 wird sdb verwendet - siehe Ihre Linux-Version für Details.
- Verwenden Sie das gleiche dd anstelle von fdisk440, um den MBR zu schreiben

```
dd if=dev/sdb count=1 bs=512 of=origMBR
cp origMBR newMBR
dd if=boot0 of=newMBR bs=1 count=440 conv=notrunc
dd if=newMBR of=/dev/sdb count=1 bs=512
```

Windows

Unter Windows ist es auch sinnvoll, den Bootloader nur auf das FAT32-Flash-Laufwerk zu legen, denn das führen Sie einfach mit dem Skript aus:

makeusb.bat E:

wobei E: der Buchstabe Ihres Flash-Laufwerks ist. Es soll nicht mehrere Partitionen auf ihm haben.

Das ist Windows!

Alle für das Skript benötigten Dateien sind im Lieferumfang von Clover enthalten. Allerdings, das Installationsprogramm muss zuvor entpackt werden, oder Sie können diese Dateien direkt von der svn herunterladen. Sie befinden sich im Ordner edk2/Clover/CloverPackage/CloverV2/BootSectors

Nach dem Ausführen des Skripts muss das Flash-Laufwerk ausgeworfen und wieder eingesteckt werden, dann wird es nach die Boot-Datei und den EFI-Ordner installieren. Noch besser ist es, BootDiskUtility.exe von Cvad zu verwenden, für die Erzeugung eines Windows-Flash-Laufwerk.

<http://www.applelife.ru/threads/delaem-zagruzochnuju-clover-fleshku-s-macosx-iz-windows.37189/>

Empfohlene Installationsoptionen

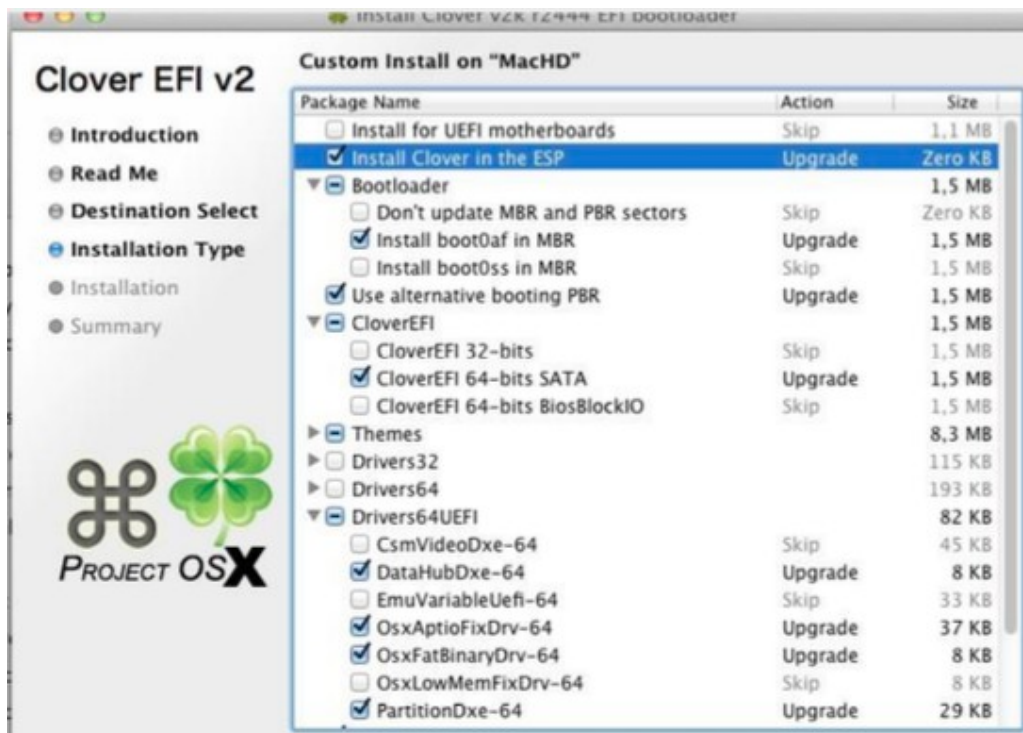
Dies sind die Optionen, die ich persönlich für die Installation empfehlen würde:

1. die primäre Festplatte ist in GPT (GUID Partition Table) partitioniert. Dies ist ein modernes Schema, das korrekt ist für das MacOS, und wird von Windows seit 7 unterstützt. In diesem Schema gibt es eine unsichtbare ESP-Partition (EFI System Partition) von 200 MB. Standardmäßig ist es bereits in FAT32 formatiert und muss nicht neu formatiert werden, ist aber auch nicht schädlich. Das ist der Ort, wo Clover sitzen sollte, sowohl für Legacy- als auch für UEFI-booten.

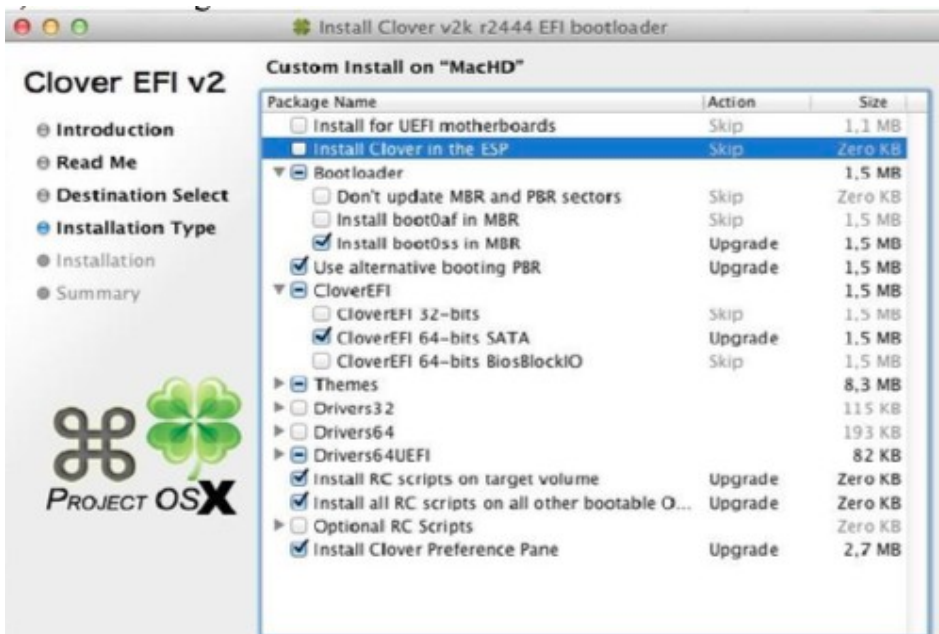
Für den UEFI-Boot müssen Sie nur den Ordner kopieren (oder mit dem Installationsprogramm installieren), die EFI mit allen notwendigen Inhalten. Aber damit das BIOS diese Boot-Option sieht müssen Sie die Datei /EFI/CLOVER/CLOVERX64.EFI nach /EFI/BOOT/BOOTX64.EFI kopieren, wenn sie nicht vorhanden ist. Für diejenigen, die es nicht verstehen, machen Sie es in zwei Schritten: Kopieren in den Ordner /EFI/BOOT/, und benennen Sie ihn dann in BOOTX64.EFI um. Um Legacy-Boot durchzuführen, müssen Sie die Datei boot0af in den MBR-Sektor dieser Festplatte und in den PBR dieser EFI-Partition die Datei boot1f32 (oder boot1f32alt, die mit einer mit Pause startet) schreiben.

Machen Sie niemals eine Partition aktiv!

Dell-Laptop hat American Megatrend EFI BIOS, und es erlaubt nicht, das Legacy Booten auf eine GPT-Platte. Leider! Für andere ist das möglich, und im Installationsprogramm tun Sie dies: (der Screenshot ist nicht mehr aktuell, aber die Bedeutung bleibt)



2. Die Festplatte ist bereits als FDISK-Partitionierungsschema partitioniert, das allgemein als MBR bezeichnet wird. Altes Schema, die schon immer für Windows XP verwendet wurde. Auf diesem Laufwerk haben wir eine Partition zugewiesen, auf der wir OSX installieren, um es auszuprobieren. UEFI-Bootung kommt hier nicht in Frage, aber Legacy, Wir könnten hier auch Clover verwenden. Dazu schreiben Sie die Datei boot0ss in den MBR-Sektor. In diesem Sektor wird nach einer HFS+-Partition gesucht, die wir für den Mac gemacht haben, und die ihm die Kontrolle gibt. Die aktive Partition kann für Windows bleiben, es braucht es mehr. Sie müssen die Datei boot1h in den PBR-Sektor dieser Partition schreiben aus dem Clover-Bausatz. **Ich erinnere Sie noch einmal daran, dass Sie, wenn Sie zuvor mit Chameleon gearbeitet haben, ist dort die gleiche Datei aus dem Chameleon-Bausatz vorgeschrieben, und es wird nicht mit Clover arbeiten. Wir müssen es ersetzen!** Die folgenden Kontrollkästchen sind im Installationsprogramm vorhanden

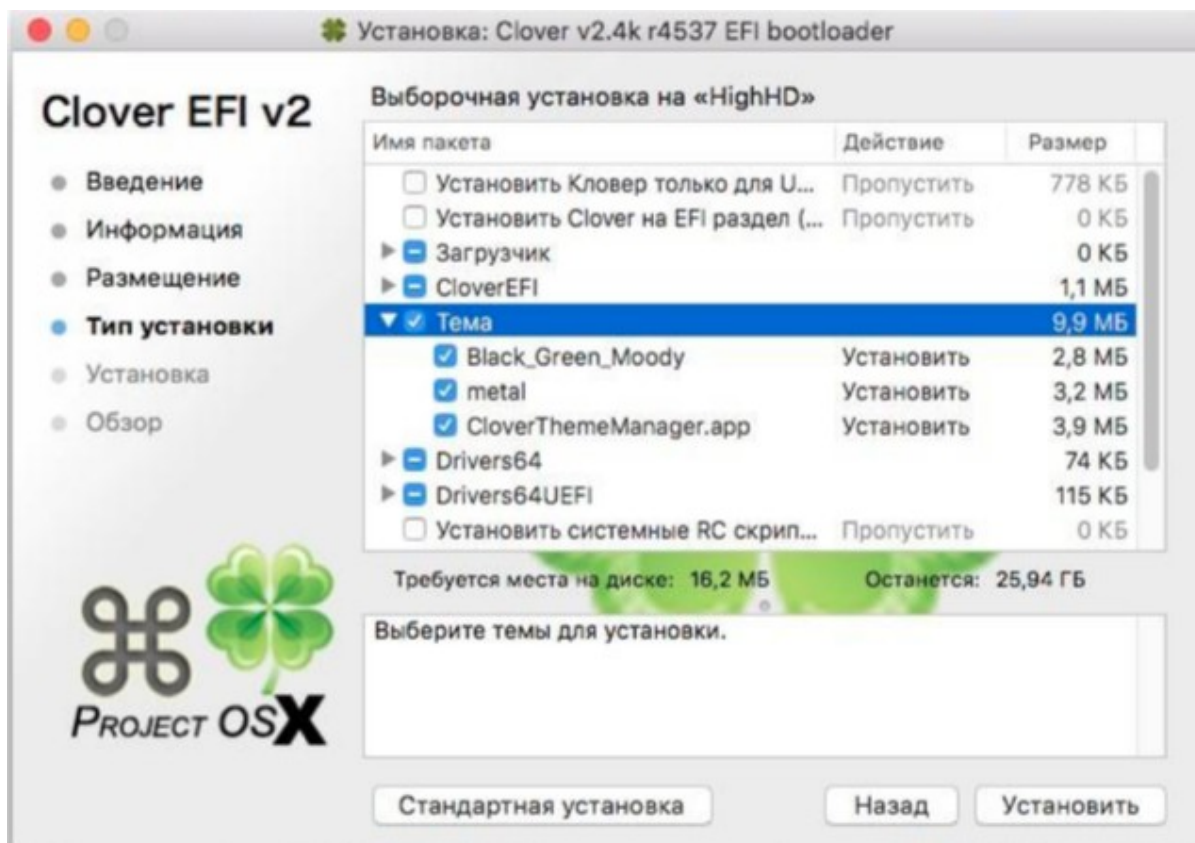


3. Sie haben sich entschieden, dem Mac eine separate Festplatte zu widmen. Hier gibt es eine einfache Möglichkeit per BIOS zu wählen, was geladen werden soll, und die Festplatte wird entsprechend dem ersten Szenario vorbereitet. Entweder Sie legen es auf die primäre die Datei boot0md in den MBR-Sektor. Seine Eigenschaft ist, dass er nach HFS+-Partition über alle Festplatten hinweg, und würde in diesem Fall als zweite Option funktionieren. Lesen Sie nun erneut das Kapitel zur manuellen Installation. Und zu guter Letzt. Ist es nicht an der Zeit, Windows XP ganz abzuschaffen, und Windows UEFI, z. B. Sieben zu nutzen? Die MBRD-Platte kann ohne Datenverlust in HTP mit dem Dienstprogramm gdisk umgewandelt werden. MacOS würde davon profitieren, und Windows sowieso, das alle sechs Monate neu installieren werden muß. Falls es jemand falsch versteht: Windows UEFI wird von Clover perfekt installiert auf Maschinen, die kein UEFI-BIOS haben.

Design

Wahl des Themas

Lassen Sie uns nun ein Thema auswählen. Was ist ein Thema? Dies sind die Gestaltungselemente: das Banner, der Hintergrund Bild, Icon- und Button-Zeichnungen, Schriftart, vereint durch eine einzige künstlerische Idee.



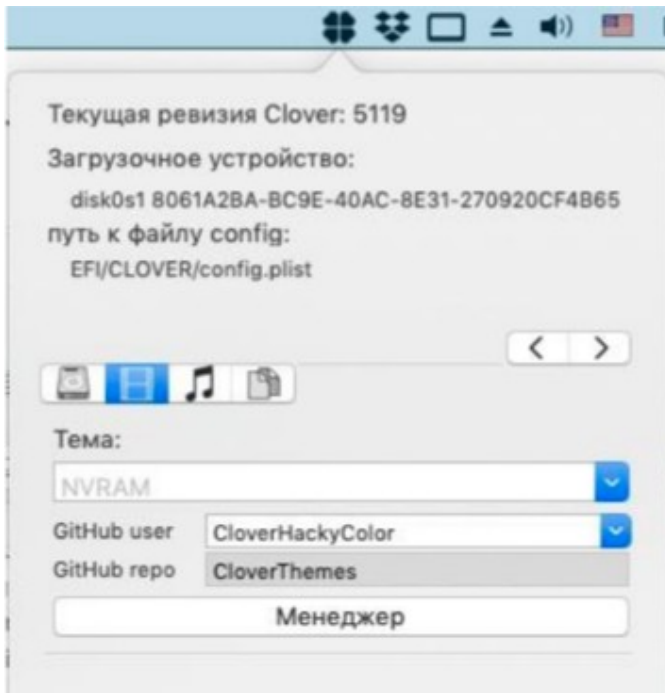
Es gibt drei Themen im Installationsprogramm, mein Cäsium-Thema, ein Thema von Clovy, das sind Vektorthemen, und ein Rasterthema von Blackosx, aus irgendeinem Grund hat er sein SVG-Theme nicht für den Installer angeboten. Außer ihnen gibt es "Ostereier", wer sie gesehen hat, weiß bescheid, für den Rest wird es eine Überraschung sein. Und dann ist da noch Clover. Clovy's eingebautest Thema, genannt embedded. Wenn Sie kein anderes Thema installieren, ist dieses immer noch zur Verfügung stehen. Und es gibt eine App, mit der Sie jedes Thema aus den 40 verfügbar über die Gita. Ich werde Ihnen die Details nicht verraten, alles ist dort transparent.

<http://www.insanelymac.com/forum/topic/302674-clover-theme-manager/>

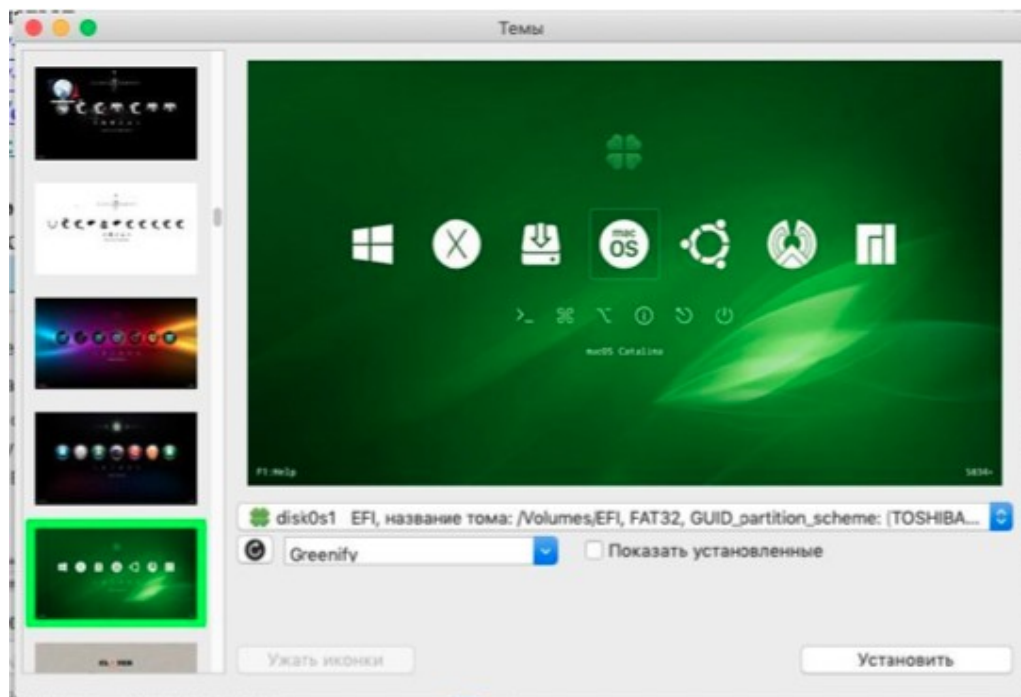
In der modernen Version können Sie ein "zufälliges" Thema angeben und jedes Mal, wenn der nächste Start machen, können sie die unterschiedlichen Designs beobachten.
Lassen Sie uns einen kurzen Überblick über die Themen geben. Der eigentliche Themenkomplex ist viel breiter, schauen Sie sich die Foren an, wer macht, was wer schlägt vor:

<http://www.applelife.ru/threads/themes-temy-dlja-zagruzchika-clover.36074/>
<http://www.insanelymac.com/forum/topic/288685-clover-themes/>
<http://clover-wiki.zetam.org/Theme-database>
<https://sourceforge.net/p/cloverefiboot/themes/>

Jetzt hat vector-sigma seinen Clover Themen-Manager innerhalb der Clover.app
Die Taste sieht aus wie Film.



Auswahl eines Themas



Clover-Bootloader-Themen
Metall. Autor Slice. Thema 1.



Schwarz-Grün. Autor blackosx. Das Theme, das standardmäßig mit diesem Installationsprogramm geliefert wird.



Clover in khakifarben. Version 5.1, Revision 5128
Moskau, 2021

Steampunk. Autor medik.



Und Dutzende von anderen... Das Erstellen eines eigenen Themas ist eine besondere Geschichte mit eigenen Regeln.

Konfigurieren der Schnittstellen in der config.plist

Die Themen umfassen auch eine Reihe von Parametern, die in die Datei config.plist geschrieben werden. Für ältere Versionen, siehe die alten Versionen der Anleitungen.

Die Schnittstelleneinstellungen werden in EFI/CLOVER/config.plist unter GUI vorgenommen
<key>GUI</key>
<dict>

Die Oberfläche kann grafisch oder textuell sein (beginnend mit der Revision 1764). Zu diesem Zweck wird geschrieben

<key>NurText</key>
<true/>

Wahrscheinlich gibt es nur in Russland Liebhaber textueller Oberflächen, Total Commander, Volkov Commander, DOS, usw., usw. Zu Ihren Diensten!

Clover - Main Menu

```
Boot Legacy from Test_Clover
Boot Windows from TEST_UISTA
Boot UEFI internal from EFI
Boot UEFI internal from Test_Clover
Boot Mac OS X from Test_Mavericks
Boot Mac OS X from Test_Lion
Boot Mac OS X from Test_Mountain_Lion
About rEFIt
Options
Clover Boot Options
Start UEFI Shell 64
Restart Computer
Exit Clover
```

Wenn false gesetzt ist, läuft Clover im grafischen Modus.

Die Auflösung des Textbildschirms, hier wie auch in der Shell und dem boot.efi-Bildschirm, kann eingestellt werden

mit Hilfe von

`<key>KonsoleModus</key>`

`<string>0/Min/Max/irgendeine Zahl</string>`

Die Zahl finden Sie in Ihrem boot.log, oder geben Sie Max ein - es wird das Maximum der möglichen Auflösung sein .

Das Aussehen der grafischen Shell hängt vom gewählten Thema ab. Das Standardthema wird in der Variablen

`<key>Thema</key>`.

`<string>Metall</string>`

Das Thema kann jedoch auch in der Systemsteuerung ausgewählt werden, und diese Auswahl wird bestimmen. Wenn dort jedoch das falsche Theme angegeben ist (es gibt keine solche theme.plist-Datei von Das Thema in der Liste wird ausgewählt, wenn es nicht vorhanden ist. Wenn es auch eine nicht vorhandene Thema, dann wird auf dem Bildschirm ein eingebettetes Thema (embedded) angezeigt, die neueste Version von Clovy. Thema kann auch im Bootloader-Menü geändert werden, dort wird eine Liste der installierten Themes (rev. 1955), und Sie können angeben, welche Sie wünschen. Die Oberfläche wird neu gezeichnet, nachdem Sie zum Hauptmenü. (Revision 1936) zurück kehren

random - das Thema wird bei jedem Laden zufällig aus der Liste der installierten Themen ausgewählt.

Eine weitere Sache, die Sie bei der Auswahl eines Themas berücksichtigen sollten. Wenn Ihr Monitor eine Auflösung von z. B. 2560x1600 hat, dann sind die Symbole und Schriften auf dem Bildschirm zu klein dargestellt. Sie sollten ein großes Thema wählen. Beginnend mit Revision 4438 Clover wird dies automatisch tun. Auf einem kleinen Monitor verwendet es das Thema, sagen Sie metal, und auf dem großen Monitor wird versucht, das metal@2x-Thema zu finden, falls es eines gibt.

Das Kriterium für einen "großen" Monitor ist die Anzahl der Zeilen über 1100, denn bei mir auf einem 1080 Linien, ein kleines Thema reicht völlig aus. Die Frage für die Designer ist, wann sie große Analogien zu ihren Themen. Ab 4862 werden skalierbare Themen erstellt. Das ist ein eigenes Kapitel.

Was diesen Konfigurationspunkt betrifft, so sieht er für Raster und Vektor-Themen, der einzige Unterschied ist der Name. Clover selbst unterscheidet das eine vom anderen durch

den Inhalt des Ordners. Wenn eine theme.plist vorhanden ist, handelt es sich um ein Bitmap-Thema. Wenn es eine theme.svg gibt, ist es ein Vektor-Thema.

```
<key>EmbeddedThemeType</key>  
<string>DayTime</string>.
```

Mögliche Varianten Dunkel und Hell. Eingeführt mit Revision 4644. Die Varianten wurden erfunden, bevor es die Erfindung der Uhr gab, Sie konnten ein dunkles oder helles eingebautes Thema wählen. Und jetzt nach Stunden, Licht am Tag, Dunkelheit in der Nacht. Eingeführt mit Revision 4773.

```
<key>Timezone</key>  
<integer>3</integer>
```

Mit der modernen Revision von Clover, 4773+, wurde das Konzept der "Tageszeit" eingeführt. Beim Starten von Clover. Es liest die aktuelle Zeit aus dem BIOS, die normalerweise die Greenwich Mean Time ist, und addiert dazu Wert aus diesem Parameter, nach der Zeit-Zone. Sie können mit einem +3-Zeichen oder, sagen wir, -5 schreiben. Zusammengefasst. So erhalten Sie die Ortszeit. Clover ermittelt dann, ob es zwischen 8:00 und 20:00 Uhr ist, dann zählt es als Tag, ansonsten als Nacht. Dies betrifft das bereits erwähnte eingebaute Thema, wenn DayTime angegeben ist, wirkt sich das sicherlich auf jedes Vektorthema aus. Um zu bestimmen, was es beeinflusst, beeinflusst es den Klang in jedem Thema. Tageszeitliche Wiedergabe des Tons aus der Datei sound.wav, Nacht aus der Datei sound_night.wav. Wenn kein Nachtton definiert ist, spielt es den Tageszeit Ton.

```
<key>PlayAsync</key>  
<false/>.
```

Ab der Revision 4833 erschien der Ton in Clover mit dem Treiber AudioDxe.efi, Autor Goldfish64, der sich nicht mit Clover befassen möchte, nun gut, es macht ihm nichts aus, das wir diese Lösungen benutzen. PlayAsync bestimmt, ob der Ton synchron mit dem Clover-Vorgang abgespielt wird, oder asynchron. Das heißt, bei synchroner Wiedergabe funktioniert nichts, bis der Ton enden wird. Bei asynchron fließt der Ton unabhängig davon, was auf dem Bildschirm passiert. Das heißt, Ton startet nach der Meldung auf dem Bildschirm ...scan entries... , dann erscheint die Oberfläche von Clover, der Ton läuft weiter, dann erscheinen die Zeilen der boot.efi Operation, der Ton läuft weiter, dann wird der Kernel gestartet, und die Kexte werden nacheinander gestartet. Ausgehend von einigen von ihnen, bricht der Klang ab. Dies gilt natürlich, wenn die sound.wav-Datei lang ist. Clover 4862 hatte einen Fehler, asynchroner Sound hat den Computer aufgehängt. Ab 4870 gibt es kein Problem, können Sie PlayAsync=true einstellen.

```
<key>CustomIcons</key>
```

```
<false/>.
```

Wenn auf <true/> gesetzt, wird jede Partition mit dem Betriebssystem suchen Sie im Stammverzeichnis der Partition nach dem Symbol .Volumelcon.icns und verwenden Sie es anstelle von Symbolen, die durch das Thema festgelegt sind. Dieses Symbol ist sehr einfach mit MacOSX zu erstellen. Wählen Sie das Symbol des Laufwerks und machen Sie ein Copy-Paste