

„Partycjologia APA”

Wraz z premierą PS2, Sony jako pierwsze wprowadziło na rynek konsol dysk twardy. Ze względu na już i tak absurdalnie wysoką cenę, zdecydowano się wtedy na HDD jako akcesorium opcjonalne, doczepiane do również opcjonalnej karty sieciowej zwanej Network Adaptor. Dysk pełnił rolę legowiska na systemy operacyjne, aktualizacje firmware i w kilku wyjątkach także na save'y, gry i ich dodatki. Oczywiście to nie jest ten sam poziom integracji jak w Xbox, ponieważ nie był ani obowiązkową częścią konsoli (z wyjątkiem modeli DESR-xxxx), ani nawet popularny wśród graczy, a co za tym idzie również deweloperów gier. Nas, czyli fascynatów tzw. sceny, interesuje oczywiście pod innym kątem, czyli jak można go wykorzystać w sposób nieprzewidziany przez twórców konsoli.

Poniższa ściana tekstu co prawda traktuje o kilku różnych zagadnieniach, ale na tyle się zazębiających, iż uznałem że nie ma sensu wydzielać każdego z nich do osobnego poradnika. Dlatego wszystko razem tworzy spójną całość, ale ani nie musisz instalować wszystkiego, ani nawet robić tego samego co ja, ponieważ po sieci krąży wiele obrazów HDD z już wgranymi i skonfigurowanymi cudami wg. widzimisię autorów. Informacje które tutaj zawarłem są dla osób ceniących sobie porządek – i jak zwykle – pasjonatów tzw. sceny z nadmiarem wolnego czasu. ;)



Sposoby podłączenia HDD

Oficjalnie dysk twardy można podłączyć wyłącznie do „grubych” modeli PlayStation 2, wyłącznie za pomocą urządzenia zwanego **Network Adaptor**, który to zależnie od modelu PS2 przyjmuje postać **karty sieciowej** (z wbudowanym bądź nie, modemem).

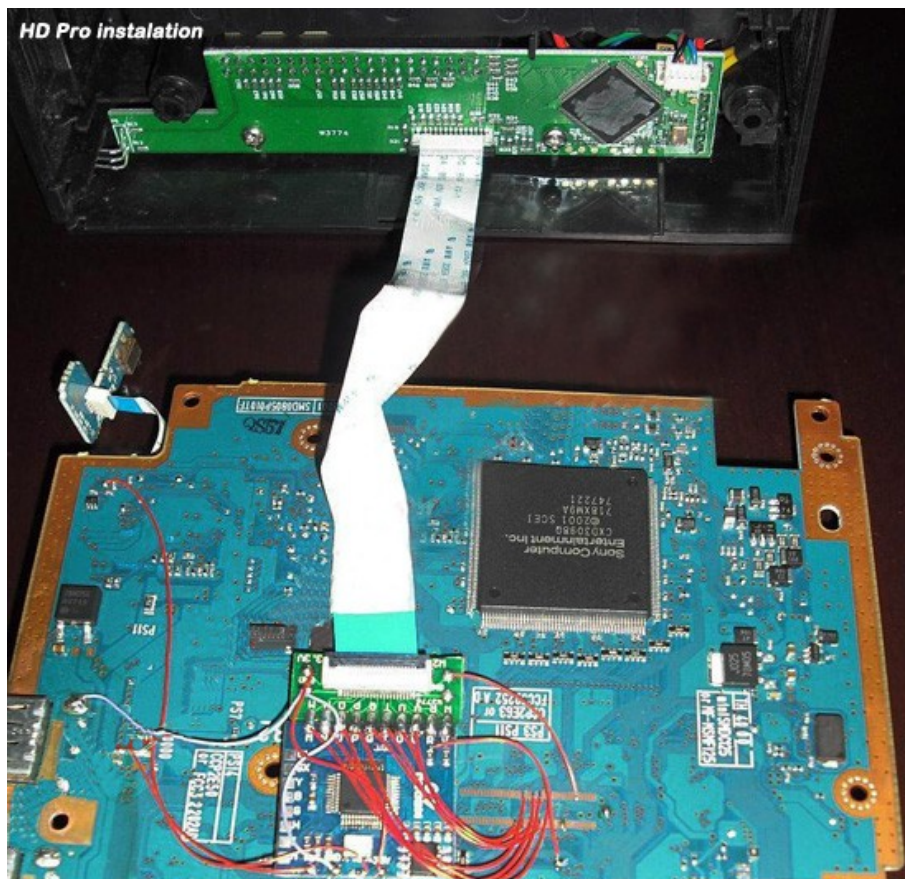
- W przypadku konsol z serii **SCPH-1xxxx** (np. SCPH-18000), karty wkładanej do konsoli, podłączanej via **PCMCIA** – i w tym przypadku dysk twardy jest zewnętrzny, zamknięty w specjalnej obudowie z doczepianym kablem do NA (oczywiście nie należy mylić z dyskiem zewnętrznym na USB).
- W przypadku konsol z serii od **SCPH-3xxxx** do **5xxxx** włącznie (np. SCPH-30004), karty nieco siermiężnej, doczepianej do konsoli przez specjalny port, zaś dysk twardy podłącza się do NA przez standardową parę **PATA** i **Molex** (niestety nie każdy HDD ma ten sam rozstaw portów). Całość wsuwa się w czeluść zatoki **Expansion Bay**, gdzie dysk chowa się w konsoli, a NA wystaje jak garb na plecach.



Nieoficjalnie dysk twardy można jeszcze podłączyć do **SCPH-70xxx** lutując srogą pajęczynę do płyty głównej, ponieważ w tych modelach kontroler IDE nadal jest obecny na MOBO i nadal wspierany w firmwarze PS2 (zlikwidowano jedynie port).

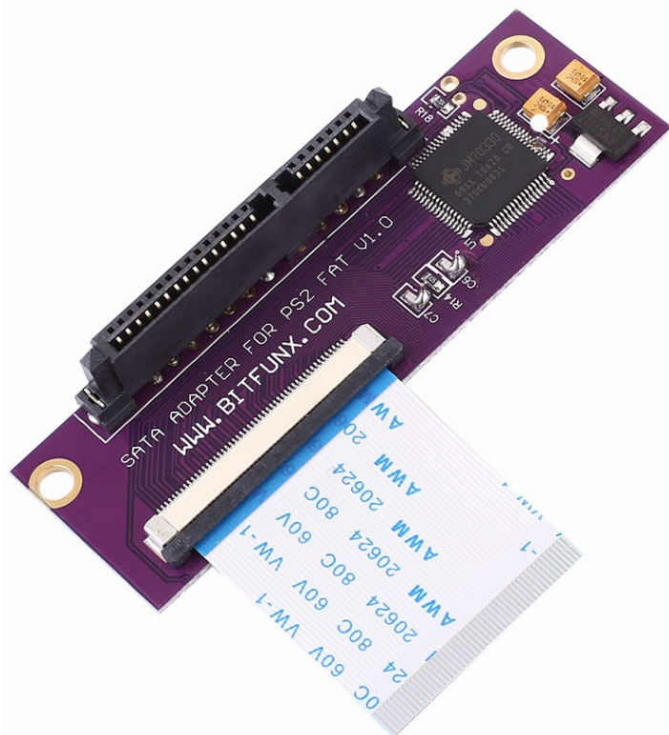
W nowszych wersjach (czyli od **SCPH-75xxx**) już pozostają rozwiązania via modchip, który zajmuje się komunikacją z dodatkowym kontrolerem IDE i patchowaniem fw.

Ewentualnie Dysk twardy można podłączyć za pomocą portu USB lub i.Link, ale to już materiał na zupełnie inny poradnik, ponieważ jako wewnętrzny, traktowany może być wyłącznie ten podłączony do Network Adaptora lub MOBO. Dla zewnętrznego, logika pamięci masowej opiera się o FAT32 lub EXT2 na MBR, z kolei wewnętrznego głównie o PFS na APA (o czym możesz przeczytać na kolejnych stronach).



HD Pro dla modeli SCPH-75xxx.

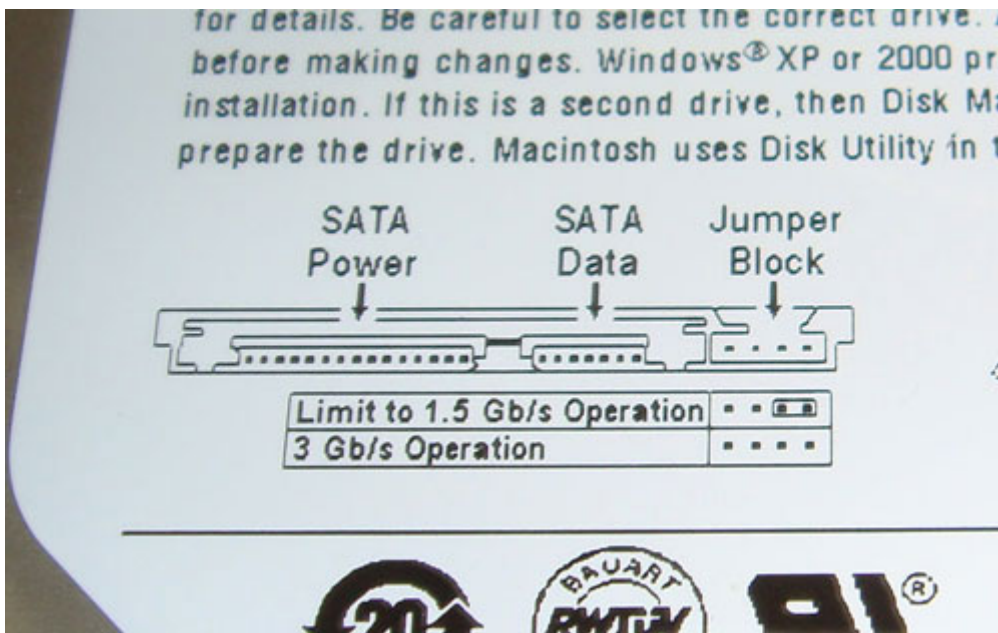
Dysk twardy musi być na **PATA**, ze zworką ustawioną na **MA** (Master) plus ewentualnie dodatkową ograniczającą pojemność. Choć są różne kontrolery, które umożliwiają podłączenie dysków na SATA (tak jak np. nieoficjalne Network Adaptors, które zamiast IDE i Molex'a mają SATA'owskie złącza), to różnie z nimi bywa bo nawet z „taśmowymi” kompatybilność nie jest stuprocentowa... Swego czasu wisiły na różnych stronach i forach dyskusyjnych przeróżne listy z modelami HDD, które sprawiają bądź nie, problemy, ale ani wszystkie raporty nie były wiarygodne, ani tym bardziej z oczywistych względów, listy kompletne. Dysk twardy, którym przyszło mi się bawić jest ST340211A, podłączony przez oryginalny NA, na modelu PS2 SCPH-30004R.



Chiński zamiennik wnętrza Network Adaptora, zastępujący PATA, nowszym SATA.

Jeśli zdecydujesz się na dysk twardy SATA to niestety wiązą się z nimi problemy związane z prędkością transferu danych. W przypadku ATA-300 (i prawdopodobnie także ATA-600), gry wgrane jako partycje (czyli uruchamiane za pomocą Open PS2 Loader lub HDLoader) mogą się zawieszać bez ustawienia trybu MDMA2, z kolei gry z PSX odpalane via emulator POPS, w ogóle się nie uruchomią.

Jeśli twój HDD na SATA-II lub III posiada możliwość jego zwolnienia do ATA-150 (może to być również nazwane trybem 1.5Gb/s) za pomocą wsuwanej zworki (rozwiązanie podobne jak w dyskach PATA dla konfiguracji Master, Slave, Cable Select i opcjonalnego ograniczania LBA) to warto zastosować i wtedy POPS nie będzie wybrzydzać, a tam gdzie można, należy wówczas stosować tryb UDMA zamiast MDMA.



Nie wszystkie HDD posiadają opis ustawiania zworek. Informacji na ten temat szukaj na stronie producenta dysku, rzecz jasna w instrukcji obsługi dla konkretnego modelu.

Brak HDD

Konsola może nie wykryć dysku twardego jeśli zworki będą błędnie ustawione. Jak napisałem wyżej, musi być przynajmniej jedna, która ustawia dysk jako Master (MA). Bywa i tak, że brak zworek to MA. Wszystko zależy od konkretnego modelu HDD. Informacje o konfiguracji dysków PATA zawsze znajdują się na etykiecie przyklejonej do obudowy.

Drugą możliwą przyczyną jest niedokładne włożenie dysku do Network Adaptora. Szczególnie jeśli rozstaw Molex'a i IDE nie pasują idealnie do rozstawu wtyczek po stronie HDD, a użytkownik musiał się siłować by go tam upchnąć. ;) Bywa, że w ogóle nie pasują i użytkownik ratował się dzierganiem mostków do wyrwanych gniazd, które nie wytrzymują zbyt wielu cykli wyjmij/włóż.

Trzecia przyczyna, która przychodzi mi do głowy to wyrobiony slot na Network Adaptor w konsoli. Jest dość płytki i delikatny, więc im rzadziej będziesz wyjmować z PS2 tym lepiej.

Czwarty powód to niedostateczne dociśnięcie Network Adaptora do konsoli. Teoretycznie NA nie musi być nawet przykręcony śrubkami, ale lepiej z tego nie rezygnować bo nawet najmniejszy luz może skutkować niewykrywaniem HDD.

Ostatnią przyczyną może być po prostu uszkodzony Network Adaptor. Nigdy nie wyjmuj go na włączonej konsoli (nawet na włączonej w trybie czuwania).



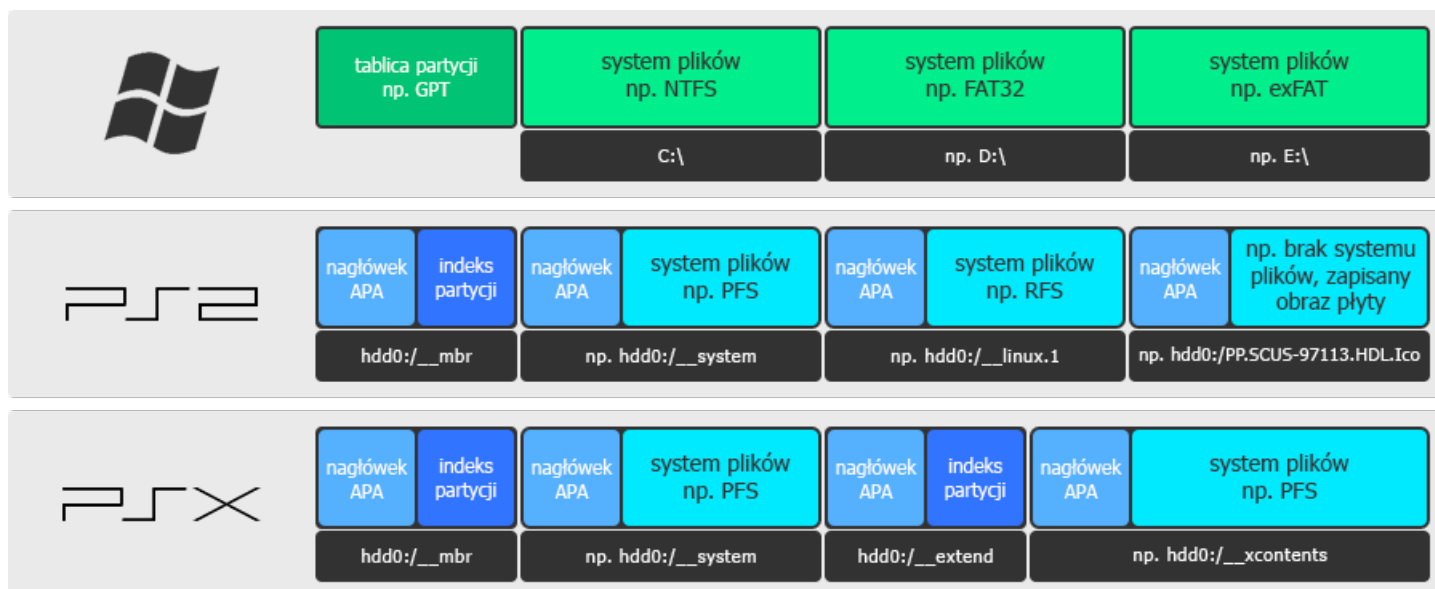
Struktura logiczna

Jak to Sony ma w zwyczaju, i tym razem nie użyło jednego ze standardowych otwartych formatów, a zaprojektowało własne, ale przynajmniej trenujące wyobraźnię. ;)

Tablica partycji?

PS2 nie używa żadnej tablicy partycji. Zamiast tego, stosuje indeks z utworzonymi partycjami, ale obudowany tak jak wszystkie pozostałe, nagłówkiem **APA** (Aligned Partition Allocation). Format ten pozwala zapisywać partycje w dość ekstrawagancki sposób bo blokami gdzie najmniejszą możliwą jednostką jest **128MiB** (a więc rozmiar partycji musi dokładnie tyle wynosić lub jej **wielokrotność**), maksymalną zaś **128GiB**. W nagłówku każdej partycji, w specjalnym obszarze zwanym **Attribute** znajdują się informacje o nazwie, typie partycji (czy jest podstawową, czy sub-partycją), można tam również znaleźć ikony do wyświetlania w menu konsoli (PS2 Browser), a nawet plik wykonywalny.

Aby lepiej zrozumieć na czym polega różnica pomiędzy standardowym modelem partycjonowania, a dziwością opracowaną w Sony, poniżej przedstawiam małe porównanie. W przypadku np. Windows (na zielono), „partycjologia” wygląda tak, że jest jedna tablica partycji, w której znajdują się informacje o zajmowanej przestrzeni przez co najmniej jedną partycję. W przypadku PS2 i PSX (na niebiesko), masz do czynienia z sekwencją partycji gdzie indeks (lub indeksy) utworzonych partycji i każdy system plików obudowany jest nagłówkiem APA.



Oryginalne oprogramowanie Sony projektowane było zawsze pod oficjalny dysk twardy i teoretycznie żadnego innego nie obsługuje. Wiąże się to nie tylko z „niewidzialnymi” HDD innymi niż oryginalny, ale i z ograniczeniem pojemności do **LBA28** – czyli mówiąc po ludzku – wielkości 128GiB (~137GB). Obecnie całe scenowe oprogramowanie obsługuje **LBA48** 2TiB (~2,2TB), włączając w to hacki firmware Sony, ale dawniej oczywiście nie i przez krótki okres, jedynym sposobem obejścia tego ograniczenia było zastosowanie **APAEXT** (Toxic Extended APA). TeamToxic, twórcy znanej serii modchipów, zaimplementowali w swoim loaderze (ToxicOS) odczytywanie sklonowanej pierwszej partycji tuż po tym gdzie się kończy zakres dla standardowego indeksu. Innymi słowy, w przypadku APAEXT są dwa indeksy, jeden dla pierwszych 128GiB, a drugi dla przestrzeni poza nią. Zaletą rozszerzonej APA jest zachowanie kompatybilności ze wszystkim co powstało dla LBA28, ale wadą niekompatybilność z oprogramowaniem obsługującym LBA48 (nie możesz wtedy formatować HDD, ponieważ stracisz partycje wpisane do drugiego indeksu, ani też tworzyć partycje przekraczających barierę 128GiB).

Zabawne, że kilka lat później, Sony postąpiło dokładnie tak samo implementując obsługę dysków twardych w modelach **DESR-xxxx** (czyli specjalnych **multimedialnych PS2**, przewrotnie nazwanymi **PSX**, a potocznie **PS2 DVR**). :) Podobnie jak w APAEXT, tak i tutaj struktura logiczna dysku podzielona została na dwa obszary, ale tym razem jeden do 40GiB, który pokrywa pierwszy indeks, i kolejny powyżej 40GiB, aż do możliwych 250GiB, który pokrywa drugi. Ten wariant umownie nazwę **APADVR** by odróżnić od wersji standardowej i „toksycznej” (oficjalnie nie ma nazwy i dla Sony to po prostu APA).



Systemy plików

Głównym systemem plików wykorzystywanym przez większość oprogramowania PS2 jest **PFS** (PlayStation File System?). Dziś ten skrót kojarzy się ze specjalnym szyfrowanym kontenerem używanym na PlayStation Vita i PlayStation 4, ale w czasach PS2 była to oficjalna nazwa ichniego systemu plików na dysku twardym, która kompletnie nie ma nic wspólnego z nowszymi konsolami. Podobno swoją budową przypomina EXT2.

Partycje systemowe to wszystkie których nazwy zaczynają się podwójnym podkreślnikiem, a więc zgodnie ze schematem: "__<etykieta>" lub "_.<etykieta>".

Pierwszą i jedyną obowiązkową jest "__mbr", która to wyjątkowo **nie posiada systemu plików**. To tam wgrany jest podpisany loader (kluczami które od czasu złamania PlayStation 3 znamy), a który to z kolei może uruchamiać również podpisany kolejny loader (np. boot manager Linuksa z **Linux Kit** lub **PSBBN** (specjalna dystrybucja, ale inna niż ta oficjalnie sprzedawana)). Jej najważniejszą funkcją jest oczywiście indeks utworzonych partycji (ekwiwalent tablicy partycji). W przypadku zastosowania APAEXT lub na modelach DVR, dodatkowo również pierwszą i obowiązkową partycją jest "__mbr_ext__" (znajdącą się tuż po pierwszych 128GiB) lub "__extend" (tu po pierwszych 40GiB).

Pozostałe partycje są obowiązkowym zestawem m.in. dla **HDD OSD** (w pewnym sensie jest to aktualizacja firmware) i **PSBBN**:

<u>__net</u>	zawiera informacje związane z DRM np. gier ściągniętych z kanałów PSBB
<u>__system</u>	zawiera zasoby firmware i loadery
<u>__sysconf</u>	zawiera zasoby firmware i konfiguracje
<u>__common</u>	zawiera dane użytkownika jak np. save'y
<u>__contents</u>	zawiera dane użytkownika, używana wyłącznie przez PSBBN

W przypadku DVR, znaleźć można jeszcze dwie dodatkowe partycje. Co prawda prawie cały dysk jest zaszyfrowany, ale odbywa się to na poziomie sprzętu, a więc owe szyfrowanie jest zupełnie transparentne dla uruchomionego oprogramowania (dzięki czemu możliwe jest z jego poziomu wykonanie posektorowego, odszyfrowanego obrazu).

<u>__xdata</u>	zawiera tymczasowe dane, np. pobrane aktualizacje firmware
<u>__xcontents</u>	zawiera dane użytkownika takie jak np. muzyka, zdjęcia, nagrania wideo, miniaturki XMB

Dodatkowe partycje z podkreślnikami są tworzone przez programy homebrew. Na przykład starsze wersje uLE po sformatowaniu dysku tworzyły jeszcze partycję "__boot", której używały modchipy do tzw. DEV2 (Devolution Mode 2, czyli uruchamiania stamtąd pliku "BOOT.ELF"). Spotkałem się jeszcze z "__main", ale nie wiem do czego służy bo do niczego co pokrywa poradnik, który właśnie czytasz.

Skasowane partycje zostają oflagowane jako puste i zmienia się ich etykieta na "__empty". Homebrew na PS2 je ignorują, ale różne scenowe programy na PC umożliwiają odzyskiwanie z nich danych lub przywracanie. Możliwa jest jeszcze partycja "__tmp", ale służy wyłącznie grom na tymczasowe operacje i jest usuwana przy starcie konsoli.

Dotychczas wszystkie **partycje homebrew**, czyli te utworzone przez nielicencjonowane programy, miały w etykiecie znak plusa poprzedzający wybraną nazwę. Odkąd jednak uLE doczekał się „forków” takich jak **double unofficial LaunchELF** (w skrócie wLE – obie te nazwy są nieoficjalne) i **unofficial LaunchELF kHn**, to można już tworzyć partycje z dowolną etykietą. Wszystkie pozostałe programy tworzą partycje wg. schematu: "+<etykieta>" (np. "+OPL", "+ScummVM" itp.).

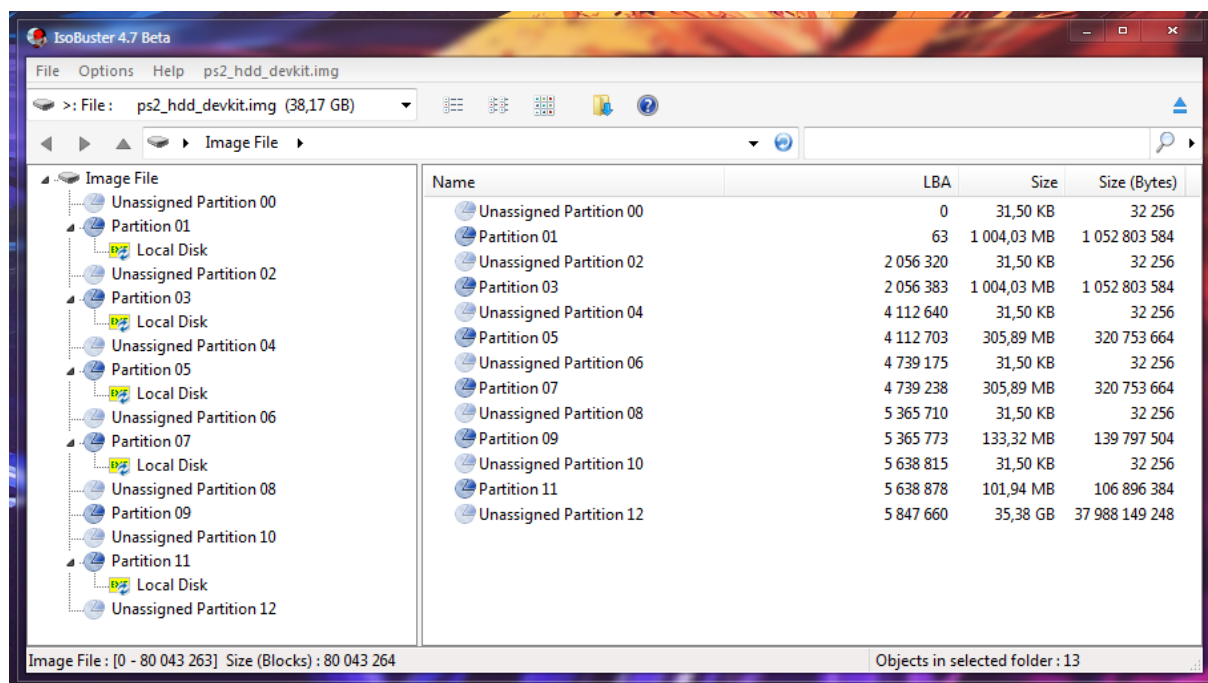
Partycje growe, czyli utworzone przez oficjalne gry (na swoje dane instalacyjne i dodatki) lub PSBBN mają swoje specyficzne przedrostki **PP.** i **PC.** (zależnie od tego czy służą jako **Partition Parent** czy jako **Partition Child**). A więc będą to zawsze: "**PP.<etykieta>**" i "**PC.<etykieta>**". HDD OSD wszystkie „dzieci” traktuje jako całość i wyświetla wyłącznie „rodzica”. Jak się zapewne domyślasz, ten system nazw służył deweloperom do wgrywania gier i „podpinania” pod nie DLC. Poza "**__common**", to jedyne partycje jakie wyświetla „Browser”.

Wszystkie wyżej wymienione partycje (z wyjątkiem "**__mbr**" i "**__extend**"), używają systemu plików PFS, a więc można je przeglądać i swobodnie zarządzać danymi za pomocą managerów plików po stronie konsoli, jak i za pomocą **PFS Shell** po stronie komputera.

Jest jeszcze jeden, specjalny rodzaj partycji growych, czyli nieoficjalne, utworzone z myślą o **HDLoader** i **Open PS2 Loader**. **Nie mają żadnego systemu plików** i zawierają pocięte na kawałki obrazy płyt wgrane jako cała partycja (rzecz jasna zerami zaokrąglane do wymagań jakie stawia APA). Gry wydane na CD muszą być skonwertowane z mode2 (bloki po 2352 bajty) do mode1 (2048), czyli być w takiej samej formie jak te na DVD. Modyfikowane są też informacje o sesji/ścieżkach, więc nie są to idealne kopie 1:1. Tak wgrane gry można wydłubać za pomocą np. **IsoBuster** (począwszy od wersji 4.7) lub **WinHIIP**, a wgrywać oprogramowaniem, które znajdziesz w rozdziale pt. „Kopiowanie gier dla PS2 na dysk twardy”.

Partycje PSBBN wyjątkowo leżą na **RFS**, i choć jest to **standardowy** system plików to w jego pierwszej, jeszcze wtedy komercyjnej wersji.

Partycje Linuksa (oficjalnego z wyjątkiem PSBBN) z kolei leżą na **EXT2**, który także jest **standardowym** systemem plików i ma pełną obsługę w dowolnym Linuksie na dowolnej platformie. Oczywiście montowanie na PC jest możliwe, ale po wskazaniu adresu skąd się zaczyna lub po ekstrakowaniu i podpięciu jako loop device. Wyjątkiem są tutaj **PS2Tool** (czyli **devkity** DTL-Txxxxx), które nie używają APA, a (niemal ;) standardowy MBR.



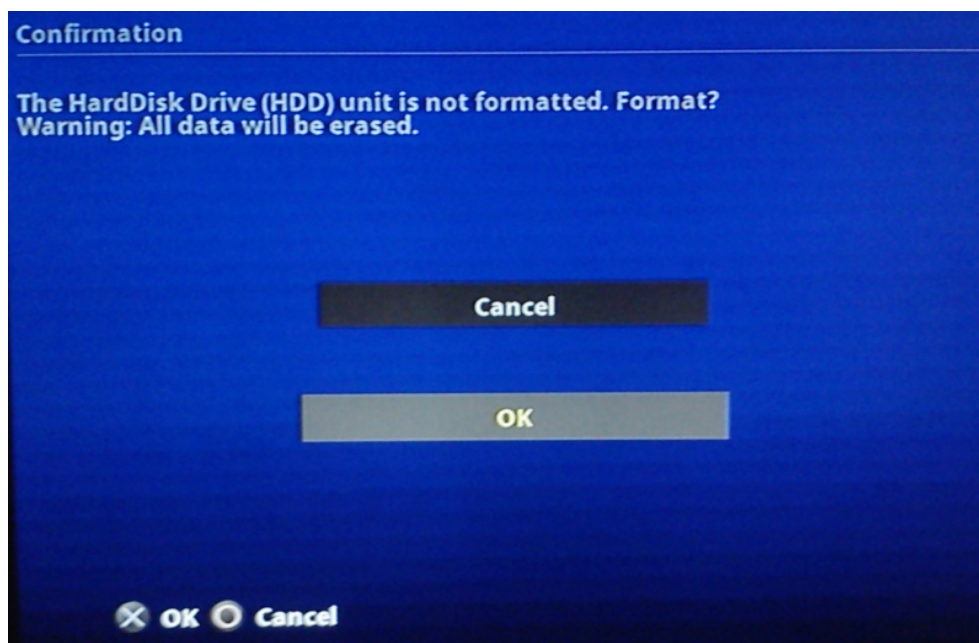
Mozaika partycji pochodząca z dysku konsoli DTL-T15000. Z jakiegoś tajemniczego powodu tablicą partycji jest MBR, ale pozwalający na więcej niż 4 partycje podstawowe (nie, to nie jest GPT), a wszystko to przeplecione 31,5KiB blobami.

Formatowanie HDD

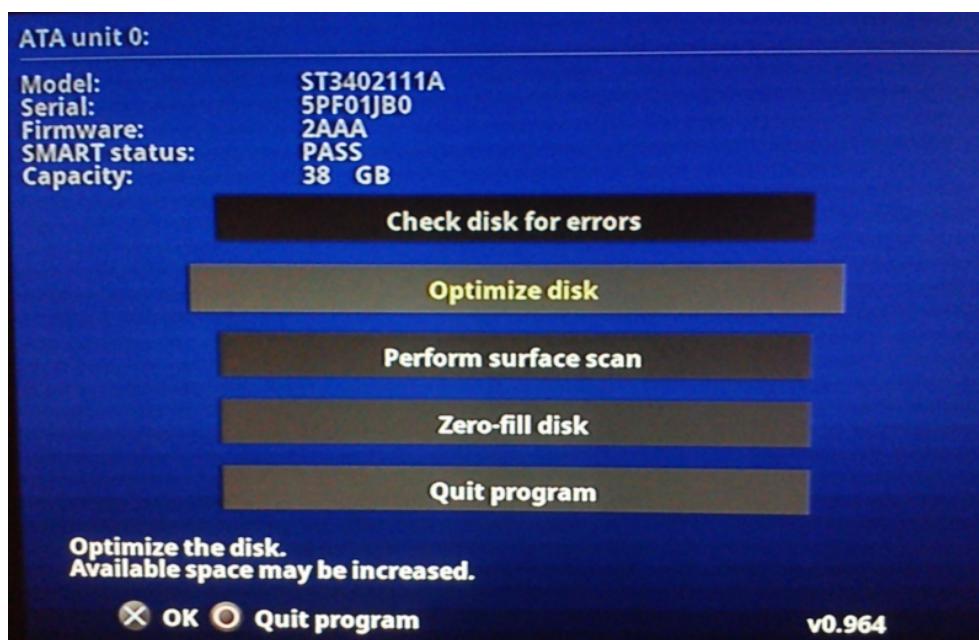
Sformatować dysk twardy można na wiele sposobów jak choćby za pomocą uLE (w menu "PS2 HDD Manager"), a nawet na PC (np. za pomocą **PFS Shell** i komendy "**initialize yes**"), ale najprościej i najszybciej jest moim zdaniem użyć **HDDChecker**, który ma wiele więcej unikalnych, a zarazem przydatnych funkcji.

Uwaga: jeśli posiadasz którykolwiek model DESR, **nie wolno ci tego robić** bo nieodwracalnie zastąpisz APADVR zwykłym APA, a co za tym idzie stracisz system operacyjny bez którego konsola się nie uruchomi. Można to jeszcze naprawić jeśli posiadasz kopię tych danych, ale nie odtworzysz APADVR przez co wszystkie funkcje związane z DVR nie będą już działać.

Po uruchomieniu programu, w przypadku gdy dysk nie posiada APA i podstawowych partycji, od razu zostaniesz zapytany czy chcesz go sformatować.



Ale jeśli wcześniej już używałeś tego dysku w konsoli i chciałbyś go wyczyścić to najlepiej wybrać opcję "Zero-fill disk", dzięki czemu wszystkie sektory zostaną nadpisane. Jeśli tylko chcesz sformatować, to po wybraniu tej opcji odczekaj kilka sekund i anuluj (to wystarczy aby zamalować kawałek "hdd0:/__mbr" by dysk przestał być rozpoznawany) po czym od razu zostaniesz zapytany o formatowanie.



Free HDBoot

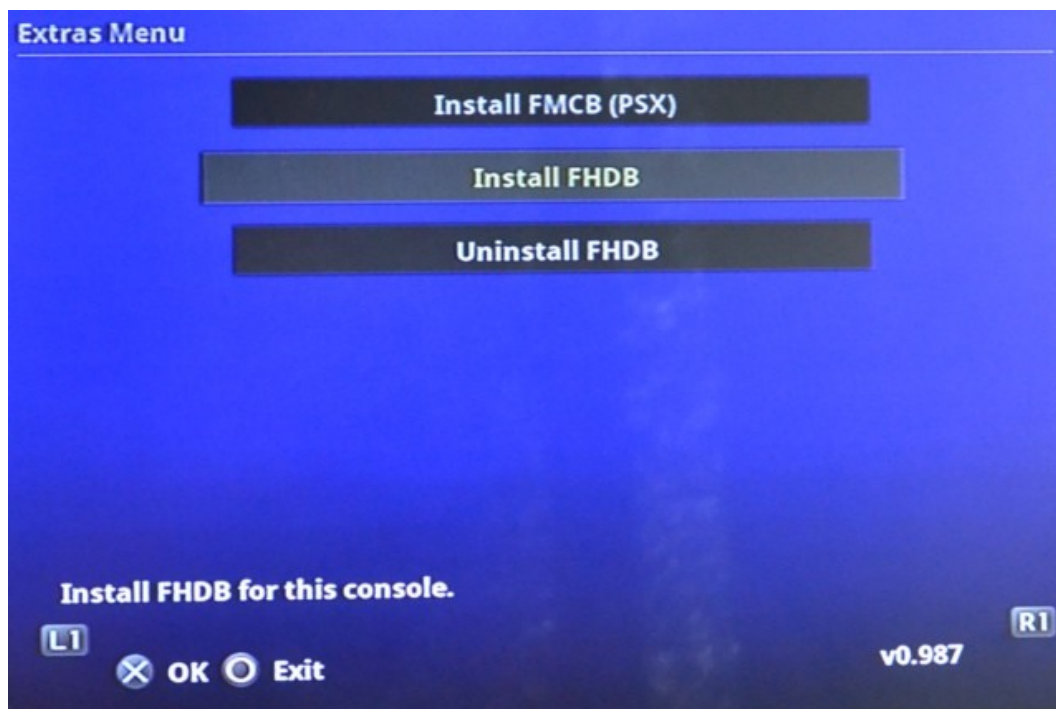
Firmware PS2 zapisany jest na pamięciach ROM, a więc niemożliwych do przeprogramowania. Skoro nie można aktualizować firmware to trzeba było pozostawić sobie furtkę w postaci wczytywania programów i ich zasobów z zewnątrz. I tak też zdecydowali inżynierzy projektujący konsolę, którzy zostawili możliwość wczytania fantów z karty pamięci (co wiele lat później eksploatowały VAST, Memory Plus, Memor32 aż w końcu Free McBoot) lub dysku twardego (co wykorzystuje Free HDBoot). Hack ten więc jest podstawą kolejnych opcjonalnych (nawet jeśli nie interesuje cię FHDB), ponieważ automatyzuje proces obowiązkowego spreparowania "__mbr", tak aby szukał kolejnego loadera już na partycji systemowej.

Uwaga: jeśli posiadasz wlutowany w konsoli **modchip** to prawdopodobnie FMCB/FHDB po zainstalowaniu na karcie/dysku będzie albo się zawieszał przy starcie albo nie pozwoli uruchomić żadnego programu, wyświetlając czarny ekran. Część modchipów posiada funkcję ich tymczasowego wyłączenia^{[1][2]}, a wielu zwyczajnie się wyłączyć nie da. Podobno niektóre starsze wersje są bardziej kompatybilne z chipami (np. **1.965** i **1.963**), ale akurat w moim przypadku również i na nich ww. problemy występują (ostatnią kompatybilną wersją była **1.8b/1.8c**, ale te nie mają wsparcia dla HDD). Najlepszym rozwiązaniem byłoby po prostu odłączenie zasilania modchipowi lub jego całkowite odlutowanie.

1. Uruchom instalator **Free McBoot & Free HDBoot** koniecznie z pendrive, ponieważ z płyty, karty pamięci lub dysku twardego nie zadziała (a przynajmniej wersja **v1.966 (2019-04-13)**).

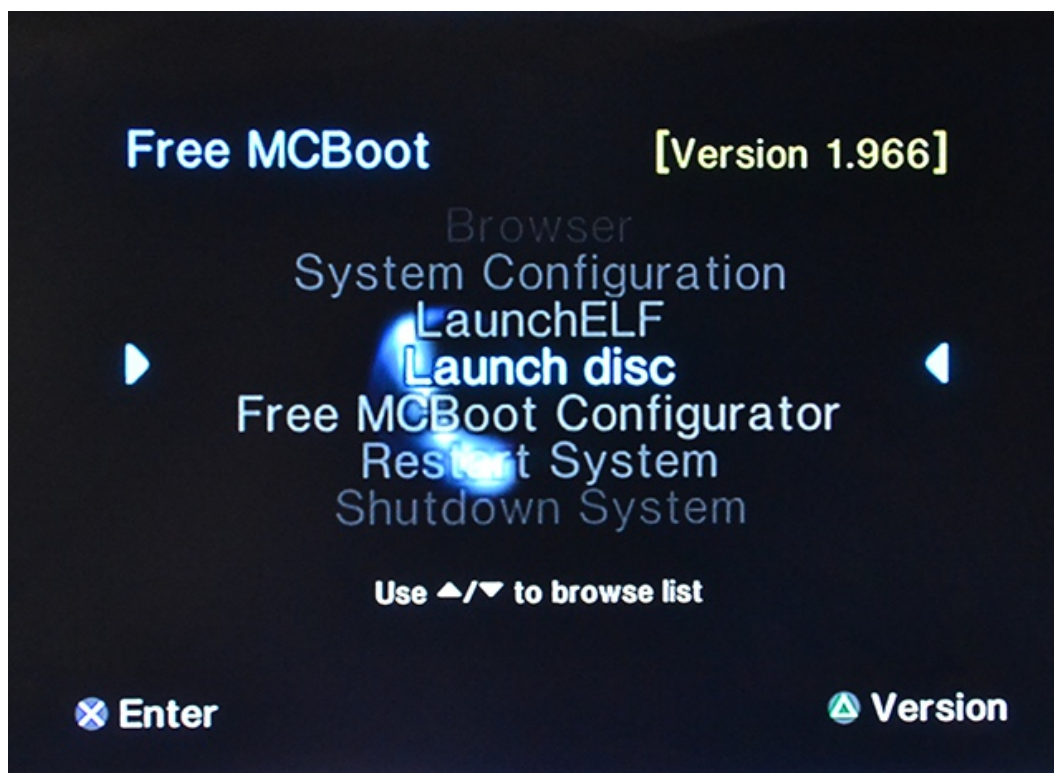
2. Wciśnij przycisk R1, aby przejść do menu instalacji na dysku twardego. Wybierz opcję "Install FHDB" i oczywiście potwierdź "Yes".

Jeśli do folderu "INSTALL\APPS-HDD" wrzucisz jakieś programy, to FHDB podczas instalacji utworzy dodatkową, 128MiB partycję "hdd0:PP.FHDB.APPS" do której skopiuje owe aplikacje.



3. Kiedy pasek postępu dojedzie do końca, dostaniesz komunikat o zakończeniu instalacji. Wróć do menu głównego i wybierz opcję "Exit". Konsola się wyłączy.

4. Po włączeniu PS2 powinno przywitać cię nieco zmodyfikowane menu.



Na modelach SCPH-10000 i 15000 wystąpi problem z wyświetlaniem specjalnych symboli z SJIS (lewa strzałka będzie w innej orientacji, a prawa-dolna z tłem), podobnie jeśli wczytasz plik konfiguracyjny FMCB na FHDB (i vice versa). Rozwiązaniem jest albo dogranie HDD OSD, albo po prostu usunięcie tych wszystkich (moim zdaniem zupełnie zbędnych bo brzydkich) wodotrysków, czyli upodobnić menu do oryginalnego lub zastępując w pliku "[hdd0:/__sysconf/FMCB/FREEHDB.CNF](#)" odpowiednie znaki (ale zrób to albo za pomocą FMCB Configurator albo edytując plik na PC edytorem tekstowym; **nie kopiuj tego z poradnika który właśnie czytasz** ponieważ kodowanie jest tutaj zupełnie inne).

Z:

```
OSDSYS_left_cursor = •o006
OSDSYS_right_cursor = •o005
OSDSYS_menu_top_delimiter = •y-99Free HDBoot          •c1[•r0.80Version %VER%•r0.00]•y-00
OSDSYS_menu_bottom_delimiter = •c0•r0.60•y+99Use •o009/•o010 to browse list•y-00•r0.00
```

Na:

```
OSDSYS_left_cursor = •o009
OSDSYS_right_cursor = •o008
OSDSYS_menu_top_delimiter = •y-99Free HDBoot          •c1[•r0.80Version %VER%•r0.00]•y-00
OSDSYS_menu_bottom_delimiter = •c0•r0.60•y+99Use •o006/•o007 to browse list•y-00•r0.00
```

Wszystkie pliki FHDB znajdziesz w:

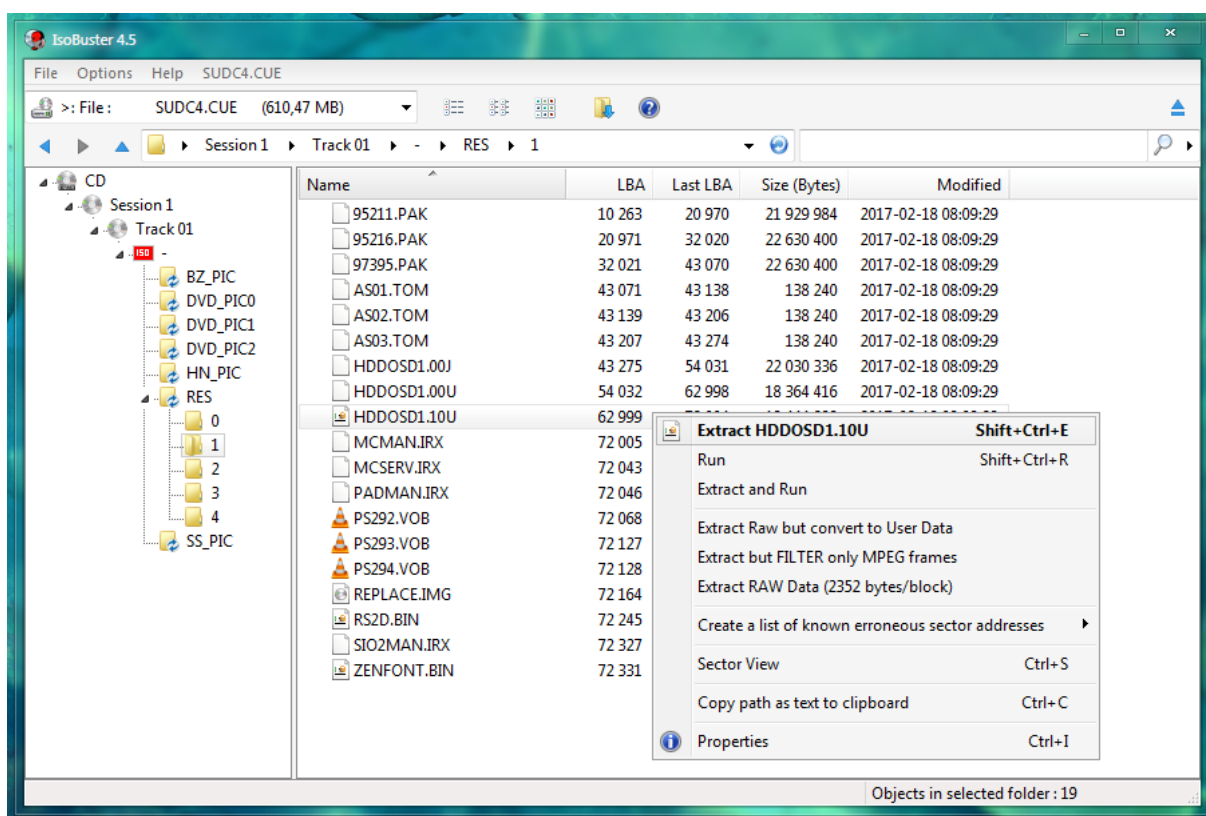
"hdd0:/__sysconf/FMCB/"	program i plik konfiguracyjny plus dodatkowe moduły
"hdd0:/__system/fsck/"	program do sprawdzania systemu plików
"hdd0:/__system/osd/"	Free HDBoot, czyli de facto plik "osdmain.elf"
"hdd0:/__system/osd100/"	lub "osd100.elf"
"hdd0:/__system/osd110/"	lub "osd110.elf"

HDD OSD

PS2 Browser v2.00, potocznie HDD OSD (lub HOSD od nazwy pliku "hosdsys.elf") jest aktualizacją firmware konsoli dla posiadaczy dysku twardego. Umożliwia np. zarządzanie „growymi partycjami” czy też kopiowanie save’ów z karty na dysk i vice versa. Niestety rozpoznawane partycje będą wyłącznie takie, które **nie są** systemowe, a na dodatek mają wypełnione pole z atrybutami (ikona itd, czego np. uLE nie robi, więc takie partycje będą wyświetlane jako **Corrupted Data**).

HDD OSD dołączany był do płyt **HDD Utility Disc**. Problem jednak polega na tym, że płyty te formatowały cały nośnik, zaś sama aktualizacja działała wyłącznie na oryginalnych HDD i na dodatek były bez obsługi LBA48 (czyli pojemności powyżej 128GiB). Dlatego poniższy opis z bólem serca opieram na pirackiej wersji „wszystko-mającej”, ale przede wszystkim spatchowanej, czyli bez wspomnianych wyżej idiotycznych ograniczeń. Niestety kompilację tych narzędzi pod nazwą **SUDC4** (Sony Utility Discs Compilation) musisz odnaleźć sam w mrocznych zakamarkach internetu. ;) Suma kontrolna **MD5** dla obrazu płyty to: **A7901D0EEFBA48960825A55F157F97EC**.

1. Otwórz obraz np. programem **IsoBuster** i wydłub plik "**HDDOSD1.10U**".



2. Skopiowana z obrazu płyty paczka jest w formacie PAK, używanym przez narzędzia Sony na potrzeby PS2. Musisz więc jej zawartość wypakować aby móc ręcznie przenieść aktualizację na HDD.

Do tego służy program **PAKer Utility**, który nie posiada interfejsu graficznego, dlatego albo uprzednio uruchom CMD lub PowerShell albo napisz stosowny *.bat.

Składnia jest bardzo prosta: "**<nazwa programu> /x <plik>**".

```
Procesor poleceń systemu Windows
Microsoft Windows [Wersja 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Wszelkie prawa zastrzeżone.

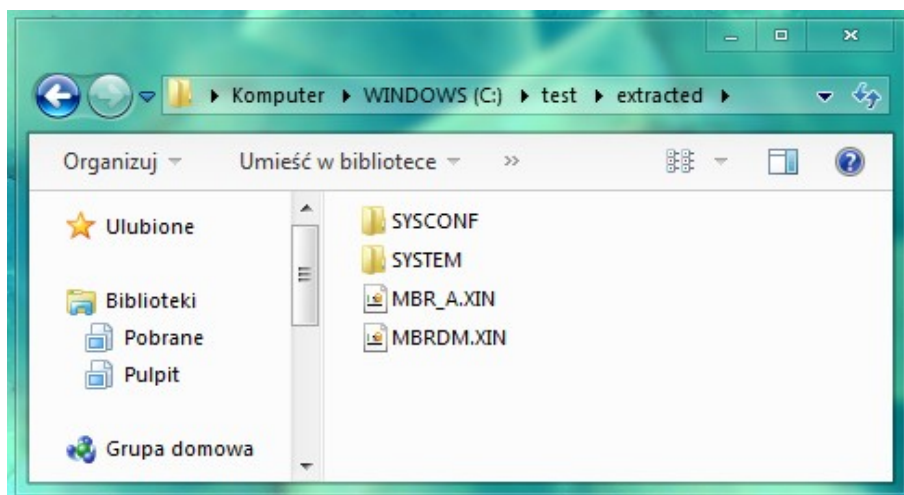
C:\Windows\System32>cd C:\test

C:\test>"PAker Utility v1.01.exe" /x HDDOSD1.10U
PAker Utility v1.01
=====
IV=0x726b7066
CRC checksums:
  Cipher key table 1: 0x5cbe2d0f
  Cipher key table 2: 0x7ba1a34f
PAK file header - ID: fpkr, files: 44, Block size: 2048, unknown1: 0x00000000
Index: 1 - Ident: rkpf, filepath: MBRDM.XIN, size: 0, crc32:OK
Index: 2 - Ident: rkpf, filepath: MBR_A.XIN, size: 153974, crc32:OK
Index: 3 - Ident: rkpf, filepath: SYSTEM/OSD/SNDIMAGE, size: 407188, crc32:OK
Index: 4 - Ident: rkpf, filepath: SYSTEM/OSD/ICOIMAGE, size: 105122, crc32:OK
Index: 5 - Ident: rkpf, filepath: SYSTEM/OSD/FNTOSD, size: 1489358, crc32:OK
Index: 6 - Ident: rkpf, filepath: SYSTEM/OSD/TEXIMAGE, size: 236964, crc32:OK
Index: 7 - Ident: rkpf, filepath: SYSTEM/OSD/JISUCS, size: 53186, crc32:OK
Index: 8 - Ident: rkpf, filepath: SYSTEM/OSD/OSDSYS_A.XLF, size: 712870, crc32:OK
Index: 9 - Ident: rkpf, filepath: SYSTEM/OSD/SKBIMAGE, size: 247891, crc32:OK
Index: 10 - Ident: rkpf, filepath: SYSTEM/FSCK/FSCK_A.XLF, size: 433825, crc32:OK
Index: 11 - Ident: rkpf, filepath: SYSTEM/FSCK/FILES/FILES_A.PAK, size: 3525952, crc32:OK
Index: 12 - Ident: rkpf, filepath: SYSCONF/FONT/S22J201.GF, size: 61952, crc32:OK
Index: 13 - Ident: rkpf, filepath: SYSCONF/FONT/S22J213.GF, size: 2138312, crc32:OK
Index: 14 - Ident: rkpf, filepath: SYSCONF/FONT/S22I646.GF, size: 30976, crc32:OK
Index: 15 - Ident: rkpf, filepath: SYSCONF/FONT/S22ULST.GF, size: 61952, crc32:OK
Index: 16 - Ident: rkpf, filepath: SYSCONF/FONT/S26J201.GF, size: 86528, crc32:OK
Index: 17 - Ident: rkpf, filepath: SYSCONF/FONT/S26J213.GF, size: 2986568, crc32:OK
Index: 18 - Ident: rkpf, filepath: SYSCONF/FONT/S26I646.GF, size: 43264, crc32:OK
Index: 19 - Ident: rkpf, filepath: SYSCONF/FONT/S26ULST.GF, size: 86528, crc32:OK
Index: 20 - Ident: rkpf, filepath: SYSCONF/FONT/SCE20I22.GF, size: 2138312, crc32:OK
Index: 21 - Ident: rkpf, filepath: SYSCONF/FONT/SCE24I26.GF, size: 2986568, crc32:OK
Index: 22 - Ident: rkpf, filepath: SYSCONF/ICON/ICON.SYS, size: 318, crc32:OK
Index: 23 - Ident: rkpf, filepath: SYSCONF/ICON/OTHERS.ICO, size: 33688, crc32:OK
Index: 24 - Ident: rkpf, filepath: SYSCONF/ICON/AUDIO/ICON.SYS, size: 317, crc32:OK
Index: 25 - Ident: rkpf, filepath: SYSCONF/ICON/AUDIO/AUDIO.ICO, size: 33688, crc32:OK
Index: 26 - Ident: rkpf, filepath: SYSCONF/ICON/HTML/ICON.SYS, size: 316, crc32:OK
Index: 27 - Ident: rkpf, filepath: SYSCONF/ICON/HTML/HTML.ICO, size: 33688, crc32:OK
Index: 28 - Ident: rkpf, filepath: SYSCONF/ICON/IMAGE/ICON.SYS, size: 317, crc32:OK
Index: 29 - Ident: rkpf, filepath: SYSCONF/ICON/IMAGE/IMAGE.ICO, size: 33688, crc32:OK
Index: 30 - Ident: rkpf, filepath: SYSCONF/ICON/TEXT/ICON.SYS, size: 316, crc32:OK
Index: 31 - Ident: rkpf, filepath: SYSCONF/ICON/TEXT/TEXT.ICO, size: 33688, crc32:OK
Index: 32 - Ident: rkpf, filepath: SYSCONF/ICON/VIDEO/ICON.SYS, size: 317, crc32:OK
Index: 33 - Ident: rkpf, filepath: SYSCONF/ICON/VIDEO/VIDEO.ICO, size: 33688, crc32:OK
Index: 34 - Ident: rkpf, filepath: SYSCONF/CONF/SYSTEM.INI, size: 382, crc32:OK
Index: 35 - Ident: rkpf, filepath: SYSCONF/CONF/SYSTEM101.INI, size: 430, crc32:OK
Index: 36 - Ident: rkpf, filepath: SYSCONF/CONF/SYSTEMOUT.INI, size: 430, crc32:OK
Index: 37 - Ident: rkpf, filepath: SYSCONF/CONF/SYSTEMEUK.INI, size: 430, crc32:OK
Index: 38 - Ident: rkpf, filepath: SYSCONF/CONF/SYSTEMFCA.INI, size: 430, crc32:OK
Index: 39 - Ident: rkpf, filepath: SYSCONF/CONF/SYSTEMFRE.INI, size: 430, crc32:OK
Index: 40 - Ident: rkpf, filepath: SYSCONF/CONF/SYSTEMGER.INI, size: 430, crc32:OK
Index: 41 - Ident: rkpf, filepath: SYSCONF/CONF/SYSTEMITA.INI, size: 430, crc32:OK
Index: 42 - Ident: rkpf, filepath: SYSCONF/CONF/SYSTEMPOR.INI, size: 430, crc32:OK
Index: 43 - Ident: rkpf, filepath: SYSCONF/CONF/SYSTEMSPA.INI, size: 430, crc32:OK
Index: 44 - Ident: rkpf, filepath: SYSCONF/CONF/FILETYPE.INI, size: 194, crc32:OK
Files decrypted: 44, Errors: 0

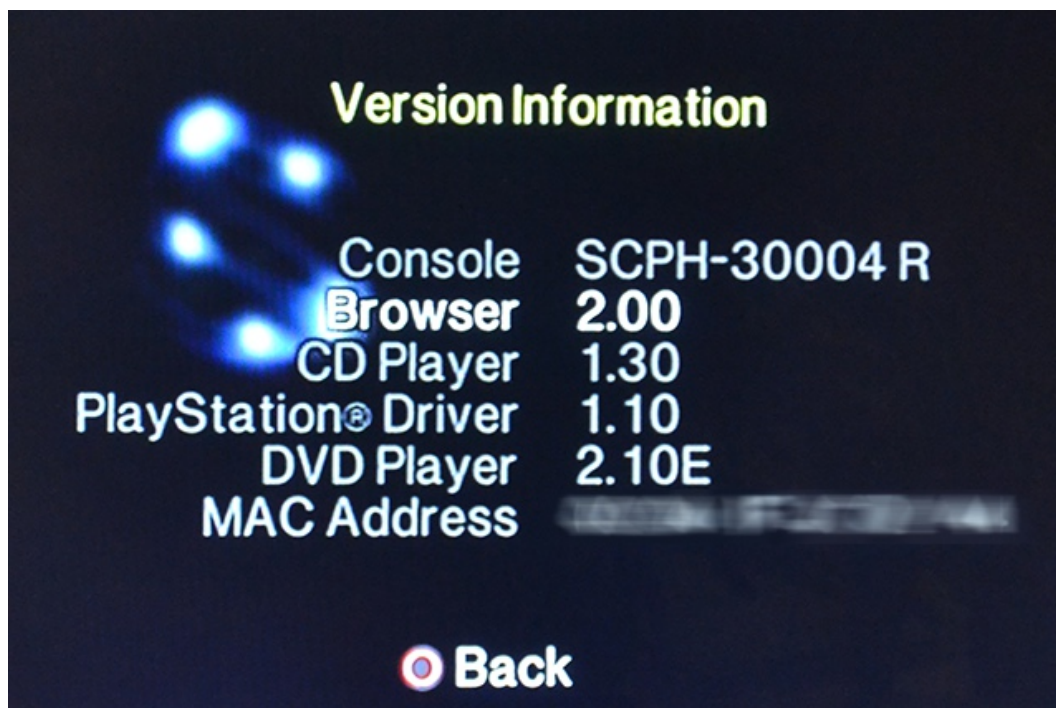
C:\test>
```

3. W miejscu w którym uruchomiłeś program, utworzony zostanie folder "extracted", a w nim pliki i foldery, które należy przenieść na dysk twardy PS2 (za pomocą uLE z np. pendrive'a (czyli z "mass:") lub za pomocą **PFS Shell** na Windows po podłączeniu dysku do komputera).

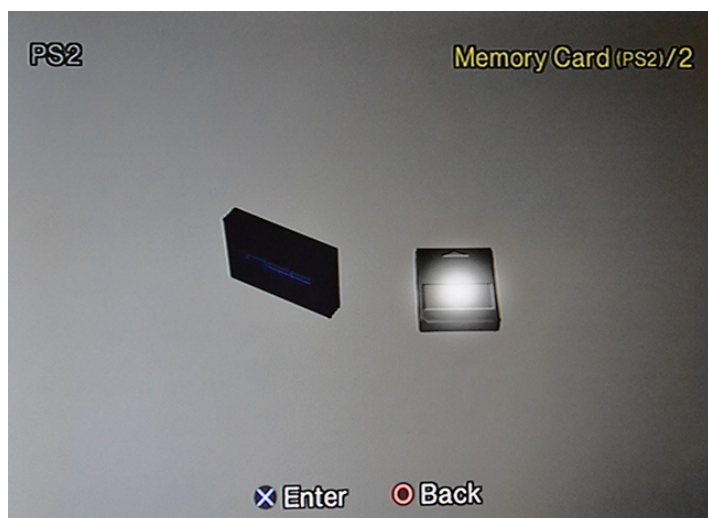
"MBR_A_XIN" i "MBRDM.XIN" można wyrzucić, **zawartość** "\extracted\SYSCONF\" należy skopiować na "hdd0:/_sysconf", z kolei "\extracted\SYSTEM\" na "hdd0:/_system". Dodatkowo nazwę pliku z "OSDSYS_A.XLF" należy zmienić na "osdmain.elf", "osd100.elf" bądź "osd110.elf" (jeśli chcesz zastąpić FHDB) lub "hosdsys.elf" (jeśli chcesz aby FHDB zaadoptował HDD OSD).



4. Jeśli wszystko wykonałeś poprawnie to po uruchomieniu konsoli, we właściwościach zobaczysz Browser w wersji 2.00, a nie starszej.



Jeśli nie zamierzasz wgrywać programów i gier na preparowane partycje (o czym później), tak aby imitowały oryginalne i dzięki temu uruchamiać je bezpośrednio z „browsera” to sens instalacji HDD OSD jest znikomy. Nawet save’y zamiast kopiować na HDD, znacznie bezpieczniej jest archiwizować na pendrive (opcja "PSU Paste" w uLE).



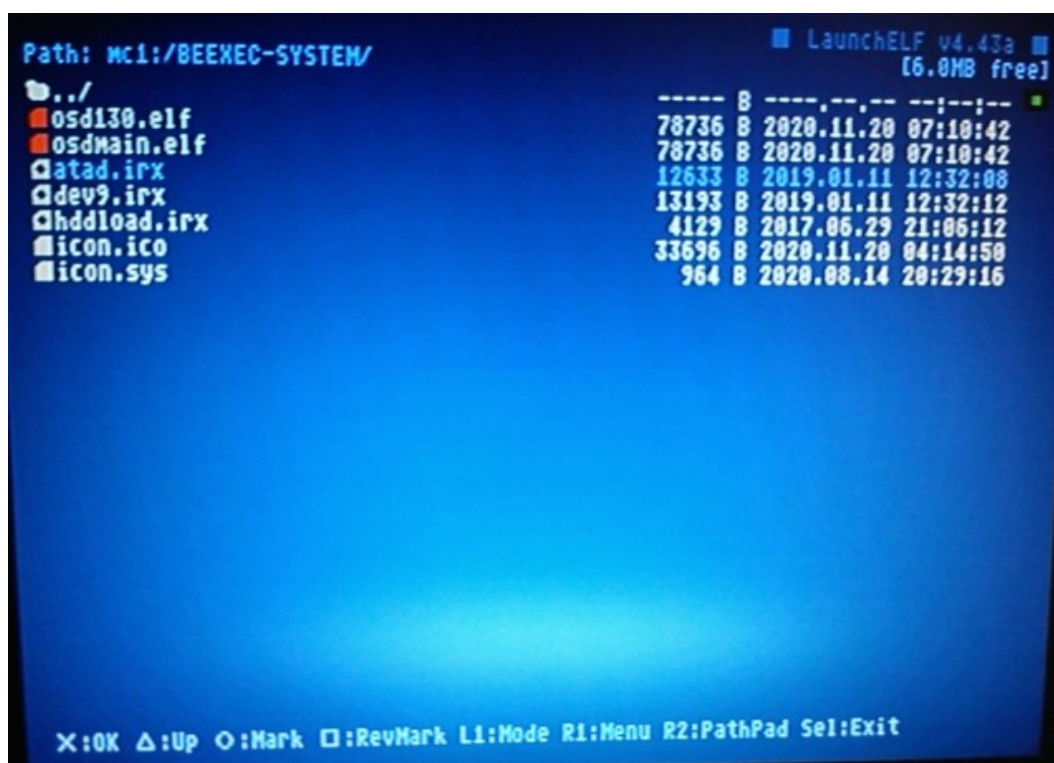
Dysk twardy symbolizuje ikona NA, z kolei folderu w "hdd0:/__common/" walizka.

HDD OSD na SCPH-7K

Na modelach SCPH-70xxx i 75xxx również jest możliwe uruchomienie HDD OSD, ale tylko jeśli przebrnąłeś przez żmudny proces lutowania taśmy do płyty głównej (inaczej oczywiście nie podłączysz dysku twardego, tak aby pełnił rolę tego wewnętrznego). Jakby problemów było mało, to Sony wyłączyło OSD Update dla HDD w tej serii konsol – a mówiąc po ludzku – nie jest już możliwy automatyczny start FHDB i/lub HDD OSD wyłącznie z samego dysku. Na szczęście można uzyskać podobny efekt za pomocą zainstalowanego na karcie pamięci FMCB (w wersji 1.966), który to ma własną implementację tej funkcji i odpali dla ciebie FHDB z dysku, przy okazji adaptując HDD OSD.

Procedura jest identyczna jak dla zwykłych fatów, z tą różnicą, że dodatkowo niezbędny będzie zainstalowany na karcie FMCB z dodatkowymi modułami: "atad.irx", "dev9.irx" i "hddload.irx". Czyli po wgraniu FMCB, uruchom uLE i skopiuj z np. pendrive wyżej wymienione niezbędne pliki, które znajdziesz w **instalatorze** (w katalogu "**INSTALL\SYSTEM**", pamiętaj aby ich nazwy miały małe litery, duże nie zostaną znalezione), a które musisz dorzucić do folderu "**B*EXEC-SYSTEM**" na „memorce” (pod gwiazdką kryje się region konsoli). Jeśli instalowałeś wersję multiregionalną to te moduły znajdziesz w "BIEXEC-SYSTEM" i stamtąd je możesz przekopiować gdzie indziej.

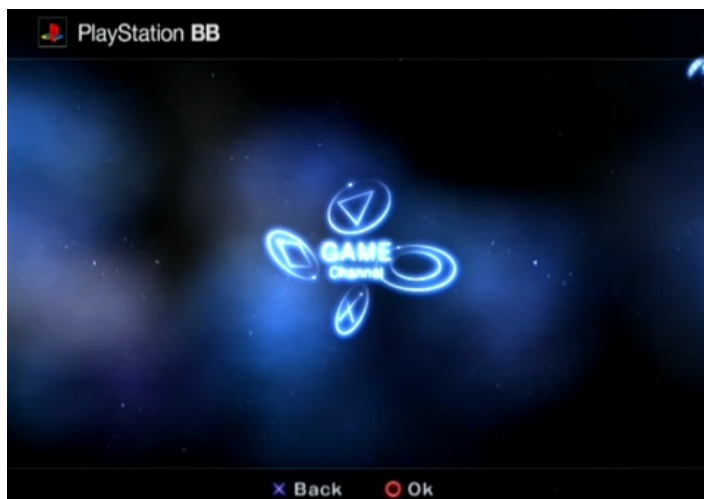
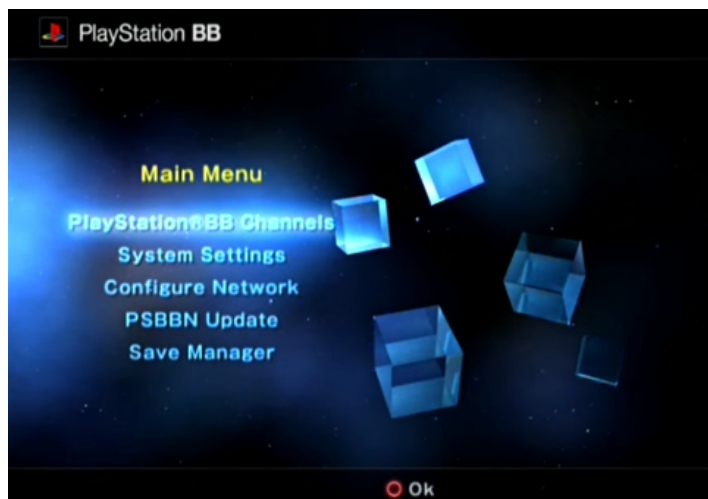
Europa	mc0:/B E EXEC-SYSTEM/
Ameryka Północna i część Azji	mc0:/B A EXEC-SYSTEM/
Chiny	mc0:/B C EXEC-SYSTEM/
Japonia	mc0:/B I EXEC-SYSTEM/



W takiej konstelacji, PS2 uruchomi FMCB z karty, a ten z kolei resztę z HDD (nie zdziw się kiedy zobaczysz dwa razy logo FMCB, a samo bootowanie nieco się wydłuży).

PlayStation BB Navigator

PlayStation BroadBand Navigator lub w skrócie **PSBBN** to specjalna dystrybucja Linuksa, wydana wyłącznie w Japonii. Z czystym sumieniem cały PSBB można nazwać protoplastą PlayStation Store zaś PSBBN klientem po stronie użytkownika. Dzisiaj, czyli już dawno po wyłączeniu wszystkich kanałów wydawców/deweloperów gier, ten egzotyczny wynalazek stanowi wyłącznie ciekawostkę i fragment historii konsoli. Jest powolny w działaniu i w gruncie rzeczy bezużyteczny poza ciekawym interfejsem i możliwością oglądania filmów i słuchania muzyki.



Niestety mimo obietnic Sony, „nawigator” nigdy nie opuścił granic kraju kwitnącej wiśni i tak jak w przypadku Utility Disc, **zabezpieczenia wymuszają japońską konsolę bez modchipa (zainstalowanie z kopii via ESR lub obrazu płyty via HDL/OPL nie jest możliwe) i oryginalny dysk twardy**. Spatchowanych instalatorów nigdy nie widziałem, co innego spatchowany PSBBN (i przetłumaczony na język angielski) w gotowych do wgrania obrazach HDD. Niestety nie będę mógł cię przeprowadzić przez proces instalacji, dlatego że nie mam odpowiedniego sprzętu, przeszczep też nie za bardzo jest możliwy ponieważ nie wszystkie dane wgrywane są na partycje z PFS, EXT2 i RFS (jedna jest surowa lub z nieznanym systemem plików). Nikt do tej pory tego w całości nie złamał, więc pozostają ci cudze posektorowe obrazy HDD, które możesz namalować na dowolnym dysku np. programem **HDD Raw Copy Tool** lub **DMDE**.

MBR Boot

Partycja "`hdd0:/_mbr`", tak jak i wszystkie pozostałe, posiada specjalnie wydzielone miejsce na „krypto elfa”. O ile aby uruchomić coś z „partycji PP” trzeba posiadać wgrany HDD OSD i przebić się przez kilka menu, o tyle w przypadku „MBR” nie, i tam wgrana aplikacja zostanie automatycznie odpalona przy starcie konsoli. Jest to ekwiwalent OSD Update (czyli np. FHDB).

Na adresie **0x130** znajduje się numer sektora, na którym OSDSYS poszukuje pliku. Sektory są reprezentowane przez system LBA, gdzie w tym przypadku jednostką jest zawsze 512 bajtów. Żeby nie było za łatwo to w odwróconym porządku bitów. ;) Czyli na przykładzie FHDB, znajdziesz tam **0020h**. Musisz więc te wartości zamienić miejscami uzyskując w ten sposób 2000h. Teraz, zmieniając tę liczbę z systemu szesnastkowego na bardziej przyjazny normalnym ludziom, czyli na dziesiętny, okazuje się że jest to 8192d. Skoro jest to numer sektora, a wielkość każdego jest równa 512B, to musisz wykonać mnożenie tych liczb i tak odkryjesz 4194304d. Wystarczy teraz ponownie skorzystać z konwersji systemu liczbowego, tym razem z dziesiętnego na szesnastkowy, aby poznać adres na jakim znajduje się KELF, czyli w tym przypadku **400000h**. Adres można ustalać dowolny, rzecz jasna pod warunkiem, że nie będzie kolidować z żadną inną ważną informacją (np. częścią indeksu). Zaleca się konkretnie ten ponieważ Sony również go używa.

Podobnie jest z rozmiarem pliku, który znajdziesz na **0x134**. Również musisz dokonać wspomnianych wyżej przekształceń. Na przykładzie FHDB rozmiarem jest 9800h, co po matematycznych perypetiach da 77824B. I tyle dokładnie waży loader FHDB. Jeśli ta wielkość pliku będzie niezgodna z rzeczywistością, OSDSYS nie uruchomi KELFa, konsola będzie długo się włączać i zawiesi w menu. Maksymalnym rozmiarem jaki można użyć jest **883200B**.

Program też nie może być dowolny, tylko taki który załaduje się na fizycznym adresie w pamięci **0x00100000**. Można to sprawdzić np. na Linuksie na komputerze, używając "`readelf -l <nazwa aplikacji>`". Później należy mu obciąć nagłówek i przerobić na KELF (opis jak to zrobić znajdziesz w rozdziale pt. „Tworzenie partycji PP”).

```
mint@mint: ~/ps2
mint@mint:~/ps2$ readelf -l "Open PS2 Loader (VMC) v0.9.3.elf"

Elf file type is EXEC (Executable file)
Entry point 0x1d00008
There is 1 program header, starting at offset 52

Program Headers:
Type           Offset    VirtAddr   PhysAddr   FileSiz MemSiz  Flg Align
LOAD          0x000060 0x01c33bc0 0x01c33bc0 0xcc615 0xcc6c0 RWE 0x10
mint@mint:~/ps2$ readelf -l "unofficial LaunchELF kHn.elf"

Elf file type is EXEC (Executable file)
Entry point 0x100008
There are 2 program headers, starting at offset 64

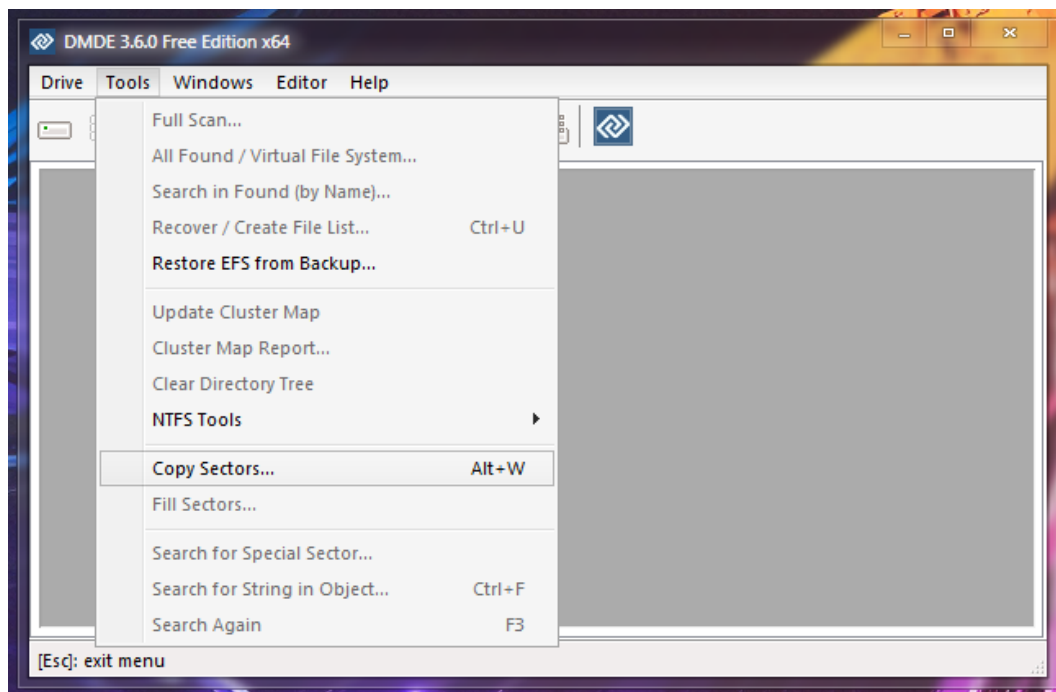
Program Headers:
Type           Offset    VirtAddr   PhysAddr   FileSiz MemSiz  Flg Align
LOAD          0x000080 0x00100000 0x00100000 0x00650 0x00740 RWE 0x10
LOAD          0x0006d0 0x00100740 0x00100740 0x6fc30 0x6fc30 RWE 0x10
mint@mint:~/ps2$ |
```

Niekompatybilny OPL i kompatybilny uLE kHn.

Przyznam szczerze że nie znalazłem żadnej aplikacji spełniającej wszystkie powyższe warunki (czyli i limitu rozmiaru i adresu w pamięci operacyjnej) dlatego jeśli potrzebujesz takiej funkcjonalności i na dodatek w postaci menedżera plików, to polecam po prostu wgrać **unofficial LaunchELF kHn**, który został już dostosowany przez autora.

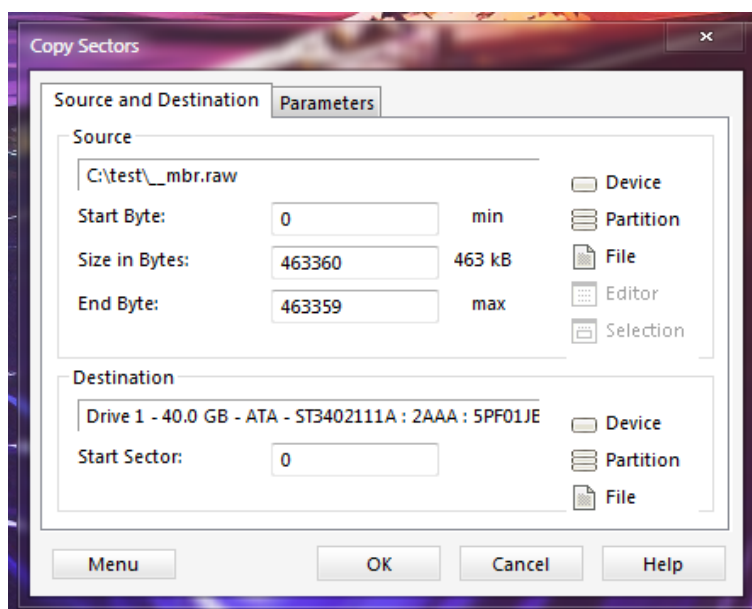
I tutaj możliwe drogi są dwie: albo kombinujesz ze wstrzykiwaniem pliku ("MBR.KELF") na konkretny adres (czyli gimnastykując się z kalkulatorem, ale zachowując wszystkie partycje), albo nadpisujesz „MBR” specjalnie spreparowanym kawałkiem tej partycji ("__mbr.raw"), ale jednocześnie **tracąc wszystkie kolejne**. Poniższy opis opieram na drugim wariantcie, ponieważ jest najprostszy i nie wymaga obycia z edytorem heksadecymalnym.

1. Uruchom program **DMDE** na prawach administratora i z menu "Tools", wybierz opcję "Copy Sectors...".

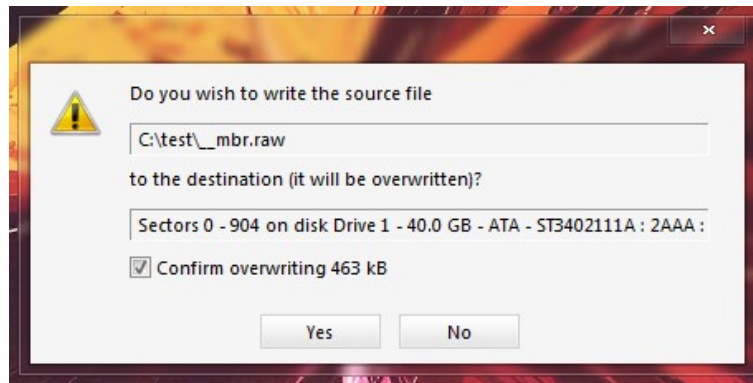


2. W kategorii "Source" kliknij w przycisk "File" i wybierz plik "__mbr.raw". "Start Byte" powinien być zerowy. Z kolei w kategorii "Destination" kliknij w przycisk "Device" i z "Physical Devices" wybierz podłączony do komputera dysk twardy z PS2. "Start Sector" również powinien wynosić zero.

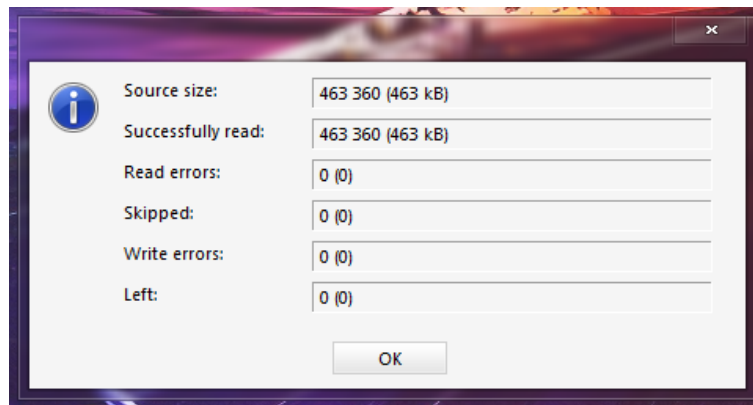
Po upewnieniu się, że wybrany HDD jest właściwy, ;) kliknij w przycisk "OK".



Zostaniesz jeszcze zapytany o potwierdzenie, więc zaznacz stosowne pole i oczywiście kliknij w "Yes".



3. Po zakończonej operacji, w podsumowaniu nie powinieneś zobaczyć żadnych błędów.



Kopiowanie gier dla PSX na dysk twardy

POPS jest to oficjalny emulator PlayStation, który pobierało się razem z grą pt. „Bishi Bashi Stepchamp 3”, z kanału Konami. Innych gier nigdy w ten sposób nie wydano, a sam „emu” tak jak i reszta zawartości dla PSBBN, był zaszyfrowany i przypisany do tego konkretnego HDD i tej konkretnej konsoli, na której gra została pobrana. Ze względu na ten absurdalnie zamordystyczny DRM, brak kluczy i popularność „nawigatora” poza Japonią bliską zeru absolutnemu, bardzo długo nie zostało to złamane. Pierwsze publiczne odszyfrowane SLBB-0001 (czyli właśnie POPS) były testowe i na różne sposoby modyfikowane. „Czysta wersja”, na której opieram poradnik składa się z dwóch plików (w zasadzie to z trzech, ale "POPS_IOX.PAK" w przypadku HDD nie jest potrzebny), które musisz odnaleźć na własną rękę w otchłani internetu (pozostałe wersje nie będą kompatybilne):

POPS.ELF	emulator	MD5: 355A892A8CE4E4A105469D4EF6F39A42
IOPRP252.IMG	obraz pamięci IOP	MD5: 1DB9C6020A2CD445A7BB176A1A3DD418

Dodatkowo potrzebujesz programu **POPStarter**. Jest to loader, który patchuje w locie dziewiczą wersję, wzbogacając ją o różne dodatkowe funkcjonalności, ale przede wszystkim znacząco zwiększając kompatybilność z grami.

Przygotowanie środowiska

Są trzy sposoby na uruchamianie gier na PSX z obrazów płyt za pomocą oficjalnego emulatora Sony. Różnią się lokalizacją i nazewnictwem loadera. We wszystkich przypadkach, emulator i obraz IOP należy wrzucić do "[hdd0:/__common/POPS/](#)".

Wariant A

Jeśli chcesz na jednej partycji trzymać wszystkie gry na PSX to musisz wrzucić do "[hdd0:/__common/POPS/](#)" pliki emulatora i POPStarter pod nazwą "[POPSTARTER.ELF](#)". Wtedy gry wrzucasz do "[hdd0:/__.POPS/](#)" (tylko i wyłącznie stąd będą wczytywane) i uruchamiasz je po prostu wybierając obraz płyty [*.vcd](#) w managerze plików [uLE kHn](#) (na chwilę obecną ten sposób **nie** zadziała z żadną inną wersją [uLE](#) niż [kHn](#)).

Schemat:

```
"hdd0:/__common/POPS/POPSTARTER.ELF"  
"hdd0:/__.POPS/*.VCD"
```

Wariant B

Ewentualnie możesz też nazwać POPStarter dokładnie tak jak obraz płyty i wrzucić do "[hdd0:/__.POPS/](#)", ale wówczas niepotrzebnie duplikujesz loader dla każdej gry z osobna, choć z kolei wtedy nie jesteś już uwiązany do [uLE kHn](#) tylko jakiegokolwiek [uLE](#), bo gry odpalasz wybierając loader o nazwie tejsze gry.

Schemat:

```
"hdd0:/__.POPS/<tytuł gry>.VCD"  
"hdd0:/__.POPS/<tytuł gry>.ELF"
```

Wariant C

Jeśli wolisz mieć grę bootowalną, ale każdą z osobnej partycji to wtedy zmarnujesz mnóstwo miejsca na dysku (niewiele gier wstrzeli się w wielokrotność 128MiB) i trochę czasu na przygotowanie partycji PP (o tym w osobnym rozdziale). Zyskujesz za to atrakcyjną formę uruchamiania gry, bo bezpośrednio z menu HDD OSD. Obraz płyty musi nazywać się "[IMAGE0.VCD](#)", zaś POPStarter musi mieć nazwę, którą wpisujesz w pliku CNF (ale o tym również przeczytasz w części poświęconej tworzeniu partycji PP) i musi to być wersja podpisana, czyli [KELF](#) (ponieważ HDD OSD tylko takie uruchamia).

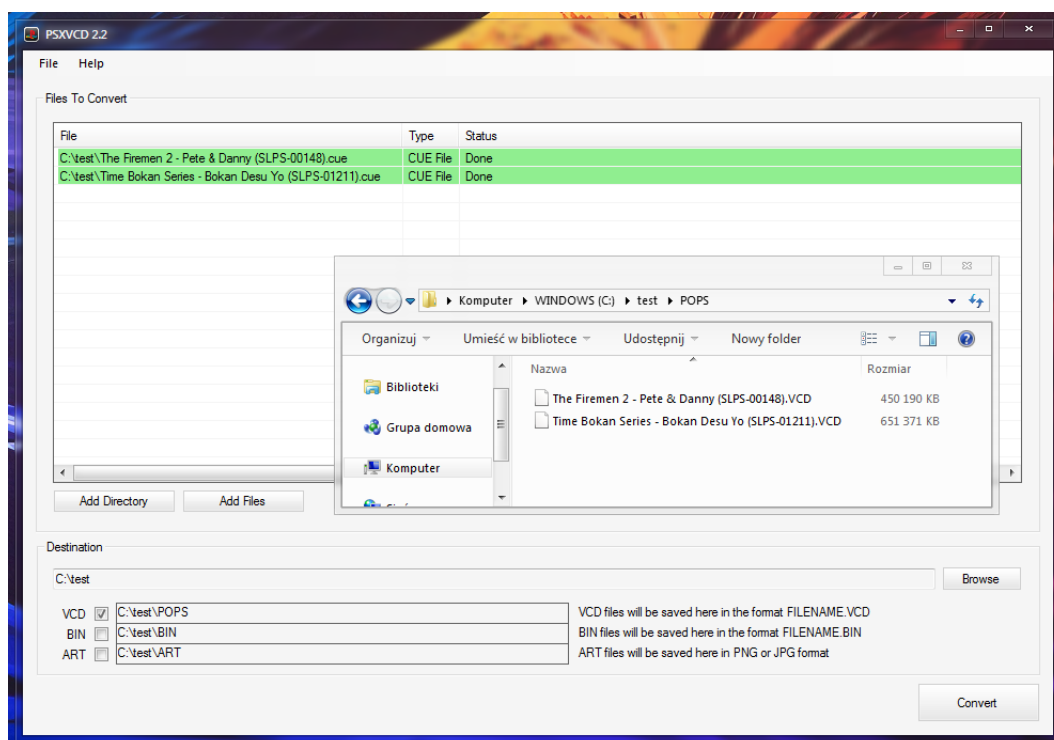
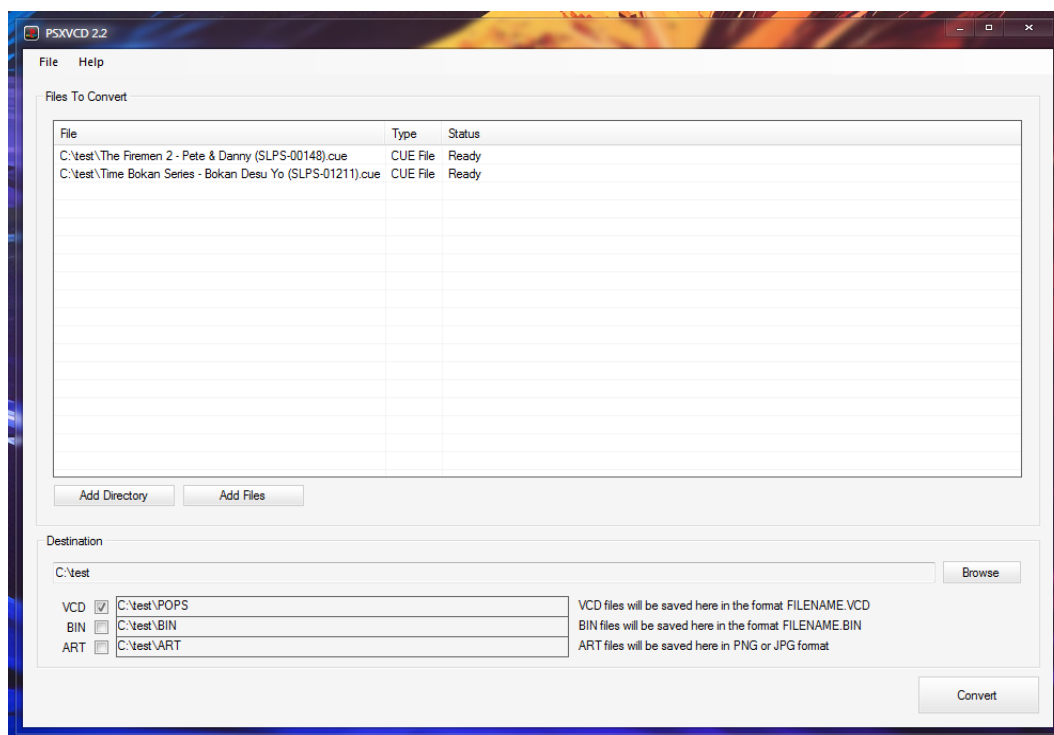
Schemat:

```
"hdd0:/PP.<tytuł gry>/IMAGE0.VCD" (w przypadku gier wielo-płytkowych, dla kolejnych krążków zamiast zera: 1, 2 i 3)  
"hdd0:/PP.<tytuł gry>/<nazwa zdefiniowana w "system.cnf">.KELF"
```

Konwersja gier

Obrazy płyt muszą mieć złączone wszystkie ścieżki CD-Audio (o ile gra takowe posiada) i zawierać TOC. Na szczęście jest narzędzie automatyzujące ten proces, czyli **PSXVCD**.

Uruchom program, dodaj obrazy płyt ("Add Files") lub katalog z obrazami ("Add Directory"), wybierz miejsce docelowe na obrazy w formacie *.vcd ("Browse") i kliknij w przycisk "Convert".



Rozszerzenie VCD koniecznie zostaw dużymi literami. Pamiętaj też aby nie przekraczać 33 znaków w nazwie pliku, ponieważ więcej nie zostanie wyświetlone w HDD OSD na podglądzie folderu gry.

Folder gry

Dla każdej uruchomionej gry (przynajmniej raz), POPStarter w swojej ostatniej wersji (czyli R13 Beta z dnia 2019-06-05) tworzy folder, w którym przechowywane są zasoby per tytuł ("[hdd0:/__common/POPS/<nazwa obrazu płyty>/](#)"), ale równie dobrze możesz go uprzedzić i nieco spersonalizować ten kąt.

Wirtualne karty pamięci

Czyli obrazy prawdziwych „memorek” powinny być już sformatowane i mieć dokładnie **128KiB** (a więc być bez śmieciowych nagłówków jakie dorzuca np. DexDrive (*.gme) czy POPS z PSP/PSV/PSTV (*.vmp)). Adekwatnie do slotu w prawdziwej konsoli, odpowiadają im pliki: "[SLOT0.VMC](#)" i "[SLOT1.VMC](#)". Możesz więc z powodzeniem wrzucić wirtualne karty PSX'a z np. PS3 (*.VM1) lub emulatora ePSXe (*.mcr), zmieniając im po prostu nazwy na wymagane (czyli np. z "epsxe000.mcr" na "SLOT0.VMC").

Problem pojawia się z grami wielopłytkowymi, ponieważ dla każdego obrazu płyty zostanie utworzony osobny folder m.in. z osobnymi „memorkami”. I o ile z tytułami pozwalającymi na zapis pomiędzy zmianą nośnika (np. „Parasite Eve”) ten problem ogranicza się do zapisu „ostatniego save’a”, wyłączenia gry i przekopiowania VMC do kolejnego folderu, o tyle są takie które takiej możliwości nie dają (np. „Chrono Cross”, „Metal Gear Solid”). Rozwiązania są dwa: albo za pomocą specjalnych **ripkitów** usuniesz kod odpowiedzialny za zmianę nośnika i scalisz wszystkie obrazy płyt w jeden, tworząc niezgodną ze sztuką i standardem, ale działającą hybrydę – albo zrobisz plik tekstowy "[VMCDIR.TXT](#)", w którym wpiszesz nazwę folderu na wspólne karty (np. "Schowek" i oczywiście taki też folder musisz stworzyć; nie wpisuj ścieżek). Jeśli nie jest to jeszcze jasne to spójrz na przykład poniżej:

```
"hdd0:/__common/POPS/Metal Gear Solid (CD 1)/VMCDIR.TXT"
```

```
"hdd0:/__common/POPS/Metal Gear Solid (CD 2)/VMCDIR.TXT"
```

```
"hdd0:/__common/POPS/Schowek/SLOT0.VMC"
```

```
"hdd0:/__common/POPS/Schowek/SLOT1.VMC"
```

Gry wielopłytkowe

A skoro już jesteśmy przy grach wydanych na więcej niż jednym krążku, to podobny problem pojawia się z wyborem kolejnej płyty. Stwórz więc w każdym z folderów odpowiadających każdej z osobna płycie, plik "[DISCS.TXT](#)". Na przykład dla MGS:

```
"hdd0:/__common/POPS/Metal Gear Solid (CD 1)/DISCS.TXT"
```

```
"hdd0:/__common/POPS/Metal Gear Solid (CD 2)/DISCS.TXT"
```

I wpisz tam nazwy obrazów płyt (maksymalnie cztery).

```
Metal Gear Solid (CD1).VCD
```

```
Metal Gear Solid (CD2).VCD
```

Wówczas kiedy gra zażąda zmiany nośnika, posłużysz się kombinacją przycisków:

Otwieranie kłapy	Select + L2 + R2 + Trójkąt
Wkładanie płyty nr. 1	Select + L2 + R2 + Góra
Wkładanie płyty nr. 2	Select + L2 + R2 + Prawo
Wkładanie płyty nr. 3	Select + L2 + R2 + Dół
Wkładanie płyty nr. 4	Select + L2 + R2 + Lewo
Zamykanie kłapy	Select + L2 + R2 + Kwadrat

Kopiowanie gier dla PS2 na dysk twardy

Gry na dysku twardym dla PlayStation 2 spoza kanałów PSBB – czyli wszystkie te wgrywane przez różne scenowe aplikacje – są obrazami płyt. W razie konieczności pociętymi na plastry jak ogórek do mizerii (względem fragmentacji partycji) i z zaokrąglonym wzwyz rozmiarem, tak aby mieścić się w limicie jaki narzuca nieszczęsna APA.

Są różne metody na wgrywanie gier i każda wiąże się z różnymi niuansami. Można gry kopiować za pomocą **WinHIIP** do fizycznie podłączonego do komputera dysku (nieradzi sobie z dużymi HDD, ale np. jako jedyny obsługuje APAEXT). Analogicznie można tak samo zrobić za pomocą **HDL Dump** lub **HDL Dumb** (również przez sieć). Można także użyć moim zdaniem najlepszego programu, czyli **HDL Game Installer**. Jest to aplikacja z poziomu której gry można instalować bezpośrednio z napędu optycznego konsoli, ale jest to także serwer dla klienta po stronie Windows. Jeśli twój czytnik płyt w PS2 ledwo zipie lub jest zabrudzony, ewentualnie jeśli gra, którą chcesz skopiować została wydana na DVD9, czyli płycie dwuwarstwowej to należy skorzystać z metody sieciowej. Jeśli chcesz zainstalować grę wydaną na DVD9, ale koniecznie z płyty włożonej do konsoli, to wówczas użyj **HDLloader**.

HDLGame Installer i HDL Dump/Dumb od razu wgrywają w pole atrybutów ikonę, opis i mini wersję Open PS2 Loadera, więc teoretycznie wszystkie gry da się uruchamiać z poziomu menu PS2 bez uruchamiania dodatkowych programów (oczywiście tylko wtedy gdy masz zainstalowany HDD OSD).



HDL Game Installer - wariant płytowy

Uwaga: brak obsługi gier na płytach DVD dwuwarstwowych (potocznie nazywanymi DVD9 lub DVD DL).

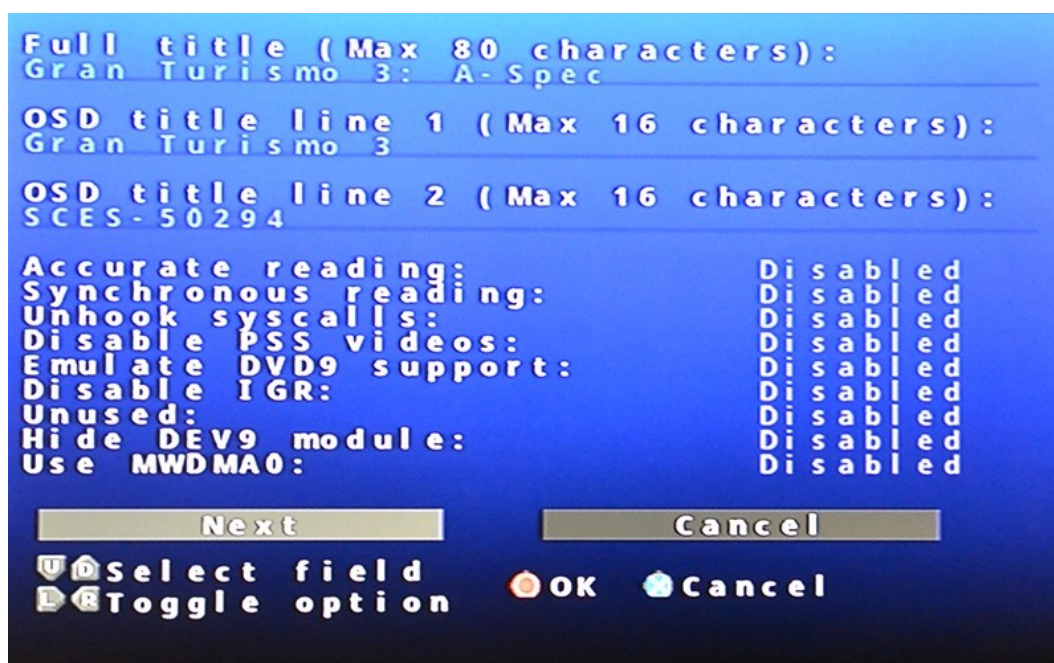
1. Uruchom program **HDL Game Installer** i odczekaj chwilę aż napęd rozpozna płytę. Kiedy przestanie rzeździć, wciśnij przycisk Start.

Kolejność ma tutaj znaczenie tylko dla osób ze słabym laserem lub brudną soczewką i/lub lustrami, ponieważ z jakiegoś powodu „browser” kiedy nie może się doczekać ;) aż Mechacon pozwoli mu na dostęp do nośnika, wyświetli „czerwone chmury” i trzeba ponownie wysunąć/wsunąć tackę („modele monolity”) lub otworzyć/zamknąć klapę (modele Slim).

2. Nadaj tytuł grze wyświetlany w menu HDLoadera i Open PS2 Loadera (maksymalnie możesz użyć 80 znaków). Tak samo zrób z tytułem i podtytułem wyświetlanym w „browserze” (16 znaków na każdy z osobna).

Dodatkowo możesz oznaczyć specjalne tryby pracy dla loadera, ale jeśli nie wiesz jakie i po co, to zostaw z domyślnymi wartościami. Zmienić te opcje można w każdej chwili także po instalacji gry.

Zaznacz "Next" i potwierdź wybór.



3. Program zapyta o potwierdzenie, więc wybierz "OK". Kolejne pytanie dotyczyć będzie użytej ikony. Jeśli masz save'a z tej gry na karcie pamięci (bo tylko stamtąd zostanie wczytana) to polecam wybrać "Game save". Na pewno będzie ładniejsza niż ta domyślna, czyli z antycznego dziś HDLoadera. Wybór oczywiście potwierdź "OK".

4. Rozpocznie się proces kopiowania gry na HDD, co reprezentuje procentowy pasek postępu, a po zakończeniu wyświetli komunikat "Game installed successfully!".



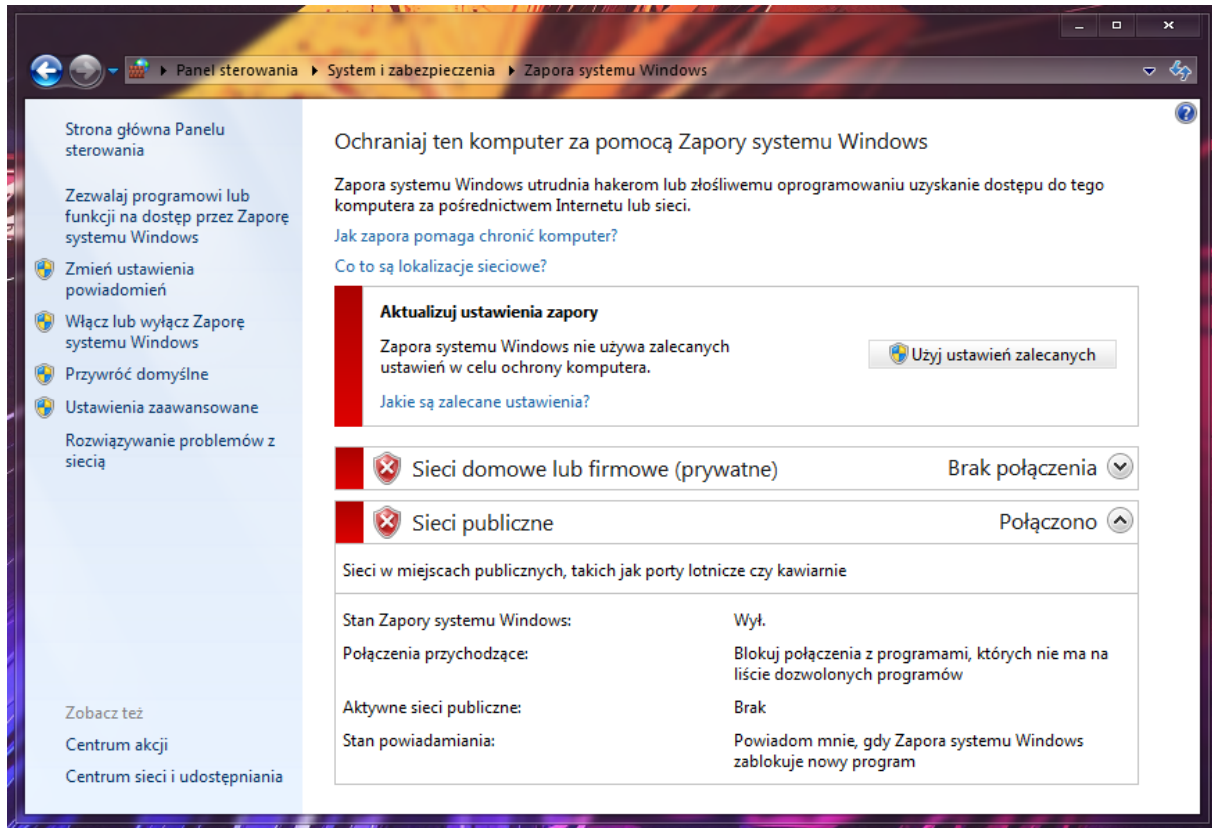
5. Po powrocie do menu głównego aplikacji, powinieneś zobaczyć grę na liście. To tutaj można zmieniać „tryby gry” ("Game options"), o których więcej dowiesz się w części poświęconej znaczeniu opcji.



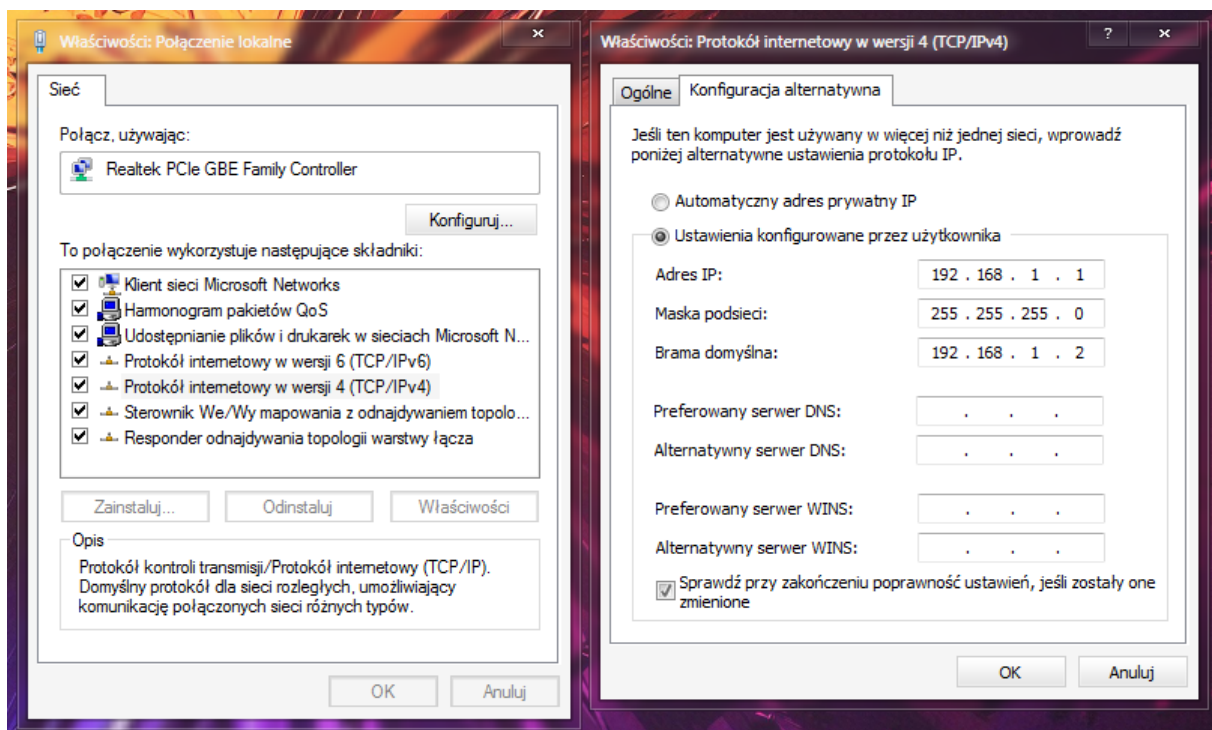
HDL Game Installer - wariant sieciowy

1. Przejdź do ustawień zabezpieczeń w Windows i wyłącz ogniomurek (lub odblokuj w nim dla TCP porty 45061 i 45062, ewentualnie przepuść program, który będzie rozmawiać z konsolą).

Na co dzień nie używam tego rozwiązania, mam tylko jedno gniazdo ethernetowe w komputerze, a konsola stoi tuż obok PC. Z lenistwa więc zamiast przebić się przez router, firewalle i inne cuda na kiju, poszedłem po linii najmniejszego oporu. Czego akurat oczywiście nie polecam ze względów bezpieczeństwa.



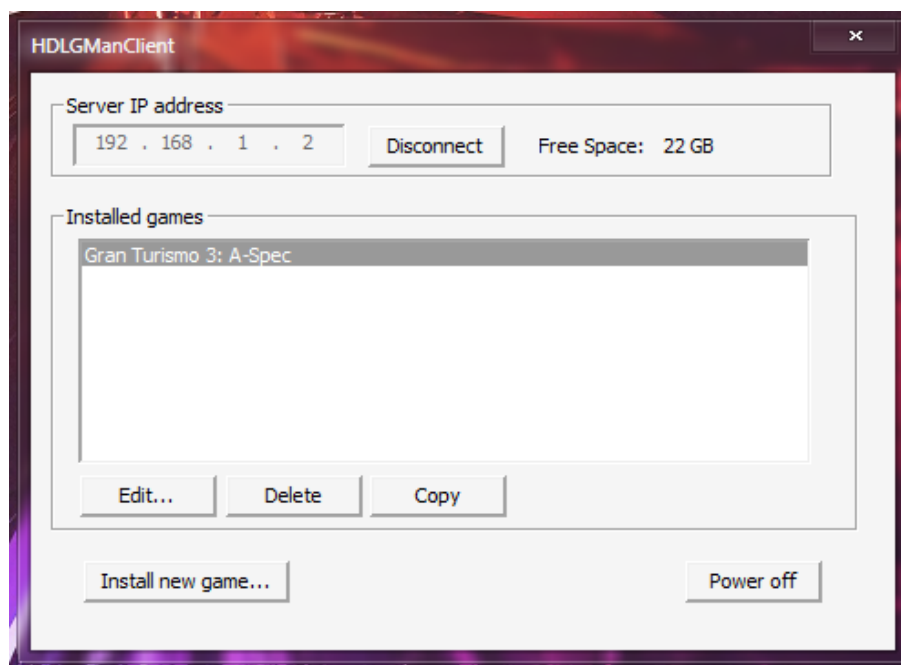
2. Przejdź do ustawień karty sieciowej i ustaw na sztywno swoją sieć lokalną.



3. Uruchom program **HDL Game Installer** i odczekaj chwilę aż spróbuje nawiązać połączenie z komputerem. Co prawdopodobnie zakończy się fiaskiem. Wejdź więc do ustawień wciskając Select, wyłącz DHCP ("Disabled") i ustaw adres, maskę i bramkę adekwatnie do ustawień w Windows.

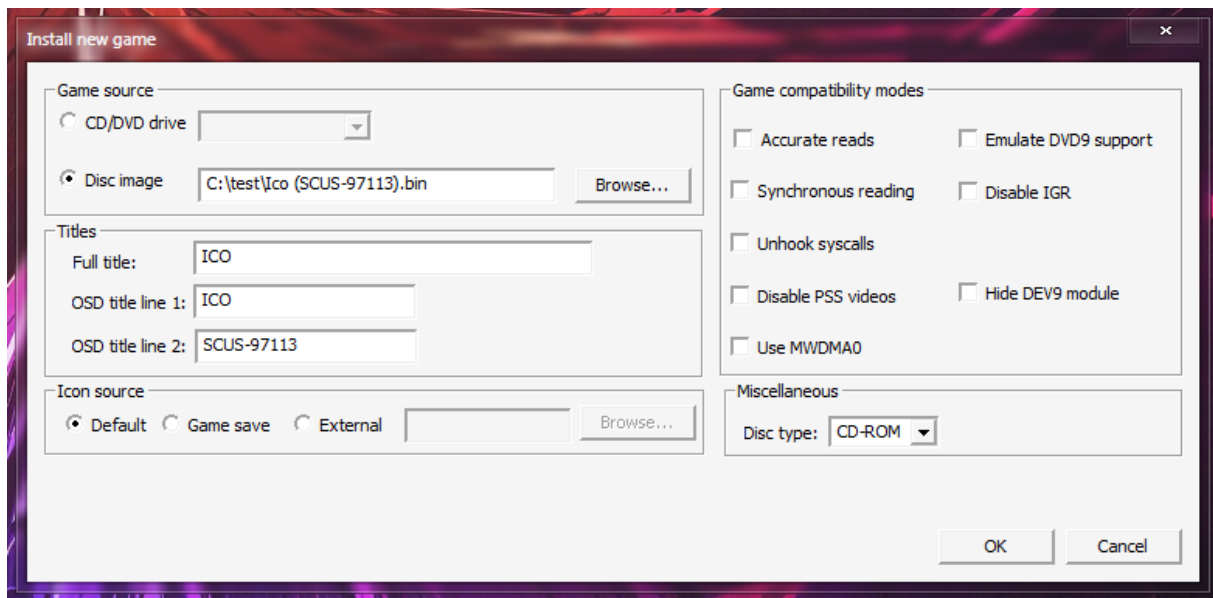


4. Wróć do PC i odpal **HDL Loader Game Manager Client for HDL Game Installer**, wpisz poprawny adres i kliknij w przycisk "Connect". Powinna pojawić się lista zainstalowanych na dysku gier.

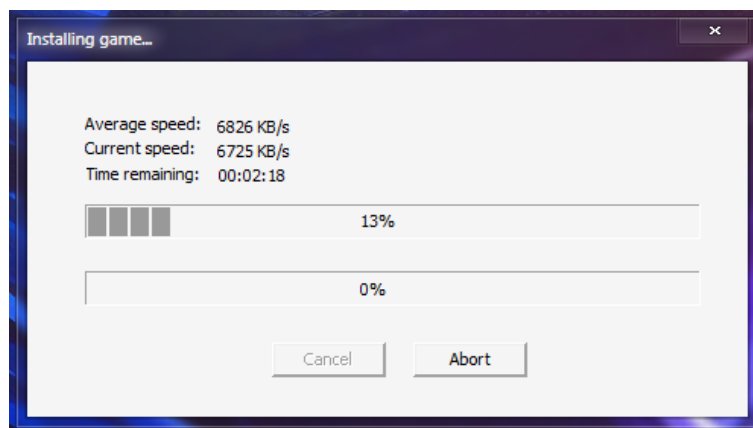


5. Kliknij w przycisk "Install new game...", a następnie w "Add Game". W kolejnym oknie wybierz sposób instalacji, czy z napędu optycznego ("CD/DVD drive"), czy z obrazu płyty ("Disc image"). W przypadku obrazu płyty, obsługiwane są *.iso dla DVD i *.bin dla CD (nie wskazuj na *.cue). Podobnie jak miało to miejsce w przypadku instalacji z płyty na konsoli, musisz określić tytuły wyświetlane w programach homebrew (limit 80 znaków) i menu konsoli (limit 16 znaków). Możesz także wybrać ikonę, np. z komputera i ustawić opcje uruchamiania gry. Wybierz też rodzaj nośnika ("CD-ROM" czy "DVD-ROM").

Na koniec kliknij w przycisk "OK"



6. Cofniesz się do menu "Job List" i jeśli nie chcesz dodawać kolejnych gier to wduś przycisk "Start" (w oknie programu rzecz jasna, nie na joypadzie). Zostaniesz jeszcze zapytany o potwierdzenie i rozpocznie się proces instalacji gry na konsoli.



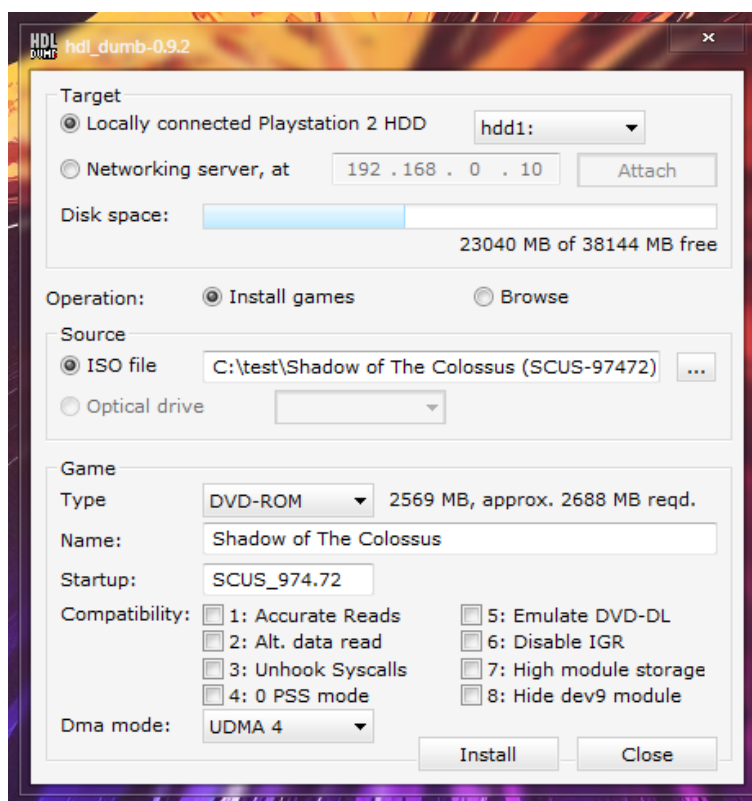
Po wszystkim wyświetli się komunikat "Games installed successfully". A w konsoli, po wyjściu z HDLGI, gra gotowa do uruchomienia.

HDL Dumb - wariant podłączeniowy

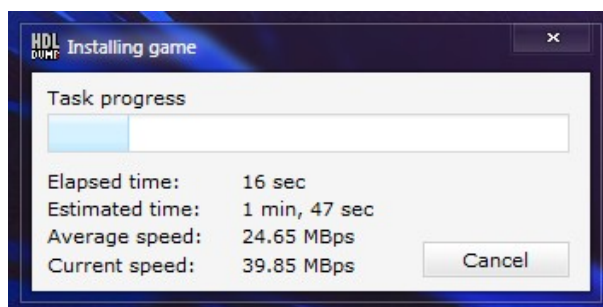
1. Podłącz dysk twardy z konsoli do komputera.

2. Uruchom program **HDL Dumb**. Zaznacz "Locally connected Playstation 2 HDD", który powinien zostać od razu wykryty. Wybierz lokalizację obrazu płyty (obsługuje formaty: *.iso, *.gi, *.iml/*, *.img, *.nrg, *.bin/*.cue). Określ rodzaj nośnika, tytuł, dodatkowe opcje i tryb transferu danych.

Na koniec kliknij w przycisk "Install".



3. Rozpocznie się proces instalowania gry. Po zakończeniu, okno zniknie, a gra będzie na Ciebie czekała już na HDD.



Znaczenie opcji

HDL Game Installer wgrywa w pole atrybutów program Open PS2 Loader w wersji bez interfejsu graficznego. Jest to starsza wersja niż obecne z GUI, dlatego część opcji może się różnić. Zawsze jednak można gry uruchamiać via zewnętrzny OPL, a nie ten z nagłówka i z tak samo je konfigurować.

- **Accurate reading** (mode 1)
Emuluje prędkość odczytu z płyty. Niektóre gry mają problemy z prędkościami wyższymi niż te jakie zapewnia napęd.
- **Synchronous reading** (mode 2)
Zmienia tryb odczytu z asynchronicznego na synchroniczny. Niektóre gry wymagają natychmiastowego odczytania zaraz po zażądaniu dostępu do danego sektora lub pliku, w przeciwnym razie się zawieszają.
- **Unhook syscalls** (mode 3)
Po włączeniu tej opcji, OPL nie pozostanie w RAM jeśli gra zresetuje (nadpisze) pamięć IOP wgrywając obraz IOPRP. Nie można wtedy używać ani IGR (InGameReset), ani GSM (Graphics Synthesizer Mode Selector), ani PS2RD (PS2 Remote Debugger (dawniej Artemis)).
- **Disable PSS videos** (mode 4)
Wyłącza odtwarzanie filmów w formacie *.pss. Przydatna opcja dla osób, które nie chcą oglądać „przerywników” w grze lub po prostu mają bardzo wolne pendrive’y (niektóre gry potrafią się z tego powodu zawiesić).
- **Emulate DVD9 support** (mode 5)
Według opisów opcja ta emuluje płytę dwuwarstwową dla gier zripowanych do jednowarstwowej, ale przyznam szczerze że nie bardzo rozumiem o co tutaj chodzi, skoro tzw. ripkity zajmują się zmianą media check na DVD5 i ewentualnym patchowaniem OTP na PTP.
- **Disable IGR** (mode 6)
Wyłącza możliwość resetowania gry za pomocą kombinacji przycisków joypada. Niektóre gry są z tym niekompatybilne i z IGR trzeba w ich przypadku zrezygnować.
- **High module storage** (mode 7, obecnie już nieużywany)
Zmienia adres przechowywania modułów w pamięci, dla gier które z nimi konfliktują.
- **Hide DEV9 module** (mode 8, obecnie już nieużywany)
Ukrywa moduł sieciowy jako „płytkowy”.
- **Use MWDMA0** (przesunięty w nowych OPL do osobnej kategorii trybu transferu danych z HDD)
Tryb odczytu HDD ustawiony jako MDMA-0. Zbliżony do prędkości odczytu z płyty. Jak to się ma do mode 1 to nie wiem, podejrzewam że zamiast tej opcji powinno używać się UDMA-4, a w razie potencjalnych problemów z tym związanych włączyć mode 1 (czyli "Accurate reading").

Tworzenie partycji PP

HDD OSD nie wyświetli żadnych innych partycji niż PP, PC i zawartości "[__common](#)" w „browserze” jeśli nie będzie mieć nazwy zgodnej z nomenklaturą, wgranej poprawnej ikony i pliku metadanych ją opisującą w polu atrybutów. Taka partycja albo zostanie zignorowana (systemowa), albo wyświetlona jako **Corrupted Data** (każda inna). Jedyną zaletą „partycji PP” jest możliwość automatycznego uruchamiania programów z poziomu „browsera”.

Przygotowanie ikony

Format ikon jest ten sam co dla kart pamięci PS2 (więc możesz wziąć dowolną z jakiegoś programu lub **zmaistrować własną**, nazwij ją "[list.ico](#)"), ale już plik je opisujący z jakiegoś tajemniczego powodu nie jest binarny, a tekstowy.

Stwórz prosty plik tekstowy ANSII z windowsowym łamaniem linii, uzupełnij wg. poniższego wzoru i zapisz jako "[icon.sys](#)".

PS2X	Nagłówek informujący PS2 o rodzaju zawartości (D: folder, X: wykonywalny).
title0=	Na przykład tytuł gry. Maksymalna ilość znaków to 16.
title1=	Na przykład identyfikator gry. Limit znaków ten sam co wyżej.
bgcola=0	Wartość przezroczystości tła.
bgcol0=0,0,0	Kolor tła w lewym górnym rogu. RGB.
bgcol1=0,0,0	Kolor tła w prawym górnym rogu. RGB.
bgcol2=0,0,0	Kolor tła w lewym dolnym rogu. RGB.
bgcol3=0,0,0	Kolor tła w prawym dolnym rogu. RGB.
lightdir0=1.0,-1.0,1.0	Kierunek padającego pierwszego światła. X,Y,Z.
lightdir1=-1.0,1.0,-1.0	Kierunek padającego drugiego światła. X,Y,Z.
lightdir2=0.0,0.0,0.0	Kierunek padającego trzeciego światła. X,Y,Z.
lightcolamb=64,64,64	Kolor ogólnego oświetlenia. RGB.
lightcol0=64,64,64	Kolor źródła pierwszego oświetlenia. RGB.
lightcol1=16,16,16	Kolor źródła drugiego oświetlenia. RGB.
lightcol2=0,0,0	Kolor źródła trzeciego oświetlenia. RGB.
uninstallmes0=	Wyświetlany komunikat podczas kasowania partycji.
uninstallmes1=	Jak wyżej, tylko drugi.
uninstallmes2=	Jak wyżej, tylko trzeci.

Przygotowanie CNF

Format pliku CNF jest podobny do tego dla płyt optycznych. Stwórz prosty plik tekstowy ANSII z windowsowym łamaniem linii, uzupełnij wg. poniższego wzoru i zapisz jako "[system.cnf](#)":

BOOT2 = pfs:/BOOT.KELF	Ścieżka bezwzględna do pliku wykonywalnego na partycji docelowej.
VER = 1.00	Wersja programu. Bez znaczenia.
VMODE = NTSC	Region. Również bez znaczenia.
HDDUNITPOWER = NICHDD	Odpowiada za niewyłączenie Network Adaptora i HDD.


Jeśli zamiast ścieżki do *.kelf wpiszesz "PATINFO" to plik zostanie wgrany w pole atrybutów, a nie w przestrzeń systemu plików PFS. Przydatne w zasadzie tylko jeśli nie zamierzasz w przyszłości zmieniać tego pliku (łatwiej bowiem podmienić plik np. managerem plików niż mordować się z linią poleceń). To samo tyczy się "IOPRP = " (jeśli potrzebujesz, powinien znaleźć się pod "BOOT2 = ").

Przygotowanie programu

Pliki wykonywalne na PS2 są w formacie ***.elf**, problem w tym, że HDD OSD uruchomi **tylko *.kelf**, czyli również „elfy”, ale podpisane i zaszyfrowane (potocznie nazywane **Krypto ELF**, mogą mieć też rozszerzenie ***.xlf**). Aby zrobić to poprawnie, należałoby użyć aplikacji **kelftool**. Niestety w obecnej wersji program pozwala na tworzenie „kelfów” wyłącznie dla OSD Update dla modeli DESR-xxxx. Pozostaje więc użyć **SCEDoormat**, który choć nie jest tak eleganckim rozwiązaniem jak wcześniej wspomniany kelftool, to przynajmniej umożliwia podpisanie pliku tak by HDD OSD nie wybrzydzał. ;)

1. Pobierz aplikację SCEDoormat (NoME) R3, razem z niezbędnym plikiem "KRYPTO.KHN".
2. Do katalogu z programem wrzucić „elfa”, którego chcesz uruchamiać po wybraniu partycji na PS2 w „browserze”.

Składnia jest prosta: "**<nazwa programu> <plik wejściowy> <plik wynikowy>**". W moim przypadku jest to akurat uLE nazwany "BOOT.ELF", zaś wypłuty plik "BOOT.KELF" (czyli zgodnie z tym co wpisałem wcześniej w "system.cnf").



```
C:\test>SCEDoormat (NoME) R3.exe BOOT.ELF BOOT.KELF

-----
| SCEDoormat (No ME Version) |
| Release 3, 2015/04/13     |
-----
INPUT.ELF == BOOT.ELF
           (1161208 bytes)
-----
OUTPUT.KELF == BOOT.KELF
             CREATED
-----
KRYPTO.KHN == KRYPTO.KHN
            (20971821 bytes)
-----

About KRYPTO.KHN :

Hello Gontlomom,
I was build 2014/04/22 to be bundled with the SCEDoormat No ME Version release 2,
and I'm suitable for retail PlayStation 2...

KRYPTO.KHN properties :
Maximum Capacity of the Container == 10485760 bytes (A00000h)
Precision Rate for Embedding Contents == 4 (WORST, round length to multiple of 16)
KELF Quantity == 655360
Length of the KELF Header == 128 bytes (0x0080)
Hashed Blocks Quantity == 1
Total Length of the MG Hashed Blocks == 8 bytes

KryptoELF specs :
User Header is MechaCon Blacklisted == False
Is a HDD Master Boot Record Container == True
Target Machine Type == PS2 (CEX Only)
Target Application Type == DVDPLAYER
M@t@cg@te Zones == Japan+USA+Europe+Oceania+Asia+Russia+China+Mexico

Writing the output KELF.....
Completed :) !

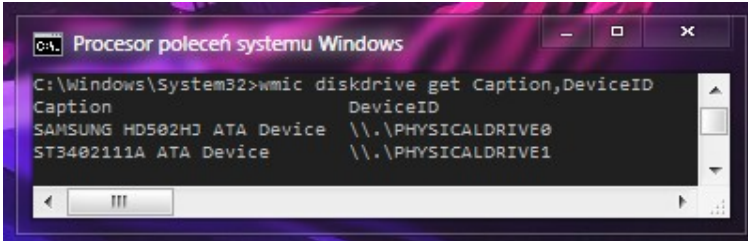
C:\test>
```

Modyfikacja pola atrybutów

Skoro wszystkie pliki masz już gotowe to wrzucić je w to samo miejsce co programy **PFS Shell** i **HDL Dump**.

Podłącz dysk do komputera pod kontroler PATA (czyli tradycyjnie taśmą 80 żyłową ATA i Molex'em). Jak zawsze w takich przypadkach, zalecam unikać kieszeni na USB ponieważ różnie bywa z ich jakością i wsparciem producenta (bywa że nie ma w pełni zaimplementowanego standardu ATA, a nawet że obsługa LBA48 odbywa się w jakiś pokraczny sposób).

1. Na samym początku uruchom CMD lub PowerShell jako **administrator**, ponieważ jeden z programów, których za chwilę użyjesz wymaga takich uprawnień.
2. Wpisz "**wmic diskdrive get Caption,DeviceID**" aby poznać ścieżkę UNC do dysku. Na przykładzie niżej jest to "\\PHYSICALDRIVE1" ponieważ moim modelem HDD używanym przez PS2 jest tym razem ST3402111A. Zapamiętaj ten numer ponieważ będzie jeszcze potrzebny.



```
C:\Windows\System32>wmic diskdrive get Caption,DeviceID
Caption                                DeviceID
SAMSUNG HD502HJ ATA Device            \\.\PHYSICALDRIVE0
ST3402111A ATA Device                  \\.\PHYSICALDRIVE1
```

3. Uruchom teraz **PFS Shell** i wklep: "**device \\.\PhysicalDrive1**". Wielkość liter nie ma znaczenia, wybrałem taką notację ponieważ jest według mnie czytelniejsza).

Niestety na Windows jest to najlepsza metoda na odwołanie się do fizycznego HDD, czyli skorzystanie z UNC. Na Linuksie wystarczy wylistować sobie urządzenia masowe przez "**lsblk**" i odwołać się standardowo do "**/dev/sdx**" lub "**/dev/loopx**". Protoplasta używał nomenklatury "**disk1:**", "**disk2:**" etc. ale na tej wersji to **nie zadziała**.

Jeśli HDD został wykryty to migający znak ">", zamieni się na "#".

4. Aby utworzyć nową partycję, wpisz "**mkpart PP.<etykieta> <rozmiar>**", czyli na moim przykładzie będzie to "mkpart PP.APPS 128".
5. Musisz także sformatować tę partycję, co zrób za pomocą polecenia "**mkfs <partycja>**". Adekwatnie do przykładu wyżej, będzie to "mkfs PP.APPS".
6. Zamontuj system plików z niedawno utworzonej partycji, wpisując "**mount <partycja>**", czyli np. "mount PP.APPS".
7. Skopiuj na dysk twardy konsoli plik "BOOT.KELF" wpisując "**put <plik>**", czyli w tym przypadku "put BOOT.KELF".
8. Na koniec wpisz "**umount**" aby odmontować partycję, a następnie "**exit**", aby opuścić powłokę.

```
Administrator: Procesor poleceń systemu Windows
C:\test>"PFS Shell v1.1.0.exe"
pfsshell for POSIX systems
https://github.com/uyjulian/pfsshell

This program uses pfs, apa, iomanX,
code from ps2sdk (https://github.com/ps2dev/ps2sdk)

Type "help" for a list of commands.

> device \\.\PhysicalDrive1
hdd: PS2 APA Driver v2.5 (c) 2003 Vector
hdd: max open = 1, 3 buffers
hdd: 17:40:52 08/10/2020
hdd: disk0: 0x04a891c0 sectors, max 0x00200000
hdd: checking log...
hdd: drive status 0, format version 00000002
hdd: version 0000 driver start.
pfs Playstation Filesystem Driver v2.2
ps2fs: (c) 2003 Sjeep, Vector and Florin Sasu
pfs Max mount: 1, Max open: 1, Number of buffers: 10
pfs version 0000 driver start.
# mkpart PP.APPS 128
# mkfs PP.APPS
pfs: Format: log.number = 8224, log.count = 16
pfs: Format sub: sub = 0, sector start = 8208, sector end = 8211
# mount PP.APPS
PP.APPS:/# put BOOT.KELF
PP.APPS:/# ls
drwxrwxrwx    512 2020-08-10 17:41 .
drwxrwxrwx    512 2020-08-10 17:41 ..
-rw-rw-rw-   454563 2020-08-10 17:41 BOOT.KELF
PP.APPS:/# umount
# exit

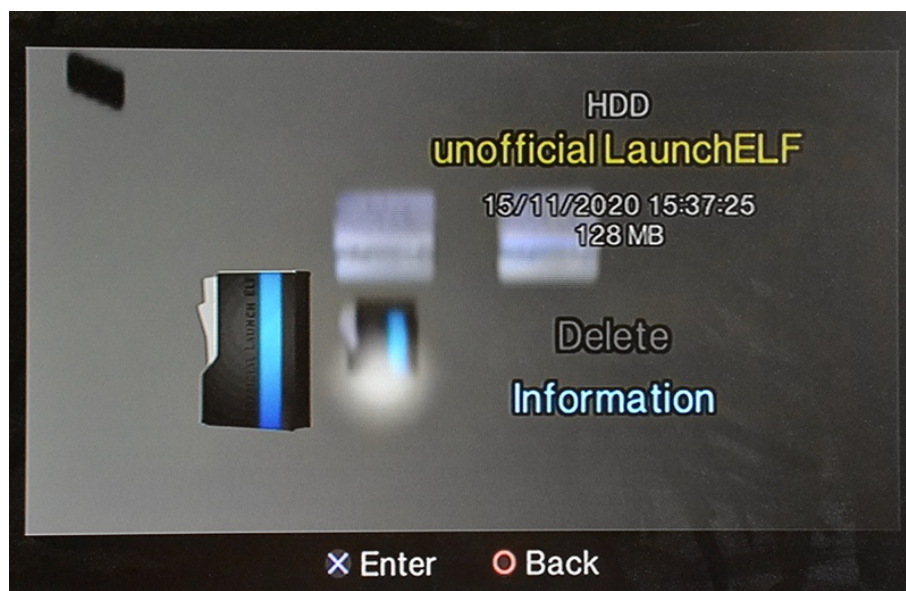
C:\test>
```

9. Uruchom teraz **HDL Dump** z parametrami "**modify_header hdd<numer dysku>: <partycja>**", czyli na moim przykładzie będzie to "**modify_header hdd1: PP.APPS**"

```
Administrator: Procesor poleceń systemu Windows
C:\test>"HDL Dump (mod by AKUHAK) v0.9.2 rev47.exe" modify_header hdd1: PP.APPS
Successfully read system.cnf
Successfully read icon.sys
Successfully read list.ico
Skipped del.ico
Skipped boot.kelf. Trying to inject boot.elf
Skipped boot.elf
Successfully read boot.kelf
Skipped boot.kirx

C:\test>
```

Teoretycznie HDL Dump sam wrzuci BOOT.KELF na wskazaną partycję, zgodnie z tym co ma wpisane do BOOT2 w pliku "system.cnf", ale na wersji którą testowałem tego nie robił. Uznałem więc że skoro i tak uruchamiam PFS Shell i tworzę nim partycję to przy okazji wrzucę też program bootowalny. Alternatywnie, równie dobrze partycję można stworzyć za pomocą uLE lub po prostu użyć którejś „PP” już istniejącej. Plik bootowalny można także przenieść via sieć lub pendrive w uLE. Jak ci jest najwygodniej.

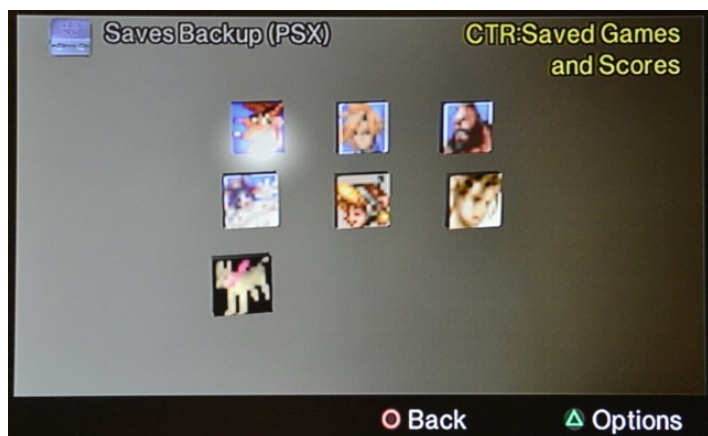


Własne ikony folderów na „partycji Common”

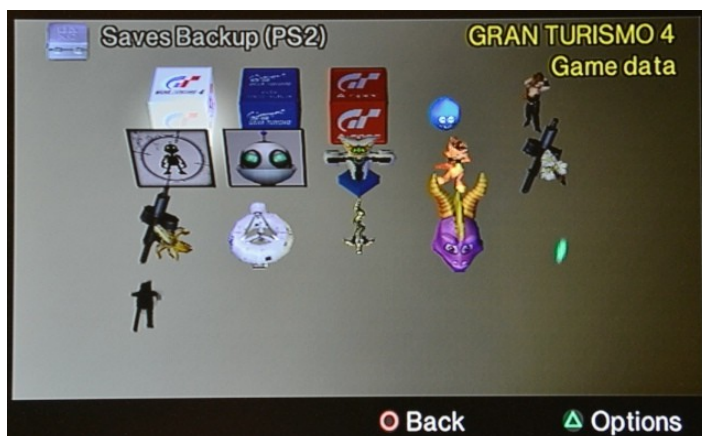
Domyślną ikoną dla folderów utworzonych na "__common" jest walizka. Z kolei domyślną dla plików i podfolderów jest biała blacha. Ale jeśli atrybut folderu nadrzędnego zmieni się na C4A7h to PS2 zacznie m.in. parsować w środku "icon.sys" i wyświetlać ikonę (czyli tak samo jak na „memorce”). Sony zarezerwowało tę funkcję dla "hdd0:/__common/Your Saves/", w którym to niektóre gry pozwalają na zapis, zupełnie tak jak na karcie pamięci. Nie ma jednak żadnych przeciwwskazań, aby zastosować ten trik dla zupełnie innych katalogów. I nie zrozum mnie źle, nie odblokuje to możliwości save'owania w takim folderze, a tym bardziej w grach które nie obsługują HDD – chodzi o to aby można było nieco spersonalizować wygląd danego zakątka na tej konkretnej partycji. Osobiście zastosowanie dla tej funkcji znalazłem w postaci osobnych folderów na save'y z PSX i PS2, coś w rodzaju magazynu skąd można kopiować je z powrotem na prawdziwe karty pamięci w razie potrzeb i oczywiście jako formę dodatkowej kopii zapasowej.

Edycję najlepiej wykonać w edytorze heksadecymalnym, ponieważ żaden znany mi program nie umożliwia zmian atrybutów wybranych abstraktów, ale musisz wtedy rozumieć jak zbudowany jest system plików PFS, bo w przeciwnym razie ryzykujesz uszkodzeniem edytowanej partycji i utratę danych.

Na szczęście jest na to trochę mozolny, ale za to całkowicie bezpieczny sposób. HDLGame Installer zapisuje ustawienia właśnie w "Your Saves", a jeśli tego folderu nie ma to go tworzy z odpowiednimi atrybutami w tablicy. Wystarczy więc w tym programie wejść do ustawień (pod przyciskiem "Select"), wybrać "OK" aby zamiany zostały zapisane na dysku i włączyć uLE. W nim zmienić nazwę folderu na pożądaną, a katalog z ustawieniami HDLGI usunąć. Operację powtarzać do skutku, czyli tyle razy ile chcesz mieć takich „specjalnych folderów”. Poniżej dwa przykłady:



hdd0:/__common/Saves Backup (PSX)/

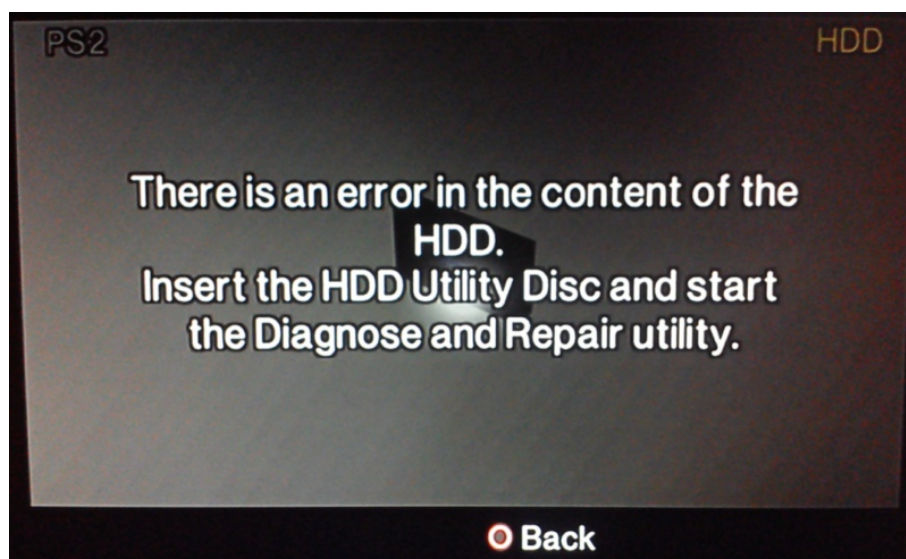


hdd0:/__common/Saves Backup (PS2)/

Wiosenne porządki

Programy homebrew po sformatowaniu dysku zakładają kilka partycji, które krótko opisałem w rozdziale poświęconym podstawom. Względem potrzeb użytkownika, niektóre są zbędne (usuwać możesz za pomocą **uLE kHn** (bo ani uLE, ani „wLE” w obecnych wersjach na to nie pozwolą), a w razie czego zawsze masz możliwość stworzyć je na nowo). A jeśli ten poradnik przyprawił cię o ból głowy ;) i zdecydowałeś się na wgranie któregoś z udostępnionych przez społeczność obrazów HDD, to dowiesz się czego możesz się z czystym sumieniem pozbyć.

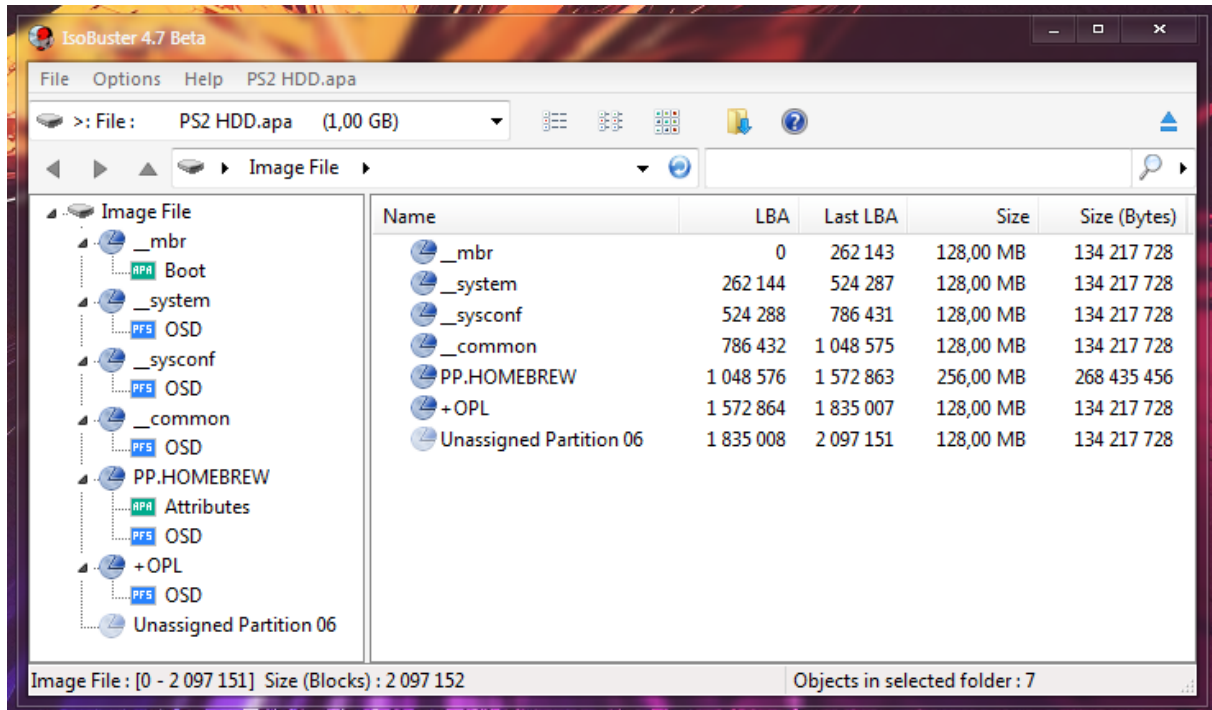
- **Jeśli** zadowala cię FMCB na karcie pamięci, i nie interesuje nic więcej niż wgrywanie gier dla PS2 na HDD to możesz usunąć wszystkie partycje z wyjątkiem "**__mbr**". Odzyskasz **całe wolne miejsce** na dysku, minus **128MiB**.
- **Jeśli** masz partycję "**__boot**", ale nie używasz trybu DEV2 modchipa to możesz ją skasować, a zwolnisz całe **128MiB**.
- **Jeśli** nie zamierzasz instalować PSBBN to partycję "**__net**" możesz usunąć. Zyskasz **128MiB**.
- **Jeśli** już masz zainstalowany PSBBN, ale go nie chcesz, a jednocześnie nie chcesz formatować nośnika, to pozbyć się "**__net**", "**__contents**", "**PP.BB**", wszystkich "**PP.<gry z kanału PSBB>**" i wszystkich "**linux.<numer>**". Uwolnisz zawrotne **10240MiB+(n*128)**. Dodatkowo możesz również skasować "**hdd0:/__system/p2lboot/**".
- **Jeśli** nie chcesz wgrywać ani FHDB, ani HDD OSD to również "**__system**", "**__sysconf**" i "**__common**" nie będą do niczego potrzebne. Zwolnisz **1792MiB**. Jeśli jednak już zainstalowałeś FHDB to po ich usunięciu konsola wpadnie w tzw. **boot loop**, dopóki nie sformatujesz dysku (lub nie nadpiszesz "**__mbr**").
- **Jeśli** nie zamierzasz emulować gier dla PSX za pomocą oficjalnego emulatora to partycja "**__common**" jest w zasadzie zbędna. Podobnie jak "**__.POPS**" i wszystkie "**PP.<tytuł gry dla PSX>**". Jeśli jednak korzystasz z HDD OSD to po usunięciu "**__common**" „browser” będzie za każdym razem wyświetlać komunikat o błędzie (można zignorować). Uwolnisz **128MiB+(n*128)**.



- **Jeśli** wrzuciłeś jakieś programy do katalogu "APPS-HDD" przed instalacją FHDB, a każdy megabajt jest dla ciebie na wagę złota to przenieś programy na inną partycję, popraw ścieżki do nich w pliku konfiguracyjnym FHDB i skasuj partycję "**PP.FHDB.APPS**". Zyskasz **128MiB**.
- **Jeśli** chcesz zdefragmentować partycje i pozbyć się pustych przestrzeni to użyj programu HDD Checker, w którym to wybierz opcję "Optimize disk".

- **Jeśli** chciałbyś zachować wszystkie systemowe partycje, ale zredukować do minimum ich rozmiar to na tę chwilę jest to niemożliwe. Możesz za to sformatować dysk twardy, usunąć wszystkie partycje (z wyjątkiem "__mbr") i utworzyć na nowo, definiując własne rozmiary. Tylko w ten sposób możliwe jest zachowanie wszystkich wymaganych przez różne oprogramowanie Sony partycji, przy jednoczesnym oszczędzeniu maksymalnie **1536MiB**.

Uwaga: minimalny rozmiar dysku twardego **nie może być mniejszy niż 16GiB**. Inaczej zarówno wszystkie odmiany uLE jak i PFS Shell utworzą za "__net" partycje o rozmiarze mniejszym niż zgodny ze specyfikacją APA (nawet i 8MiB! zależne jest to od maksymalnej pojemności HDD), ale nie będzie możliwe utworzenie żadnych nowych partycji, dlatego też ta wielkość jest na chwilę obecną graniczna.

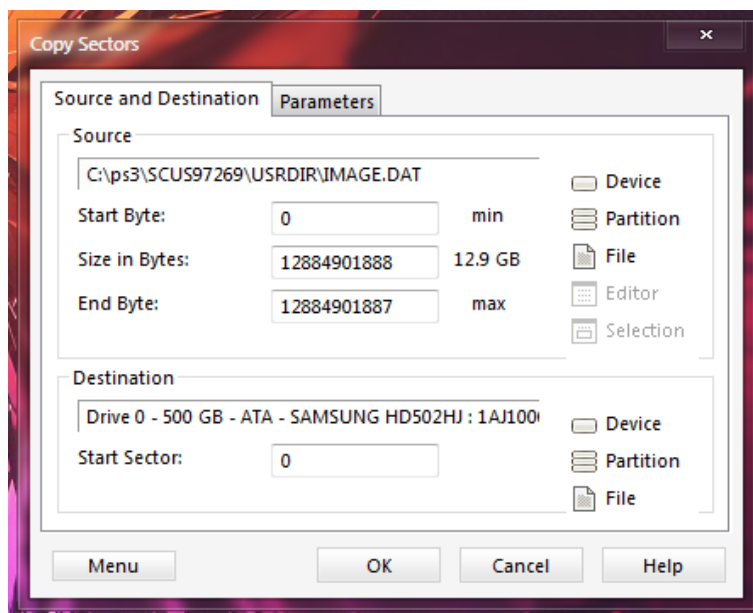


Przykład jak to rozmiarowo wygląda u mnie.

IMAGE.DAT

Na wstecznie kompatybilnych **PlayStation 3**, które posiadały na pokładzie przylutowane Emotion Engine i Graphics Synthesizer (lub tylko jeden z nich), Sony umożliwiło **instalację kilku wybranych gier** na wirtualnym dysku twardego PS2, jeśli użytkownik zainstalował wcześniej z PlayStation Store paczkę **PS2 System Data** zawierającą szablon wykorzystywany przez te gry do instalacji swoich danych.

Wspomniany wirtualny dysk to plik "IMAGE.DAT", który jest posektorowym obrazem dysku twardego PS2. Można go po prostu wgrać na prawdziwy HDD (np. za pomocą **DMDE** lub **HDD Raw Copy Tool**) i prawdopodobnie po zainstalowaniu HDD OSD, uruchomić grę z partycji wyświetlanej w „Browserze”.



Plik znajduje się w "[dev_hdd0/game/<GameID>/USRDIR/IMAGE.DAT](#)" i można go przesłać na PC za pomocą FTP (lub jeśli jest mniejszy niż 4GiB to na pendrive za pomocą menedżera plików np. wbudowanego w **multiMAN**).

Zagadnienia, które pozostały do omówienia:

- Preparowanie partycji z Linuksem Black Rhino w przestrzeni APA
- Instalacja PlayStation BroadBand Navigator

Podziękowania dla:

- **Jimmikaelkael, Neme**, i pozostałym zaangażowanym duszom w projekt FMCB v1.8x
- **SP193** za tytaniczną pracę nad FMCB/FHDB 1.9xx, uwagi i **dokumentacje**
- **krHACKen**-owi za SUDC, POPStarter, SCEDoormat, dumpy i wskazówki
- rządowi dobrych dusz, stale pracujących w pocie czoła nad OPL
- **ffgriever**-owi za ESR
- **Shaolin Assassin**-owi za świetną **dokumentację** dotyczącą POPS i POPStartera
- **Zabuza** za zdjęcie Network Adaptor
- **KaiQ** za podarowanie mi Network Adaptor
- **Haker120** (aka **A youkai of love**) za dumpy, dodatkowe uwagi dotyczące poradnika i pomoc w testach
- **Sandungas** za próbki IMAGE.DAT
- **Darkshadow_** za wyjaśnienie algorytmu wyliczającego sumę kontrolną nagłówka partycji
- **Pinky1** za napisanie programu poprawiającego sumy kontrolne nagłówków partycji