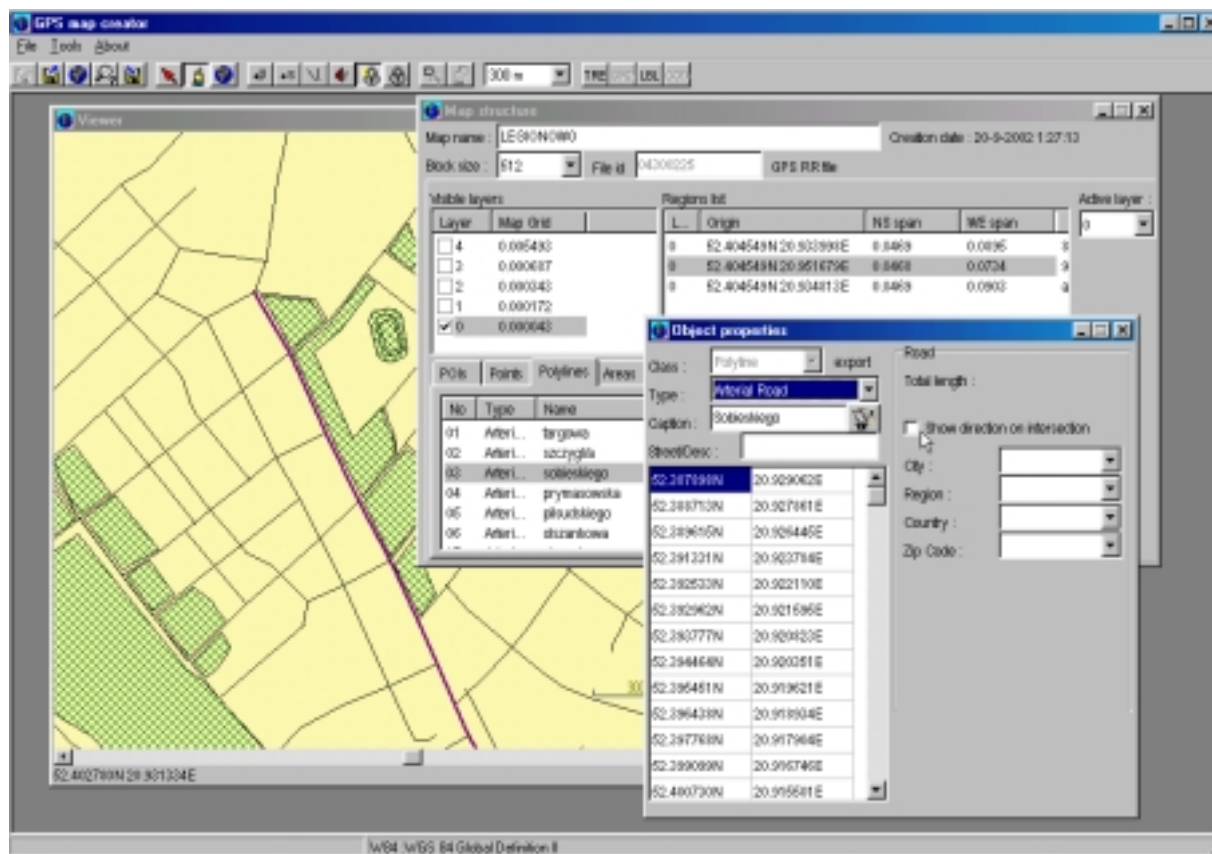


Zr-UMP To Sam Czyli Jak zrobić mapę ładowalną do GPS Garmin

Dokumentacja do oprogramowania GPSMAPPER



Wprowadzenie

Ideą UMP jest tworzenie darmowych map wektorowych w formacie Garmin. Informacje i dane zebrane przez osoby tworzące UMP są ogólnodostępne dla zastosowań niekomercyjnych, w szczególności można z nich (w całości lub fragmentów) kompilować własne mapy w formacie IMG.

UMP funkcjonuje jako projekt grupy zapaleńców, którzy udostępniają wyniki własnej pracy w postaci źródłowej (pliki TXT i/lub PLT) akceptowalnej przez program kompilujący – cGpsMapperper – dostępny na stronie GPS Mappera (gps.chrisb.org).

Każdy użytkownik odbiornika GPS może pobrać wyniki pracy autorów UMP zarówno w postaci gotowego pliku IMG jak i wersji jawnej, może dowolnie modyfikować jej zawartość dodając, zmieniając lub usuwając dane, których nie potrzebuje.

Autorzy plików źródłowych zastrzegają jednak, że UMP nie może być przedmiotem transakcji handlowych i musi być udostępniane za darmo. UMP nie może również służyć celom zarobkowym w sposób pośredni, np. poprzez wykorzystywanie map przez firmy transportowe lub jako produkt bonusowy przy sprzedaży odbiorników GPS. Szczególnie w tym drugim przypadku autorzy UMP wymagają podawania jasnej informacji, że UMP jest projektem darmowym i niezwiązanym z firmą Garmin ani żadnym z jej przedstawicieli.

Wszystkie pytania nt. UMP można zadawać na liście usenet.pl.rec.gps, pod adresem <http://www.emap.prv.pl/> dostępny również jest FAQ GPS oraz zbiór wykonanych w ramach UMP map – gotowych do wgrania i służących jako przykłady.

Autorzy UMP nie ponoszą odpowiedzialności za jakiegokolwiek szkody pośrednie i bezpośrednie wynikłe z używania danych i map publikowanych w ramach UMP, w szczególności związane z niedokładnością danych lub uszkodzeniami odbiornika. Autorzy dokumentacji, oprogramowania i osoby zaangażowane w dystrybucję plików UMP nie biorą odpowiedzialności za dane nadesłane od innych autorów plików IMG i TXT, w szczególności za legalność ich pochodzenia.



Wszystkie uwagi dotyczące danych UMP należy kierować na listę usenet.pl.rec.gps lub bezpośrednio do autora konkretnych danych (ewentualnie za pośrednictwem osoby przygotowującej dystrybucję). Taka procedura zapewni integralność danych dzięki wprowadzaniu poprawek przez jedną osobę.

Zarówno ostatnia edycja UMP (3) jak i niniejsza dokumentacja opiera się o funkcjonalność oprogramowania GpsMapperper w wersji 0.45 lub wyższej. Polecamy stosowanie ostatniej, najwyższej dostępnej wersji oprogramowania, ponieważ ulega ono ciągłemu rozwojowi i usuwane są błędy zauważone przez użytkowników. Opisywane postępowanie oraz uzyskane mapy były testowane i są używane na różnych typach odbiorników z mapami.



Na liście dyskusyjnej usenet.pl.rec.gps podawane są także informacje o

terminach i miejscach spotkań użytkowników GPS i UMP. Zapraszamy do udziału w dyskusjach na liście i podczas spotkań.

UWAGA

Zwróć uwagę na możliwość wystąpienia nieścisłości w rozumieniu słowa TYP. Tak się składa, że program kompilujący mapy z formatu TXT na IMG używa dwóch parametrów do określenia wyglądu obiektu na mapie – Type i RgnType. Określenie TYP będziemy stosować do parametru RgnType, parametr Type będziemy określać słowem RODZAJ.

Oprogramowanie narzędziowe

Będziemy korzystać z pakietu oprogramowania GpsMapperper. Pakiet (w wersji 0.045) składa się z trzech głównych aplikacji:

- GpsMapperper
- cGpsMapperper
- sendmap

Szczegóły dotyczące wykorzystania powyższych aplikacji będą podawane w dalszej części instrukcji, zaznaczyć tylko należy, że całość tej dokumentacji, zawarte w niej przykłady i – w szczególności - UMP dotyczy pakietu ogólnodostępnego na stronie <http://gps.chrisb.org/>, czyli wersji SHAREWARE.

Proszę zwrócić uwagę, że do poprawnej pracy oprogramowanie wymaga kilku bibliotek – plików DLL, które także są do pobrania ze strony GpsMapperpera. Tak też umieszczono instrukcję, gdzie należy umieścić te pliki.

Pełna, komercyjna wersja pakietu umożliwi znacznie wygodniejszą pracę, w tym indeksowanie POI w poziomym pliku tekstowego (my musimy sięgać do GpsMapperper) oraz import danych z plików SHP.

Do tworzenia komercyjnych map należy stosować pakiet w pełnej wersji – wszelkie informacje są dostępne bezpośrednio u autora projektu pod adresem gps@chrisb.org.

Dodatkowym oprogramowaniem narzędziowym jest **Ozi Explorer**. Oprogramowanie to pozwala na wygodne tworzenie linii wyznaczających późniejsze obszary, linie na mapie GPS oraz przetwarzanie Waypointów.

Dane zapisane w formacie Ozi Explorer mogą być bezpośrednio importowane do plików IMG otwartych do edycji w GpSmapper oraz przetwarzane przez cGpsMapper z formatu tekstowego do formatu IMG akceptowanego przez odbiorniki Garmin.

Uwaga do wersji dokumentacji 1.04 i nowszych

Od wersji 0.045 oprogramowanie narzędziowe nazywa się GpsMapper i cGpsMapper. W wersjach poprzednich funkcjonowały pod nazwą GpsMap i cGpsMap. W starszych wersjach dokumentacji mogą występować obie nazwy lub nazwa starsza, jednak generalne zasady korzystania nie uległy zmianie.

Szybki start

Jeżeli interesuje Cię wykonanie mapy ze szczytami górskimi (lub innymi punktami typu POI¹ – bary, stacje CPN, schroniska, bankomaty), ładowalnego do odbiorników przyjmujących pliki z punktami, choć nie wyświetlającymi map, po lekturze rozdziału **Szybki Start** koniecznie zajrzyj do rozdziału **Mapa Gór**.

Nie wnikając zbyt głęboko w strukturę mapy Garmina musimy przynajmniej wiedzieć, że gotowa mapa składa się w warstw o różnej dokładności. Warstwy możemy wyobrazić sobie jak kalka w kratkę, im warstwa dokładniejsza, tym kratka jest oparta o gęściej rozlokowane punkty – węzły. Linia (obiekt) umieszczona w warstwie 0 (najdokładniejszej) i powielana później na kolejne warstwy będzie pozbawiana części punktów. O tym, kiedy (przy jakim powiększeniu) zobaczymy na mapie (ekranie GPS) dany element decyduje jego przydział do warstw. Dokładniejsze rozważania i przykład zawarty jest w Załączniku 4. Osoby zainteresowane ingerencją w widzialność i rozmieszczenie obiektów w warstwach odsyłamy do lektury Załączników i części dla zaawansowanych.

Na początek możesz przyjąć uproszczenie, że wszystkie elementy występują na warstwie 0 i są powielane na warstwy wyższe w zależności od ważności – np. drogi główne są na wszystkich warstwach, a ścieżki leśne tylko na tych, które widać po powiększeniu mapy. To Ty decydujesz o powielaniu elementów wpisując wartość parametru Levels=. Wracając do przykładu z drogami – mogą mieć parametr Levels=4, a ścieżki leśne Levels=2. Wrócimy do tego w dalszej części instrukcji.

1. Jeżeli nie chcesz samodzielnie tworzyć nowych elementów mapy (albo po prostu nie masz Oziego), a jedynie wykorzystać istniejące, opublikowane w UMP, możesz opuścić lekturę punktów 2, 3, 4, 6 i 7.
2. Uruchom Ozi Explorer ze skalibrowaną mapą. Jeżeli nie posiadasz żadnej skalibrowanej mapy lub szukasz mapy jakiegoś regionu, dobrym pomysłem jest zapytanie na liście pl.rec.gps lub wizyta na stronie Oziego <http://www.ozieplorer.com/>.
3. Na skalibrowanej mapie nanieś ślady (Track) i punkty (Waypoints). Możesz także odczytać zapisane wcześniej własne ślady i punkty, lub ślady i punkty pobrane z grupowego serwera FTP (dane dostępne do serwera pojawiają się regularnie jako porada na grupie pl.rec.gps, są w FAQ, można także zadać pytanie na wspomnianą grupę).

Ślady będą służyły do tworzenia obiektów typu (RgnType) linia (0x40 – dróg, strumieni, linii kolejowych) i obszar (0x80 – lasy, parki, stawy, rzeki, obiekty inne – stadiony itp.). Punkty będą służyły do tworzenia obiektów POI (0x10) wyszukiwanych w odbiornikach przez Find i obiektów punktowych (0x20 – miasta, wieś jako znacznik).



W odbiornikach umożliwiających zapis POI ale nie posiadających możliwości wyświetlania map, możemy zapisać pliki generowane w ramach UMP pozbawione obiektów typów 0x20, 0x40 i 0x80. Dla takich odbiorników tworzymy więc jedynie pliki WPT, np. apteki.wpt, szczyty.wpt, parkingi.wpt i co tam nam jeszcze przyjdzie do głowy :).

¹ POI – Point Of Interest – obiekty mapy zaznaczone jako punkt. POI można wyszukiwać korzystając z funkcji FIND odbiornika, co pozwala na przykład na odnajdywanie POI najbliższego naszej pozycji i włączenie nawigacji do niego. Dokładniejsze informacje nt. funkcjonalności odbiorników i POI znajdziesz w instrukcji użytkownika odbiornika Garmin.

Zwróć uwagę, że w Ozim (do wersji 3.90.4g) manualnie tworzysz zawsze track nr 1. Jeżeli chcesz zaznaczyć więcej linii (wiele ulic, parki itp.) musisz nanosić je kolejno, zapisywać w osobne pliki PLT. Od tej reguły istnieje wyjątek, ale zostanie on opisany w dalszej części instrukcji (wymaga zastosowania dodatkowego pliku txt, np. można zapisać wszystkie ulice jednego typu, powiedzmy miejskie, w jednym pliku PLT jako jeden track o wielu sekcjach – miejskie.plt, a dodatkowo utworzyć plik miejskie.plt.txt zawierający listę nazw w kolejności segmentów w pliku plt). Tworzenie śladów wielosegmentowych składających się z linii przerywanych oraz inne specyficzne operacje na śladach opisuje help do Ozi Explorera. W naszych zastosowaniach wystarczy wiedzieć, że przy rysowaniu śladu, postawienie każdego kolejnego punktu powoduje dołączenie go niego i przedłużenie linii o kolejny odcinek. Aby dodać punkt ale pozostawić odstęp, należy postawić punkt trzymając wciśnięty klawisz ALT. Dzięki temu możemy tworzyć ślady rozgałęzione – rysujemy pierwszą gałąź, wracamy do punktu rozgałęzienia, wciskamy i przytrzymujemy ALT, stawiamy punkt w miejscu rozgałęzienia, puszczaemy ALT i rysujemy kolejną gałąź. Postawienie punktu poza „starym” śladem pozwala na narysowanie kolejnego fragmentu tego samego śladu jako oddzielnej linii.

Pliki PLT warto nazywać nazwami obiektów np. skwer_ratajskiego.plt, miedziana.plt. Zwróć uwagę, aby nie używać polskich znaków – na ekranie odbiornika będą wyświetlane niepoprawnie, mogą uniemożliwić wyszukiwanie oraz kompilację. Przyjmij zasadę usuwania ogonków ze wszystkich napisów, które tworzysz.

Pliki WPT mogą zawierać więcej niż jeden obiekt, ale zachowuj regułę – jeden plik WPT – jeden rodzaj (Type) obiektów. Dzięki temu później te same obiekty będą miały tę samą ikonę w odbiorniku i będzie łatwo nimi zarządzać.

4. Po zakończeniu prac jesteś w posiadaniu zestawu plików PLT i WPT, np. takich

```
wisniowa.plt
pl_sloneczny.plt
egipska.plt
droga_619.plt
apteki.wpt
bankomaty.wpt
park001.plt
miejscowosci.wpt
```

5. Tworzymy plik wyjściowy do kompilacji, nazywając go np. moja_mapa.txt. Plik składa się z kilku sekcji identyfikowanych przez nagłówek w nawiasie kwadratowym i analogicznym zakończeniem (nazwa sekcji poprzedzona słowem END-), na początek wystarczy, że wykorzystasz sekcje IMG ID i FILE.

Jeżeli chcesz dodawać komentarze – nie ma problemu – linie rozpoczynające się znakiem ; (średnik) są ignorowane przez program kompilujący.

Przykładowy plik nagłówek dostarczany jest z każdą edycją UMP (dostępne na serwerze FTP) pod nazwą headline.bak. Dla wersji UMP-3 z 19.09.2002 wyglądał następująco:

```
[IMG ID]
ID=04300001
Name=warszawa
AlignMethod=0.0001
```



Początek sekcji IMG ID, która zawiera definicje nazwy pliku wynikowego, nazwę mapy i jej parametry. Dokładny opis parametrów zawiera Załącznik 3 oraz witryna <http://gps.chrisb.org/>

```
TreMargin=0.02
Datum=W84
WorldMap=N
TreSize=2048
LevelLimit=250
LevelFill=50
RgnLimit=500
Levels=5
Level0=22
Level1=21
Level2=19
Level3=17
Level4=15
Zoom0=0
Zoom1=1
Zoom2=2
Zoom3=3
Zoom4=4
[END-IMG ID]
```



Proponujemy nie modyfikować zawartości sekcji IMG, poza liniami opisanymi poniżej. Aktualny nagłówek może wyglądać inaczej, program cGpsMapper ulega ciągłym zmianom, które wpływają na zawartość sekcji IMG. Nawet powyższy przykład zawiera parametry ignorowane przez cGpsMapper od wersji 0.045, a najbliższa edycja UMP będzie dystrybuowana ze zmienionym plikiem *headline.bak* - dlatego na potrzeby Szybkiego Startu będziemy posługiwać się następującym nagłówkiem, zgodnym z ostatnią wersją pakietu:

```
[IMG ID]
ID=04300225
Name=moja mapa
Datum=W84
TreSize=2048
Levels=5
Level0=22
Level1=21
Level2=19
Level3=17
Level4=15
Zoom0=0
Zoom1=1
Zoom2=2
Zoom3=3
Zoom4=4
[END-IMG ID]
```

To co zmieniamy, to:

- a) ID=xxxxxxxx – zastępując osiem iksów ośmioma cyferkami. Po kompilacji uzyskamy plik IMG o nazwie zgodnej z tym numerem. Dla potrzeb tego opisu wpisujemy 04300225, ale możesz wybrać inny numer, o ile jest inny od dotychczas posiadanych. **UWAGA.** ID w postaci ciągu cyfr i liter nie spowoduje błędów przy kompilacji, ale ID powinno być liczbą. Wprowadzenie wartości alfanumerycznej (np., 0430WAWA) może spowodować nieprzewidywalne zachowanie odbiornika i dlatego nie należy używać liter w ID.
- b) Name=tu_wpisujemy_swoją_nazwę – ta nazwa będzie wyświetlana w odbiorniku w menu MapInfo

Dla zaspokojenia ciekawości – zmiany parametrów Level i Zoom wpływają na widzialność kolejnych szczegółów przy różnych skalach. Levels dotyczy widoku w odbiorniku, Zoom w programie MapSource. Szczegółowy opis tego parametru zawiera Załącznik 3.

6. Dodajemy sekcję FILE (jedną, bo będzie zawierać odwołania do naszej radosnej twórczości)

```
[FILE]
name=!widok_og.txt
name=!ulice.txt
name=!obrysy.txt
name=!zielone.txt
[END-FILE]
```

Jak widać, przyjąłem regułę nazywania plików zbiorczych od znaku wykrzyknika, aby można było łatwo znaleźć je w gąszczu innych, szczegółowych plików TXT. Oczywiście można stosować dowolne nazwy.

7. Musimy teraz utworzyć pliki wskazane w sekcji FILE. Każdy z nich będzie zawierał wywołania konkretnych plików PLT i WPT, definicje typów oraz położenia obiektów na warstwach.

Tworzymy np. plik !ulice.txt w którym umieścimy wywołania naszych uliczek – tych mniejszych i tych większych.

```
[PLT]
Type=0x03
RgnType=0x40
LABEL=~[0x2d]619
DIRINDICATOR=0
Levels=3
FILE0=droga_619.plt
[END-PLT]
```

- definicja typu podpiętego pliku
- określamy rodzaj obiektu
- określamy typ obiektu
- to się pojawi przy obiekcie na ekranie GPS
- to pomijamy²
- ten obiekt znajdzie się na warstwach 0, 1, 2, 3
- a tak nazywa się plik z danymi obiektu
- koniec definicji

```
[PLT]
Type=0x06
RgnType=0x40
LABEL=wisniowa
DIRINDICATOR=0
Levels=3
FILE0=wisniowa.plt
[END-PLT]
```

```
[PLT]
Type=0x06
RgnType=0x40
LABEL=pl. sloneczny
DIRINDICATOR=0
Levels=3
```

² DIRINDICATOR=0 - jeżeli ustawione na 1, po najechnaniu wskaźnikiem na skrzyżowanie, do nazwy automatycznie będzie dodawany kierunek świata, w którym prowadzi dana droga

```
FILE0=pl_sloneczny.plt
[END-PLT]
```

itd.

Nieco inaczej wygląda definicja POI i punktów definiowanych w plikach WPT. Sekcje WPT można wstawić w każdy z plików !....txt, np. w pliku !ulice.txt możemy umieścić oprócz odwołania do plików PLT także definicje POI.

```
;Miejscowości
[WPT]
Type=0x0b
RgnType=0x20
Levels=4
FILE0=miasteczka.wpt
[END-WPT]
```

W tym przykładzie plik miasteczka.wpt zawiera zbiór miejscowości, które będą przedstawione na mapie w odbiorniku jako obiekt rodzaju 0x0b w sześciu warstwach.

```
; linie i stacje PKP
[WPT]
Type=0x2f08
RgnType=0x10
Levels=4
FILE0=stacje_pkp.wpt
[END-WPT]
```

Tu mamy plik zawierający obiekty rodzaju 0x2f08, ale już typu 0x10 – **POI**. Trafia w pięć warstw: 0, 1, 2, 3 i 4.

```
[PLT]
Type=0x14
RgnType=0x40
Label=~[0x1f]
DirIndicator=1
Levels=4
FILE0=g_kolej_01.plt
[END-PLT]
```

Analogicznie możemy dodać plik z listą szczytów górskich (jeżeli w definicji użyjemy RgnType=0x10 staną się one POI), restauracji, pizzerii, bankomatów itp. Dokładniejszy opis postępowania przy tworzeniu map z POI zawiera rozdział **Mapa Gór lub samych POI** w dalszej części dokumentacji.

Alternatywnym sposobem włączenia punktów do naszej mapy jest wywołanie plików dla kolejnych warstw w jednej sekcji WPT

```
[WPT]
Type=0x0b
RgnType=0x20
FILE0=miasteczka_warstwy0.wpt
FILE1=miasteczka_warstwy1.wpt
FILE2=miasteczka_warstwy3.wpt
FILE3=miasteczka_najwieksze.wpt
[END-WPT]
```

Dzięki temu możemy rozdzielić posiadane punkty na warstwy widoczne w różnych skalach – File0 dla warstwy najdokładniejszej (0), File1 dla warstwy 1 itd. Rozwiązanie poprzednie (osobne sekcje WPT dla każdego pliku) pozwala natomiast na uzyskanie innej ikony (przez zmianę parametru Type, czyli rodzaju obiektu) dla obiektów umieszczanych w różnych warstwach. Np. w poniższym przykładzie:

```
[WPT]
Type=0x0c
RgnType=0x20
Levels=3
FILE0=miasteczka_warstwy0.wpt
[END-WPT]
```

```
[WPT]
Type=0x0a
RgnType=0x20
Levels=4
FILE0=miasteczka_warstwy1.wpt
[END-WPT]
```

uzyskujemy efekt widzialności obiektów z pliku miasteczka_warstwy1.wpt jako SmallCity od warstwy 0 (FILE0) do warstwy 4 – bo Levels=4 – (czyli w sumie na pięciu), a z pliku miasteczka_warstwy0.wpt jako SmallTown i tylko na czterech warstwach (warstwy od **zero** do **trzy** – File0, Levels=3).

Wracając do opisu kolejnych parametrów jeszcze kilka słów wyjaśnienia:

Type i RgnType - rodzaj i typ obiektu. Wszystkie typy i zawierające się w nich rodzaje zawiera tabela w Załączniku 1. Np. 0x40 oznacza linię, 0x03 drogę krajową, 0x06 drogę o niższej ważności.

Levels – numer ostatniej warstwy, do której zostanie powielony ten obiekt. Levels=3 dla danych wywołanych FILE0=... oznacza, że obiekt będzie widoczny na czterech warstwach (0, 1, 2, 3). Levels=2 oznacza, że pojawi się dopiero przy kolejnym powiększeniu – nie będzie widoczny na warstwie numer 3.

WAŻNE – należy unikać polskich znaków zarówno w nazwach plików, jak i w etykietach (Label=). Należy używać **TYLKO** małych liter w nazwach obiektów – odbiornik będzie je konwertował na wielkie litery.

Reguły wyboru Type i RgnType stosowane dotychczas w UMP zawiera Załącznik 2

- Możemy oczywiście dopisać jeszcze zawartość plików TXT dostarczanych w kolejnych edycjach UMP. Wystarczy przekopiować żądane fragmenty na koniec naszego pliku, mogłby więc wyglądać on następująco:

```
[IMG ID]
ID=04300225
Name=moja mapa
Datum=W84
TreSize=2048
Levels=5
Level0=22
Level1=21
Level2=19
Level3=17
Level4=15
Zoom0=0
Zoom1=1
Zoom2=2
```

```
Zoom3=3
Zoom4=4
[END-IMG ID]
```

```
[FILE]
name=!widok_og.txt
name=!ulice.txt
name=!obrysy.txt
name=!zielone.txt
[END-FILE]
```

```
; ponizej fragment pliku PIA.ZZ.POI.TXT z edycji z 19.09.2002
[RGN10]
Type=0x0057
Label=zle psy
Origin0=(52.0270657539,21.0151505470)
[END-RGN10]
;-----stacje benzynowe-----
```

```
[RGN10]
Type=0x2f01
Label=orlen
Levels=2
Origin2=(52.0922112465,21.1194348335)
[END-RGN10]
```

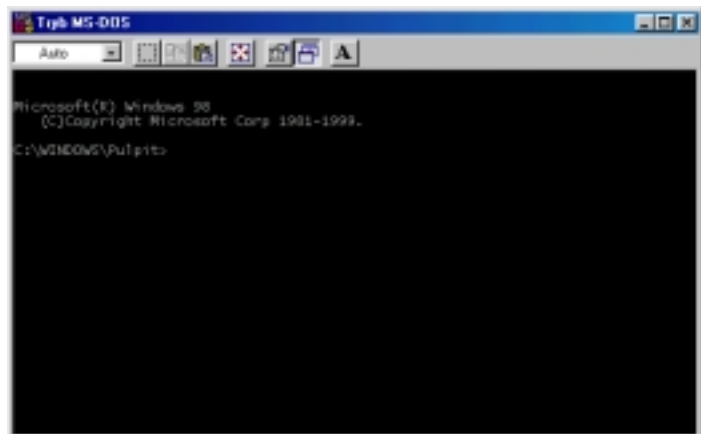
```
[RGN10]
Type=0x2f01
Label=orlen
Levels=2
Origin2=(52.0269799232,21.1173748970)
[END-RGN10]
```

Tu kolejna uwaga: dla punktów POI parametr zawierający nazwę pliku nosi nazwę Origin. Dla linii i obszarów – File. Od wersji 0.045 możemy stosować dla wszystkich sekcji parametr o nazwie File.

9. Ten plik zapisujemy, już niedaleko do końca! Wystarczy teraz skompilować go, obserwując czy nie wystąpiły błędy. Program do kompilowania map pobierzesz ze strony GpsMapperpera – <http://gps.chrisb.org/>

W katalogu z zainstalowanym programem utwórz podkatalog, nazywając go po swojemu, np. MOJA_MAPA. Skopiuj do tego katalogu wszystkie pliki PLT, WPT i plik moja_mapa.txt.

Uruchamiamy command prompt (DOSowe okno w którym można wydawać komendy), to okno może wyglądać tak:



Jeżeli nie możesz, nie chcesz albo po prostu nie lubisz okien DOS i trybu wsadowego, polecam lekturę Załącznika 5, który zawiera opis postępowania służącego stworzeniu

środowiska do kompilacji bezpośrednio z poziomu pulpitu i przez przeciąganie ikon (drag and drop) – po lekturze możesz przejść od razu do p. 10.

Przechodzimy do swojego katalogu, służy do tego polecenie cd.

Wpisujemy więc:

cd "\soft\gps\GpsMapper\moja_mapa"

i naciskamy Enter

```

Microsoft(R) Windows 98
(C)Copyright Microsoft Corp 1981-1999.

C:\WINDOWS\Pulpit>cd \soft\gps\gpsmap\moja_mapa_
  
```

Uruchamiamy kompilację naszego pliku moja_mapa.txt komendą

..\cGpsMapperper ac moja_mapa.txt

Obserwujemy ekran, na którym będą pojawiać się informacje o stanie kompilacji. Na początek nie jest to może najistotniejsze, więc tylko cierpliwie czekamy...

Przykładowy wygląd ekranu na początku procesu znajduje się obok.

```

Microsoft(R) Windows 98
(C)Copyright Microsoft Corp 1981-1999.

C:\WINDOWS\Pulpit>cd \soft\gps\gpsmap\moja_mapa

C:\Soft\gps\GPSmap\moja_mapa>..\cGpsMapperper ac moja_mapa.txt
GPSmap command line utility version: 0045 Sep 9 2002
Shareware edition
GPSmap import library version 00061
Importing data
Current import datum is: W84 :WGS 84 Global Definition II
!WIDOK_0G.TXT - processing included file
Reading: MIASTECZKA.WPT ...6 waypoints
Reading: STACJE_
  
```

Czekamy, czekamy i, jeżeli nie strzeliliśmy jakiegoś byka, zobaczymy coś takiego:

```

*****END*****
2 regions to import... done
2 regions to import... done

Size optimization in progress
Done!
File :04300225.img is written

C:\Soft\gps\GPSmap\moja_mapa>
  
```

10. Wynikiem poprawnej kompilacji będzie plik IMG zawierający mapę (lub same POI, jeżeli tak zdecydowałeś), którą możesz załadować do odbiornika. W powyższym przykładzie (patrz ilustracja) uzyskaliśmy plik 04300225.img.

11. Są trzy drogi aby plik IMG znalazł się w odbiorniku:

- a. Mapsource nieelegancko
- b. MapSource elegancko
- c. SendMap

Ponieważ przerabiamy wersję „szybki start” wybierzemy wariant c. Zainteresowanych wariantami a i b odsyłam do lektury dalszych rozdziałów.

SendMap pobierzemy ze strony <http://gps.chrisb.org/>.

Podpinamy odbiornik kabelkiem do portu szeregowego (COM). Wracamy do okna poleceń DOS i wpisujemy kolejną komendę (jeżeli COM1 jest zajęty przez inne urządzenie, w linii polecenia wpisujemy numer portu do którego przypieliśmy nasz odbiornik – com2, com3 itp.)

sendmap com1 043000225.img

Obserwujemy ekran, całość wygląda np. tak:

```
C:\soft\GPS\GpsMapper\moja_mapa\>sendmap.exe com1 043000225.img
GpsMapper communication library version 00031
Got ID
Product_ID=169
SWVersion=229
VersionString=eTrex Vista Beta Software 2.29c
Got protocol array
Map upload capable
Unlock map capable
Got serial number
SerialNumber=2865291853
Got valid speeds
Valid speeds:
9472
14208
18944
28417
37889
56834
113668
Got Flash Map description
Bytes to send : 6956
Available memory: 25165824
Got change speed
Speed change: 113668
Got Memory clear
86 % (*)
C:\>
```

(*)mimo tego że nie pojawia się tu 100% to wszystko jest OK i działa - przynajmniej na Viscie i eMapie

Mapa Gór lub samych POI

Przyjmujemy założenie, że przebrnąłeś przez poprzednie przykłady i – z grubsza – rozumiesz istotę tworzenia własnych map. Ponieważ trening czyni mistrza, poniższy przykład stworzenia własnej mapy zawierającej szczyty (w ogólności tylko obiekty POI, które możesz załadować do odbiornika bez możliwości wyświetlania pełnej mapy) będzie rozszerzony o komentarze, które mogą Ci się wydać niepotrzebne (no jasne, skoro to czytasz, to już nie jesteś lamerem ☺)

Przyjmujemy następujący rozkład warstw:

```
>; 22 8.58306884765625E-05 6m 500m
>; 20 0.00034332275390625 25m 2km
>; 19 0.0006866455078125 50m 3km
>; 18 0.001373291015625 100m 5 - 8km
```

Ponieważ znasz ogólne zasady możemy przejść do tworzenia pliku TXT. Linie bez komentarza pozostaw bez zmian, ta wiedza przyjdzie z czasem ;)

```
[IMG ID]
ID=00000712 - ten numer dobieramy sami pamiętając żeby był unikalny w stosunku do innych map, które zechcesz ładować
Name=Nowy Sacz - nazwa mapy
PreProcess=F
Datum=W84
TreSize=1000
Levels=4 - cztery warstwy (patrz wyżej)
Level0=22 - pamiętasz co było wyżej w kolumnie 'Grid'?
Level1=20
Level2=19
Level3=18
Zoom0=1 - tylko dla MapSource, po prostu wpisz tutaj kolejne wartości od 1 i tyle
Zoom1=2
Zoom2=3
Zoom3=4 - zwróć uwagę, żeby Zoom-ów było tyle samo co Level-i
[END-IMG ID]
```

Teraz dorabiamy sekcję WPT, w której umieścimy nasze szczyty (bary, bankomaty itp.)

```
[WPT]
RgnType=0x10 - 0x10 oznacza punkty
Type=0x6616 - to jest typ oznaczający szczyty (patrz Załącznik 1)
FILE0=gory.wpt - dane dla warstwy 0
FILE1=gory.wpt - dla warstwy 1...
FILE2=gory.wpt
FILE3=gory.wpt
[END-WPT]
```

albo – alternatywnie (pamiętasz z Szybkiego Startu?)

```
[WPT]
RgnType=0x10 - 0x10 oznacza punkty
Type=0x6616 - to jest typ oznaczający szczyty (patrz Załącznik 1)
Levels=3 - obiekty będą na warstwach do numeru 3 (czterech, bo od 0 do 3)...
FILE0=gory.wpt - ... począwszy od warstwy 0
[END-WPT]
```

Dalej już umiemy – okno DOS (command prompt), kompilacja itd., zgodnie z rozdziałem Szybki Start, p. 9

Jak dotąd nie było wiele nowego, ale doszliśmy do miejsca, w którym dociekliwy turysta zapyta, co z wysokościami punktów... I oto co:

- wszystkie wysokości muszą (MUSZA) być wyrażone w stopach. Oczywiście możemy je przechowywać w metrach, ale trzeba uwzględnić to pragnienie w sekcji IMG:

```
[IMG]
.
.
Elevation=m
.
.
[END-IMG]
```

- wysokość mogą przechowywać tylko niektóre rodzaje obiektów. Dodatkowo, wysokość należy wyraźnie wydzielić w nazwie. Oczywiście możemy dla parametru Label wpisać tylko wysokość, ale jeżeli chcemy też wyświetlać nazwę własną, wykorzystujemy separator, podobnie jak wydzielamy numer drogi. Dla szczytów będzie to ~[0x1f] (który jednocześnie ukrywa drugą część nazwy i pozwala na konwersję wysokości stopy-metry), np.

```
Label=rysy~[0x1f]2499
```

W przypadku plików WPT nazwa jest zawarta w opisie punktu, można ją edytować w Ozim lub z poziomu zwykłego edytora.

- dodatkowo konwersji podlegają tylko następujące rodzaje i typy:

```
RGN10 : rodzaje pomiędzy 0x62 a 0x66
RGN20 : rodzaje pomiędzy 0x62 a 0x63
RGN40 : rodzaje pomiędzy 0x20 a 0x25
```

- różne rodzaje i typy przechowują wysokość z różną dokładnością

```
Dla RGN20:
Rodzaj: 0x62 typ: 0x20 - pokazuje wysokość zaokrąglona do pełnych metrów (stóp)
Rodzaj: 0x63 typ: 0x20 - pokazuje wysokość z jedną cyfrą po przecinku!!
to samo dla RGN10:
0x6201 0x10 - pokazuje wysokość zaokrąglona do pełnych metrów (stóp)
0x6301 0x10 - pokazuje wysokość z jedną cyfrą po przecinku
```

Wiedząc to wszystko tworzymy docelowy plik TXT i go kompilujemy. Następnie udostępniamy innym ;) i ładujemy do odbiornika.

Plik WPT z punktami może wyglądać np. tak:

```
24,025 , 49.472730, 20.539380,37380.80635, 70, 1, 3,0, 65535,skalka-[0x1f]1163 , 0, 0, 0, 3793,6, 0,17
27,27A , 49.496650, 20.264187,37380.81491, 75, 1, 3,0, 65535,runek-[0x1f]1005 , 0, 0, 0, -777,6, 0,17
36,NIEMCOWA , 49.445510, 20.646820,37380.80638, 70, 1, 3,0, 65535,niemcowa-[0x1f]1001 , 0, 0, 0, 3219,6, 0,17
38,LUBAN , 49.489290, 20.339000,37380.80638, 70, 1, 3,0, 65535,luban-[0x1f]1210 , 0, 0, 0, 3970,6, 0,17
41,ZL PD , 49.453990, 20.583080,37380.80639, 70, 1, 3,0, 65535,zlomisty wierch-[0x1f]1226 , 0, 0, 0, 3940,6, 0,17
```

mało widać?

OK. – ale teraz linie się poprzrenoszą ☺

```
24,025 , 49.472730, 20.539380,37380.80635, 70, 1, 3,0,
65535,skalka-[0x1f]1163 , 0, 0, 0, 3793,6, 0,17
27,27A , 49.496650, 20.264187,37380.81491, 75, 1, 3,0,
65535,runek-[0x1f]1005 , 0, 0, 0, -777,6, 0,17
36,NIEMCOWA , 49.445510, 20.646820,37380.80638, 70, 1, 3,0,
65535,niemcowa-[0x1f]1001 , 0, 0, 0, 3219,6, 0,17
38,LUBAN , 49.489290, 20.339000,37380.80638, 70, 1, 3,0,
65535,luban-[0x1f]1210 , 0, 0, 0, 3970,6, 0,17
41,ZL PD , 49.453990, 20.583080,37380.80639, 70, 1, 3,0, 65535,zlomisty
wierch-[0x1f]1226 , 0, 0, 0, 3940,6, 0,17
```

Oczywiście wszyscy zwrócili uwagę na omawiany separator w nazwie Waypointa?

Informacje dla zaawansowanych

Wstęp

Opisane niżej zagadnienia nie wymagają żadnych specjalnych zdolności, jednak zakłada się, że czytający rozumie notację szesnastkową (hex) oraz zapoznał się z wcześniejszymi rozdziałami i prawidłowo interpretuje określenia Typ i Rodzaj, co jest niezbędne do właściwego korzystania z niniejszej dokumentacji.

1. Kontrola nad rozmieszczeniem obiektów (parametr Levels)

W Szybkim Starcie nie tłumaczono w pełni logiki rozmieszczania obiektów za pomocą parametrów przekazywanych w ich sekcji. O ile parametr Levels został częściowo omówiony, to część z wywołaniem nazwy pliku wymaga kilku słów.

Wiemy już, że aby umieścić obiekt w pięciu warstwach należy:

- a. Upewnić się, że pięć lub więcej warstw zostało zdefiniowanych w sekcji IMG ID
- b. W sekcji obiektu (np. WPT) wpisać parametr Levels=4

Istnieje jednak możliwość umieszczenia obiektu w kilku warstwach, ale nie koniecznie od warstwy 0 (najdokładniejszej). Aby tego dokonać należy połączyć cechę parametru Levels i parametru wywołującego nazwę pliku.

Parametr File0 zawiera nazwę pliku z danymi przeznaczonymi do warstwy 0 i powielenia na wszystkie kolejne warstwy, do warstwy której numer określa parametr Levels. Analogicznie – File2 wskaże na dane dla warstwy 2. Aby więc dane obiekty umieścić w warstwach 2 i 3 należy zgłosić je w następujący sposób:

```
[PLT]
Type=...
RgnType=...
Levels=3           (obiekty mają być powielone do warstwy 3)
File2=...         (a ładowane od warstwy 2 jako najniższej)
[END-PLT]
```

2. Kontrola nad rozmieszczeniem obiektów – sekcja DICTIONARY

Nadrzędnym w stosunku do parametrów Levels i FileX jest filtr ustawiany w sekcji DICTIONARY. Dzięki niemu można globalnie sterować przydziałem obiektów danego typu i rodzaju (odpowiednio: RgnType i Type) do danej warstwy.

Sekcja DICTIONARY składa się z szeregu definicji. Dla każdego typu obiektu i każdej warstwy musi istnieć wpis, jeżeli chcemy sterować filtrowaniem położenia obiektów z rodzaju (Type) tego typu (RgnType). Ten wpis jest parametrem o nazwie zbudowanej w następujący sposób:

Level<X>RGN<YY>=nnnnnn...(256 zer i/lub jedynek)

Gdzie:

- <X> jest numerem warstwy, której dany parametr dotyczy,
- <YY> jest typem obiektu

Aby móc wyszukiwać POI po nazwach w naszych mapach, należy włożyć trochę dodatkowej pracy i posłużyć się aplikacją GpsMapper. Po jej uruchomieniu wczytujemy nasz skompilowany plik IMG...

... a reszta później ☺ - w skrócie – dla dociekliwych – należy wskazać obiekt i wpisać jego właściwości – kod pocztowy, kraj i in, następnie zapisać IMG.

Uwaga. W wersji shareware po eksporcie do pliku TXT tracimy wpisane w ten sposób informacje!!! Nie ma także możliwości importu niezbędnych informacji z pliku TXT – to umożliwia tylko pełna wersja oprogramowania GpsMapper.

4. Kody separatorów

Na mapach można używać dwóch rodzajów znaków – standardowych znaków ASCII i kodowanych (6 bitów). Do wersji 0.045 cGpasMapper obsługuje w pełni kodowanie 6 bitami, obsługa kodowania ośmiobitowego nie jest jeszcze w pełni obsługiwana.

Dopuszczalne znaki dla kodowania 6 bitowego:

```
(space)abcdefghijklmnopqrstuvwxyz0123456789@!"#$%&'()*+,-/:;<=>?[\]
```

Dla osiągnięcia specjalnych efektów można posłużyć się specjalnymi separatorami, i tak:

1Btt= wymuszenie małych liter (w GPS) ~[0x1b]
 1B2B= tylko druga część widoczna przy widoku w skali ponad 1km
 1B2C= tylko pierwsza część widoczna
 1E= tylko druga część widoczna – wstawiany separator (spacja)
 1F= tylko pierwsza część widoczna – wstawiany separator
 1D= pomiędzy pełną nazwa regionu lub kraju - oddziela pełną nazwę od skrótu
 (np: POLSKA~[0x1d]POL)

Kodowanie niestandardowe:

2A=Interstate Highway (name: only digits) np. ~[0x2a]12
 2B=US Highway - shield – znak tarczy (nazwa: tylko cyfry)
 2C=US Highway – okrągły znak (nazwa: tylko cyfry)
 2D=Highway - duży symbol (nazwa: cyfry i litery)
 2E=Main Road - średni symbol (nazwa: litery i cyfry)
 2F=Road - mały symbol (nazwa: litery i cyfry)

5. Ładowanie map

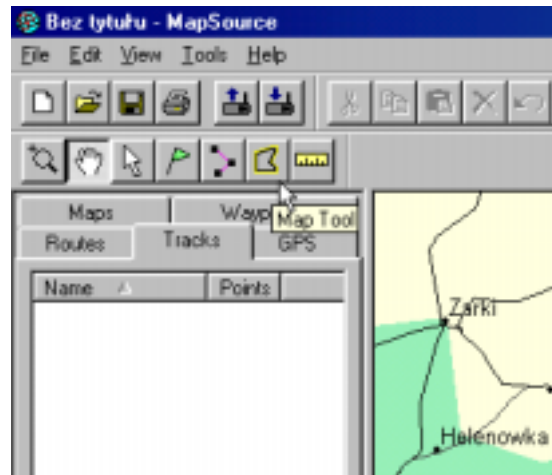
W Szybkim Starcie opisano procedurę ładowania map za pomocą oprogramowania **sendmaps** stanowiącego część pakietu GpsMapper. Posiadacze oprogramowania MapSource chcieliby prawdopodobnie korzystać z map UMP z poziomu jego środowiska.

Jest to możliwe, jak wspomniano wcześniej, na dwa sposoby: elegancki i nieelegancki. Oba są równie skuteczne, choć sposób nieelegancki powoduje, że musimy zrezygnować z jednej z oryginalnych map MapSource na czas ładowania własnej mapy.

a. metoda nieelegancka

Założmy, że nasza mapa UMP zawiera się w obszarze objętym mapą Warsaw z pakietu WorldMap. Plik MapSource zawierający tę mapę nosi nazwę *04307850.img* i znajduje się w podkatalogu **Worldmap**. Musimy odnaleźć ten plik i zmienić jego nazwę (np. na *043007850.im_*).

W sekcji IMG ID w nazwie mapy wpisujemy 04307850 jako ID= i normalnie kompilujemy plik TXT. W wyniku otrzymamy plik IMG, który kopiujemy do katalogu **Worldmap**. Uruchamiamy MapSource, zmieniamy powiększenie do skali, w której pojawiają się nasze obiekty (patrz Załącznik 3, parametry Zoom) i delektujemy się wyglądem naszej mapy. Teraz możemy załadować ją do odbiornika zaznaczając ją narzędziem Map (patrz rysunek obok).



UWAGA. Pamiętaj o zrobieniu kopii oryginalnego pliku IMG. Podczas wykonywania powyższych operacji MapSource powinien być zamknięty, lub selektor map powinien być ustawiony na NoMap. W innym razie oryginalny plik IMG może być użytkowany przez MapSource i zmiana jego nazwy nie będzie możliwa.

b. metoda elegancka

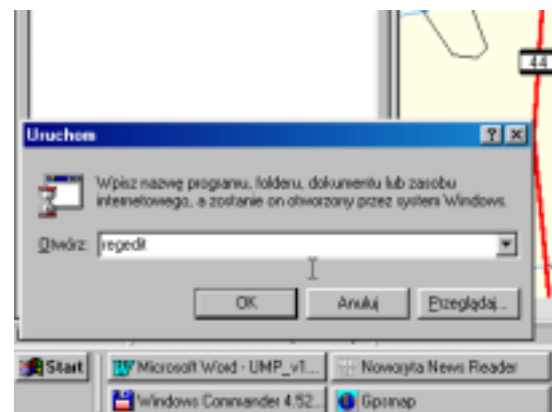
Metoda ta wymaga od Ciebie modyfikowania rejestrów Windows. Jeżeli nie czujesz się na siłach edytować rejestr systemu, lepiej tego nie rób, ponieważ błąd lub nieumiejętnie dokonane wpisy mogą spowodować konieczność ponownej instalacji systemu operacyjnego.

Jeżeli jednak chcesz spróbować, postaram się przedstawić możliwie łopatologiczną metodę dopisania własnego zestawu map.

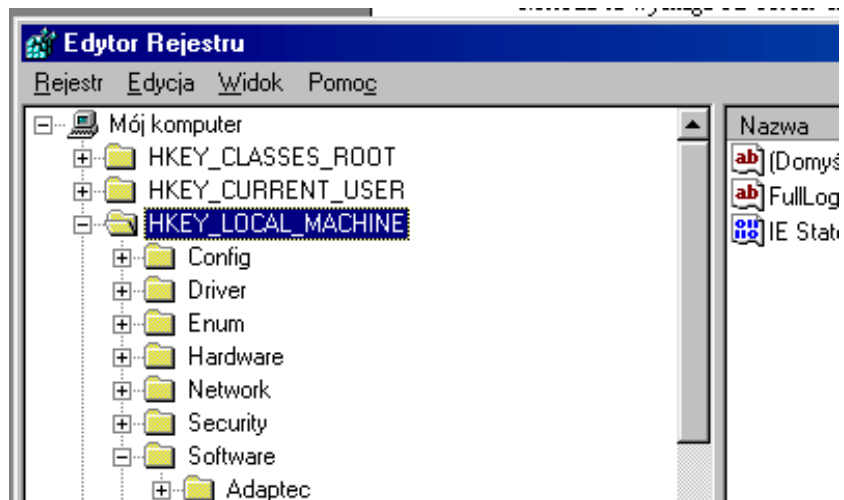
UWAGA. Użytkownicy zaawansowani. Proszę przejść do Załącznika 6, który zawiera kompletny wpis do rejestru. Proszę zwrócić uwagę, że do poprawnej obsługi naszych map przez MapSource wymagane jest stworzenie dwóch dodatkowych plików. Z tego powodu polecamy lekturę tego rozdziału, pomimo że część dotyczące rejestru może wydawać się nieciekawa.

Zaczynamy od uruchomienia edytora rejestru – przycisk Start, Uruchom, 'regedit'

Ilustracja obok przedstawia okno Uruchom w Windows wersji 98.



Po uruchomieniu rozwijamy **HKEY_LOCAL_MACHINE**



Następnie

Software,

Garmin,

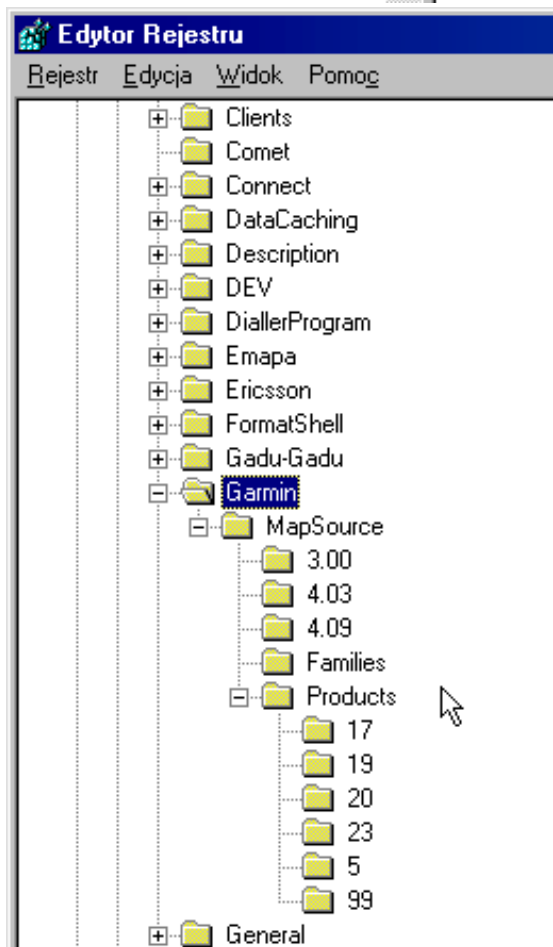
MapSource,

Products

Klikamy prawym przyciskiem na **Products** i wybieramy *Nowy > Klucz*

Następnie wprowadzamy wartość liczbową inną od dotychczasowych. Przykład obok pokazuje, że na swoje potrzeby wybrałem numer **99**. GpsMapper w przykładzie w Załączniku 6 zastosował numer **66**. Jest tutaj pełna dowolność, ale należy wybrać unikalną liczbę, zgodną z parametrem ProductCode w pliku PV z Załącznika 7.

Teraz klikamy prawym przyciskiem na utworzonym 'katalogu', wybieramy *Nowy>Wartość Ciągu* i wpisujemy **Bmap**, powtarzamy *Nowy>Wartość Ciągu* jeszcze dwukrotnie wpisując kolejno **Loc** i **Tdb**.



W rezultacie naszych poczynań uzyskamy wpis do rejestru, który pozostaje jedynie uzupełnić o właściwe ścieżki lokalizujące nasze pliki.

Osoby średniozaawansowane wiedzą już pewnie, co dalej. Dla początkujących podamy sposób postępowania narzucając lokalizację plików, ale po zrozumieniu idei działania wpisów można je w każdej chwili zmodyfikować dostosowując do zmienionej lokalizacji.

W katalogu, w którym zainstalowano MapSource (np. 'C:\Program Files\MapSource') tworzymy nowy katalog – MG_UMP. Do tego katalogu kopiujemy własne pliki IMG uzyskane z kompilacji plików TXT lub przez dodanie obiektów do innych plików IMG.

Klikamy dwukrotnie na wpisie **Bmap** i w okno, które się otworzy wpisujemy

C:\Program Files\MapSource\mg_ump.img

Klikamy dwukrotnie na wpisie **Loc** i w okno, które się otworzy wpisujemy

C:\Program Files\MapSource\mg_ump

Klikamy dwukrotnie na wpisie **Tdb** i w okno, które się otworzy wpisujemy

C:\Program Files\MapSource\mg_ump.tdb

Aby uzyskać pełną funkcjonalność MapSource trzeba jeszcze utworzyć pliki **mg_ump.img** i **mg_ump.tdb** do których odwołania właśnie utworzyliśmy. Posłużymy się w tym celu programem cGpsMapper.exe z parametrem pv oraz specjalnie utworzonym plikiem tekstowym, który znajduje się w Załączniku 7.

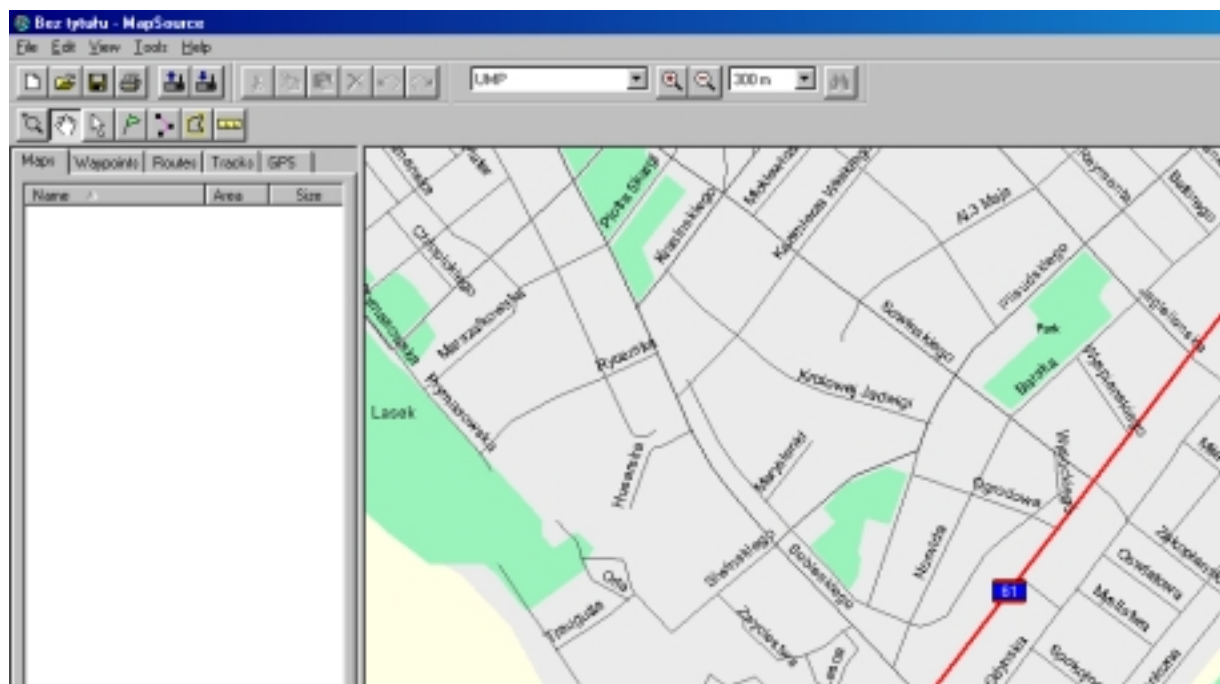
Przykładowy plik z Załącznika 7 musimy zmodyfikować tak, aby zawartość sekcji FILE wskazywała na nasze pliki IMG.

Wykonujemy teraz kompilację tego pliku. Zapisujemy go pod nazwą **pv.txt**, dla uproszczenia umieszczamy go w katalogu GpsMappera, uruchamiamy okno Command Prompt i wykonujemy polecenie **cGpsMapper.exe -pv pv.txt**.

W wyniku działania programu powstaną dwa pliki, które należy przenieść do katalogu wskazanego w kluczu **Loc**.

Teraz uruchamiamy MapSource i na liście dostępnych produktów widzimy nasze UMP (lub inną nazwę – parametr MapSourceName z pliku pv). Widzialność obiektów na mapie zależy od wybranych wartości Zoom, więc jeżeli nie widzimy własnej mapy, być może trzeba zmienić skalę.

Oto rezultat dla UMP v.3 z października 2002 (fragment Legionowo):



6. Tworzenie obiektów typu 0x40 z innymi obiektami 0x40 pośrodku

Przy tworzeniu bardziej skomplikowanych map należy trzymać się jednej zasady: unikać nakładania się obszarów, ponieważ wpływa to niekorzystnie na szybkość wyświetlania mapy w odbiorniku oraz trudno czasami określić kolejność wyświetlania obiektów.

Dodatkowo pamiętać należy, że tłem mapy (najniższą - domyślną warstwą) jest ląd, a więc morza, jeziora i oceany stanowią 'plamy' na tym tle. Jeżeli chcemy uzyskać efekt wyspy, musimy skonstruować obszar wodny w kształcie litery C, prawie domkniętej do O. Ponieważ zachowanie punktowej przerwy spowoduje w rezultacie utworzenie półwyspu o punktowej szerokości, należy zbliżyć oba brzegi tego półwyspu aby się nałożyły.

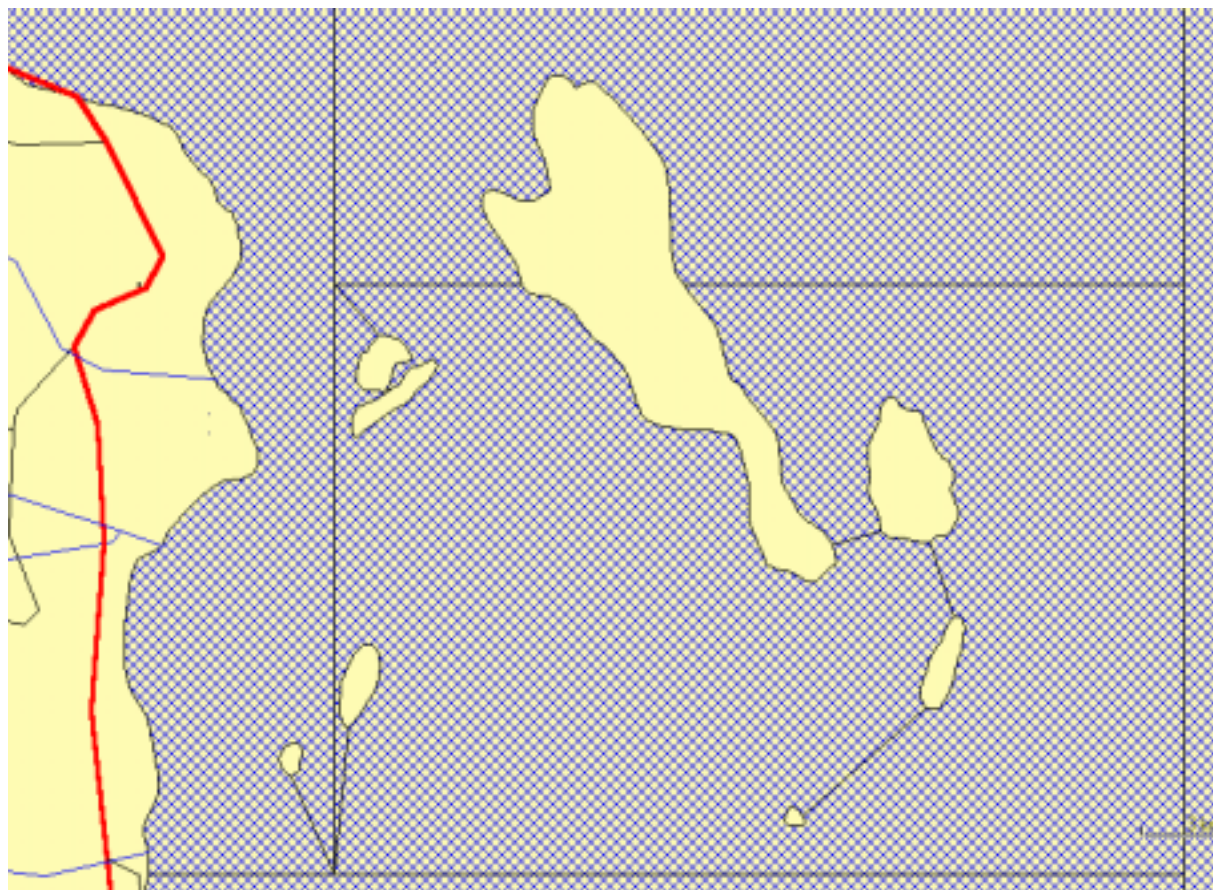
Alternatywnym rozwiązaniem jest stworzenie szeregu elementów typu woda, które zestawione ze sobą pozostawią odkryte fragmenty. To rozwiązanie stosowane jest przez twórców oryginalnych map Garmin, co ilustrują poniższe przykłady:



Fragment wybrzeża Afryki (Egipt), Morze Czerwone i kilka wysp w okolicy Hurgady.

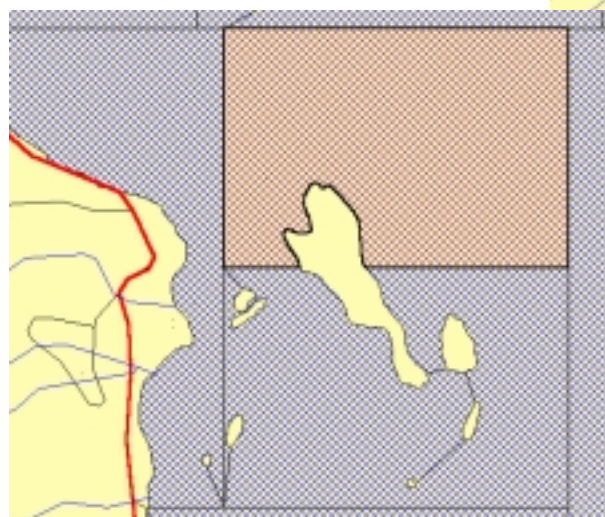
A oto jak wygląda ten fragment przed kompilacją (albo po dekompilacji, cokolwiek uznamy za stan wyjściowy ☺)

Przykład poniższy świetnie ilustruje wykorzystanie obu wspomnianych technik do stworzenia wysp.



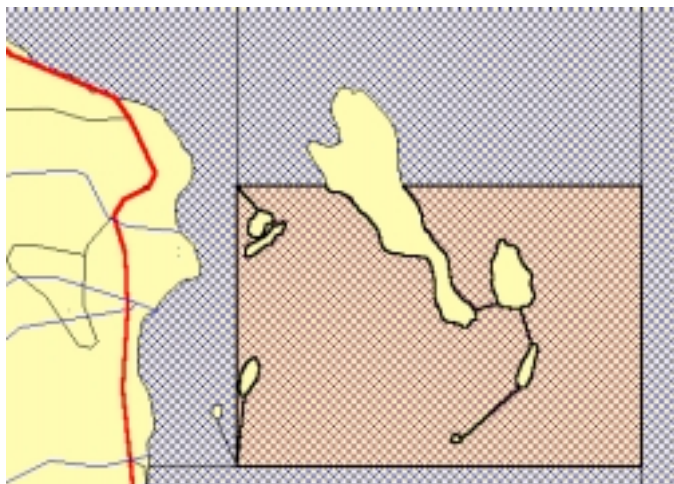
Przyjrzyjmy mu się bliżej:

Zaznaczony fragment tworzy linię brzegową oraz jedną wysepkę korzystając z techniki pierwszej. Z pozostałymi fragmentami styka się bokami tworzącymi linie proste.



Drugi element tworzy dalszy ciąg morza czerwonego oraz pozostawia odsłonięty fragment lądu – północny fragment wyspy.

Trzeci element wykorzystuje obie opisywane techniki. Południowa część wyspy połączona 'mostami' o zerowej szerokości (widoczne jako linie) z trzema pozostałymi, mniejszymi wysepkami oraz pozostałe wyspy na zachód.



Oczywiście w przypadku jeziora, pozostałe brzegi obszarów nie będą liniami prostymi ale utworzą całkowitą linię brzegową.

GpsMapper – opis podstawowych funkcji programu

Do czego służy GpsMapper?

Daje

- możliwość 'finalnej edycji'
- możliwość usuwania wybranego regionu i wszystkich podregionow (można używać do dzielenia mapy na mniejsze części)
- możliwość 'doimportowania' do istniejących IMG plików PLT/WPT
- możliwość wymuszenia indeksowania POI poprzez edycje POI i dodanie informacji dodatkowych

Pozwala na

- wyszukiwanie pliku wg wskazanych współrzędnych (lokalizacja odpowiedniego pliku IMG)
- przeglądanie plików IMG wg wyboru

Opisy szczegółowe w następnej wersji.

Załącznik 1

Definicja Typów i Rodzajów obiektów

Nazwa	Type (HEX)	RgnType (HEX)
POI – Punkty indeksowane do wyszukiwania		
Namedpoint (big)	0x0100	0x10
Namedpoint (big point)	0x0600	0x10
Namedpoint (big point)	0x0e00	0x10
Namedpoint (small point)	0x0b00	0x10
Namedpoint (small point)	0x0c00	0x10
Namedpoint (small point)	0x0d00	0x10
Namedpoint (small point)	0x2800	0x10
Namedpoint (big)	0x1400	0x10
Namedpoint (middle)	0x1e00	0x10
Exit	0x20	0x10
Exit with facilities	0x2100	0x10
Exit(Restroom)	0x22	0x10
Exit(Convenience Store)	0x23	0x10
Exit(Weigh Station)	0x24	0x10
Exit(Toll Booth)	0x25	0x10
Exit(Information)	0x26	0x10
Exit	0x27	0x10
Dining(Other)	0x2a	0x10
Dining(American)	0x2a01	0x10
Dining(Asian)	0x2a02	0x10
Dining(Barbecue)	0x2a03	0x10
Dining(Chinese)	0x2a04	0x10
Dining(Deli/Bakery)	0x2a05	0x10
Dining(International)	0x2a06	0x10
Fast Food	0x2a07	0x10
Dining(Italian)	0x2a08	0x10
Dining(Mexican)	0x2a09	0x10
Dining(Pizza)	0x2a0a	0x10
Dining(Sea Food)	0x2a0b	0x10
Dining(Steak/Grill)	0x2a0c	0x10
Dining(Bagel/Donut)	0x2a0d	0x10
Dining(Cafe/Diner)	0x2a0e	0x10
Dining(French)	0x2a0f	0x10
Dining(German)	0x2a10	0x10
Dining(British Isle)	0x2a11	0x10
Hotel(Other)	0x2b00	0x10
Hotel/Motel	0x2b01	0x10
Bed&Breakfast inn	0x2b02	0x10
Camping/RV-Park	0x2b03	0x10
Resort	0x2b04	0x10
Amusement Park	0x2c01	0x10
Museum/History	0x2c02	0x10
Libraries	0x2c03	0x10
Land Mark	0x2c04	0x10
School	0x2c05	0x10
Park	0x2c06	0x10
Zoo	0x2c07	0x10
"Sportcenter,Stadion,Race"	0x2c08	0x10
Conference	0x2c09	0x10
Vine	0x2c0a	0x10
Theatre	0x2d01	0x10
Bar	0x2d02	0x10
Cinema	0x2d03	0x10
Casino	0x2d04	0x10
Golf	0x2d05	0x10
Skiing Center	0x2d06	0x10
Bowling	0x2d07	0x10
Ice-Skate	0x2d08	0x10
Swimming	0x2d09	0x10
Sports	0x2d0a	0x10
Sport Airport	0x2d0b	0x10

Department Store	0x2e01	0x10
Grocery	0x2e02	0x10
General Merchandiser	0x2e03	0x10
Shopping-Center	0x2e04	0x10
Pharmacie	0x2e05	0x10
Convienc	0x2e06	0x10
Special	0x2f	0x10
Petrol	0x2f01	0x10
Auto Rental	0x2f02	0x10
Auto Repair	0x2f03	0x10
Airport	0x2f04	0x10
Post Office	0x2f05	0x10
Bank	0x2f06	0x10
Autohandel	0x2f07	0x10
Bus Station	0x2f08	0x10
Marina	0x2f09	0x10
Wrecker Service	0x2f0a	0x10
Parking	0x2f0b	0x10
Restroom	0x2f0c	0x10
Automobil Club	0x2f0d	0x10
Car Wash	0x2f0e	0x10
Policestation	0x3001	0x10
Hospital	0x3002	0x10
Public Office	0x3003	0x10
Court	0x3004	0x10
Disco	0x3005	0x10
Zoll	0x3006	0x10
Golf	0x4000	0x10
Fish	0x4100	0x10
Wreck	0x4200	0x10
Marina	0x4300	0x10
Gas Station	0x4400	0x10
Restaurant	0x4500	0x10
Bar	0x4600	0x10
Boat Ramp	0x4700	0x10
Camping	0x4800	0x10
Park	0x4900	0x10
Picnic Area	0x4a00	0x10
Hospital	0x4b00	0x10
Information	0x4c00	0x10
Parking	0x4d00	0x10
Restroom	0x4e00	0x10
Shower	0x4f00	0x10
Drinking Water	0x5000	0x10
Telephone	0x5100	0x10
Scenic Area	0x5200	0x10
Skiing	0x5300	0x10
Swimming	0x5400	0x10
Dam	0x5500	0x10
Danger Area	0x5700	0x10
Heliport	0x5904	0x10
Airport	0x5900	0x10
Airport	0x5905	0x10
"Daymark,Green Square"	0x5d00	0x10
"Daymark,Red Triangle"	0x5e00	0x10
Bridge	0x6401	0x10
Building	0x6402	0x10
Cemetery	0x6403	0x10
Church	0x6404	0x10
Civil	0x6405	0x10
Crossing	0x6406	0x10
Dam	0x6407	0x10
Hospital	0x6408	0x10
Levee	0x6409	0x10

Locale	0x640a	0x10
Military	0x640b	0x10
Mine	0x640c	0x10
Oil Field	0x640d	0x10
Park	0x640e	0x10
Post	0x640f	0x10
School	0x6410	0x10
Tower	0x6411	0x10
Trial	0x6412	0x10
Tunnel	0x6413	0x10
Drinking Water	0x6414	0x10
Ghost Town	0x6415	0x10
Subdivision	0x6416	0x10
Arroyo	0x6501	0x10
Sand Bar	0x6502	0x10
Bay	0x6503	0x10
Bend	0x6504	0x10
Canal	0x6505	0x10
Channel	0x6506	0x10
Cove	0x6507	0x10
Falls	0x6508	0x10
Geyser	0x6509	0x10
Glacier	0x650a	0x10
Harbor	0x650b	0x10
Island	0x650c	0x10
Lake	0x650d	0x10
Rapids	0x650e	0x10
Reservoir	0x650f	0x10
Sea	0x6510	0x10
Spring	0x6511	0x10
Stream	0x6512	0x10
Swamp	0x6513	0x10
Arch	0x6601	0x10
Area	0x6602	0x10
Basin	0x6603	0x10
Beach	0x6604	0x10
Beach	0x6605	0x10
Cape	0x6606	0x10
Cliff	0x6607	0x10
Crater	0x6608	0x10
Flat	0x6609	0x10
Forest	0x660a	0x10
Gap	0x660b	0x10
Gut	0x660c	0x10
Isthmus	0x660d	0x10
Lava	0x660e	0x10
Pillar	0x660f	0x10
Plain	0x6610	0x10
Range	0x6611	0x10
Reserve	0x6612	0x10
Ridge	0x6613	0x10
Rock	0x6614	0x10
Slope	0x6615	0x10
Summit	0x6616	0x10
Valley	0x6617	0x10
Woods	0x6618	0x10
Fog Horn	0x1b01	0x10
Fog Horn	0x1a01	0x10
Fog Horn	0x1901	0x10
Fog Horn	0x1801	0x10
Fog Horn	0x1701	0x10
Fog Horn	0x1601	0x10
Radio Beacon	0x1b02	0x10
Radio Beacon	0x1a02	0x10
Radio Beacon	0x1902	0x10
Radio Beacon	0x1802	0x10
Radio Beacon	0x1702	0x10
Radio Beacon	0x1602	0x10

Racon	0x1b03	0x10
Racon	0x1a03	0x10
Racon	0x1903	0x10
Racon	0x1803	0x10
Racon	0x1703	0x10
Racon	0x1603	0x10
"Daybeacon,red Triangle"	0x1b04	0x10
"Daybeacon,red Triangle"	0x1a04	0x10
"Daybeacon,red Triangle"	0x1904	0x10
"Daybeacon,red Triangle"	0x1804	0x10
"Daybeacon,red Triangle"	0x1704	0x10
"Daybeacon,red Triangle"	0x1604	0x10
"Daybeacon,green Square"	0x1b05	0x10
"Daybeacon,green Square"	0x1a05	0x10
"Daybeacon,green Square"	0x1905	0x10
"Daybeacon,green Square"	0x1805	0x10
"Daybeacon,green Square"	0x1705	0x10
"Daybeacon,green Square"	0x1605	0x10
"Daybeacon,white Diamond"	0x1b06	0x10
"Daybeacon,white Diamond"	0x1a06	0x10
"Daybeacon,white Diamond"	0x1906	0x10
"Daybeacon,white Diamond"	0x1806	0x10
"Daybeacon,white Diamond"	0x1706	0x10
"Daybeacon,white Diamond"	0x1606	0x10
"unlit Navaid,white"	0x1b07	0x10
"unlit Navaid,white"	0x1a07	0x10
"unlit Navaid,white"	0x1907	0x10
"unlit Navaid,white"	0x1807	0x10
"unlit Navaid,white"	0x1707	0x10
"unlit Navaid,white"	0x1607	0x10
"unlit Navaid,red"	0x1b08	0x10
"unlit Navaid,red"	0x1a08	0x10
"unlit Navaid,red"	0x1908	0x10
"unlit Navaid,red"	0x1808	0x10
"unlit Navaid,red"	0x1708	0x10
"unlit Navaid,red"	0x1608	0x10
"unlit Navaid,green"	0x1b09	0x10
"unlit Navaid,green"	0x1a09	0x10
"unlit Navaid,green"	0x1909	0x10
"unlit Navaid,green"	0x1809	0x10
"unlit Navaid,green"	0x1709	0x10
"unlit Navaid,green"	0x1609	0x10
"unlit Navaid,black"	0x1b0a	0x10
"unlit Navaid,black"	0x1a0a	0x10
"unlit Navaid,black"	0x190a	0x10
"unlit Navaid,black"	0x180a	0x10
"unlit Navaid,black"	0x170a	0x10
"unlit Navaid,black"	0x160a	0x10
"unlit Navaid,yellow or amber"	0x1b0b	0x10
"unlit Navaid,yellow or amber"	0x1a0b	0x10
"unlit Navaid,yellow or amber"	0x190b	0x10
"unlit Navaid,yellow or amber"	0x180b	0x10
"unlit Navaid,yellow or amber"	0x170b	0x10
"unlit Navaid,yellow or amber"	0x160b	0x10
"unlit Navaid,orange"	0x1b0c	0x10
"unlit Navaid,orange"	0x1a0c	0x10
"unlit Navaid,orange"	0x190c	0x10
"unlit Navaid,orange"	0x180c	0x10
"unlit Navaid,orange"	0x170c	0x10
"unlit Navaid,orange"	0x160c	0x10
"unlit Navaid,multi colored"	0x1b0d	0x10
"unlit Navaid,multi colored"	0x1a0d	0x10
"unlit Navaid,multi colored"	0x190d	0x10
"unlit Navaid,multi colored"	0x180d	0x10
"unlit Navaid,multi colored"	0x170d	0x10
"unlit Navaid,multi colored"	0x160d	0x10
"Navaid,unknown"	0x1b0e	0x10
"Navaid,unknown"	0x1a0e	0x10

"Navaid,unknown"	0x190e	0x10
"Navaid,unknown"	0x180e	0x10
"Navaid,unknown"	0x170e	0x10
"Navaid,unknown"	0x160e	0x10
"lighted Navaid,white"	0x1b0f	0x10
"lighted Navaid,white"	0x1a0f	0x10
"lighted Navaid,white"	0x190f	0x10
"lighted Navaid,white"	0x180f	0x10
"lighted Navaid,white"	0x170f	0x10
"lighted Navaid,white"	0x160f	0x10
"lighted Navaid,red"	0x1b10	0x10
"lighted Navaid,red"	0x1a10	0x10
"lighted Navaid,red"	0x1910	0x10
"lighted Navaid,red"	0x1810	0x10
"lighted Navaid,red"	0x1710	0x10
"lighted Navaid,red"	0x1610	0x10
"lighted Navaid,green"	0x1b11	0x10
"lighted Navaid,green"	0x1a11	0x10
"lighted Navaid,green"	0x1911	0x10
"lighted Navaid,green"	0x1811	0x10
"lighted Navaid,green"	0x1711	0x10
"lighted Navaid,green"	0x1611	0x10
"lighted Navaid,yellow or amber"	0x1b12	0x10
"lighted Navaid,yellow or amber"	0x1a12	0x10
"lighted Navaid,yellow or amber"	0x1912	0x10
"lighted Navaid,yellow or amber"	0x1812	0x10
"lighted Navaid,yellow or amber"	0x1712	0x10
"lighted Navaid,yellow or amber"	0x1612	0x10
"lighted Navaid,orange"	0x1b13	0x10
"lighted Navaid,orange"	0x1a13	0x10
"lighted Navaid,orange"	0x1913	0x10
"lighted Navaid,orange"	0x1813	0x10
"lighted Navaid,orange"	0x1713	0x10
"lighted Navaid,orange"	0x1613	0x10
"lighted Navaid,violet"	0x1b14	0x10
"lighted Navaid,violet"	0x1a14	0x10
"lighted Navaid,violet"	0x1914	0x10
"lighted Navaid,violet"	0x1814	0x10
"lighted Navaid,violet"	0x1714	0x10
"lighted Navaid,violet"	0x1614	0x10
"lighted Navaid,blue"	0x1b15	0x10
"lighted Navaid,blue"	0x1a15	0x10
"lighted Navaid,blue"	0x1915	0x10
"lighted Navaid,blue"	0x1815	0x10
"lighted Navaid,blue"	0x1715	0x10
"lighted Navaid,blue"	0x1615	0x10
"lighted Navaid,milti colored"	0x1b16	0x10
"lighted Navaid,milti colored"	0x1a16	0x10
"lighted Navaid,milti colored"	0x1916	0x10
"lighted Navaid,milti colored"	0x1816	0x10
"lighted Navaid,milti colored"	0x1716	0x10
"lighted Navaid,milti colored"	0x1616	0x10
Unclassified Obstruction	0x1c00	0x10
Wreck	0x1c01	0x10
"submerged Wreck,dangerous"	0x1c02	0x10
"submerged Wreck, non-dangerous"	0x1c03	0x10
"Wreck,cleared by Wire-drag"	0x1c04	0x10
"Obstr.,visible at high Water"	0x1c05	0x10
"Obstruction,awash"	0x1c06	0x10
"Obstruction,submerged"	0x1c07	0x10
"Obstr.,cleared by Wire-drag"	0x1c08	0x10
"Rock,awash"	0x1c09	0x10
"Rock,submerged at low Water"	0x1c0a	0x10
Sounding	0x1c0b	0x10
PUNKTY ZWYKLE		
City1 - Large City	0x01	0x20
City2 - Large City	0x02	0x20

City3 - Large City	0x03	0x20
City4 - Large City	0x04	0x20
City5 - Large City	0x05	0x20
City6 - Large City	0x06	0x20
City7 - Medium City	0x07	0x20
City8 - Medium City	0x08	0x20
City9 - Small City	0x09	0x20
City10 - Small City	0x0a	0x20
City11 - Small City	0x0b	0x20
City12 - Small Town	0x0c	0x20
City13 - Unknown City	0x0d	0x20
*	0x0e	0x20
*	0x0f	0x20
*	0x10	0x20
*	0x11	0x20
Info(not vis)	0x12	0x20
*	0x13	0x20
Region name	0x14	0x20
Region name	0x15	0x20
Navaid	0x16	0x20
Navaid	0x17	0x20
Navaid	0x18	0x20
Navaid	0x19	0x20
Navaid	0x1a	0x20
Tide	0x1d	0x20
WC	0x22	
Scales	0x24	
Restaurant	0x2a	
Hotel	0x2b	
Shop	0x2e	
Golf	0x40	
Fish	0x41	0x20
Wreck	0x42	0x20
Marina	0x43	0x20
Fuel	0x44	0x20
Restaurant	0x45	
Beer	0x46	
Boat	0x47	
Ramp	0x47	0x20
Campground	0x48	0x20
Park	0x49	0x20
Picnic Area	0x4a	0x20
First Aid	0x4b	0x20
Information	0x4c	0x20
Restroom	0x4e	0x20
Showers	0x4f	0x20
Drinking Water	0x50	0x20
Telephone	0x51	0x20
Scenic Area	0x52	0x20
Skiing	0x53	0x20
Swimming	0x54	0x20
Dam	0x55	0x20
Danger Area	0x57	0x20
Airport	0x59	0x20
Daymark, Green Square	0x5d	0x20
Daymark, Red Triangle	0x5e	0x20
(height point [feet])	0x62	0x20
(height point [feet])	0x63	0x20
(blue point)	0x64	0x20
(blue point)	0x65	0x20
(blue point)	0x66	0x20
LINIE		
Major HWY - thick	0x01	0x40
Principal HWY - thick	0x02	0x40
Principal HWY - medium	0x03	0x40
Arterial Road - medium	0x04	0x40
Arterial Road - thin	0x05	0x40
Road - thin	0x06	0x40

Unpaved Road - thin	0x0a	0x40
Alley - thin	0x07	0x40
Ramp - thin	0x08	0x40
Ramp - thin	0x09	0x40
Unpaved Road	0x0a	
Major HWY Connector - thick	0x0b	0x40
Roundabout - thin	0x0c	0x40
Railroad	0x14	0x40
Shoreline	0x15	0x40
Trail	0x16	0x40
Stream	0x18	0x40
Time-Zone	0x19	0x40
Ferry	0x1a	0x40
Ferry	0x1b	0x40
Political Boundary	0x1c	0x40
Conty Boundary	0x1d	0x40
Intl. Boundary	0x1e	0x40
River	0x1f	0x40
Land Contour - thin	0x20	0x40
Land Contour - medium	0x21	0x40
Land Contour - thick	0x22	0x40
Depth Contour - thin	0x23	0x40
Depth Contour - medium	0x24	0x40
Depth Contour - thick	0x25	0x40
Intermittent River	0x26	0x40
Airport Runway	0x27	0x40
Pipeline	0x28	0x40
Powerline	0x29	0x40
Marine Boundary	0x2a	0x40
Marine Hazard	0x2b	0x40
OBSZARY, POLA itp.		
City	0x01	0x80
City	0x02	0x80
City	0x03	0x80
Military	0x04	0x80
Parking Lot	0x05	0x80
Parking Garage	0x06	0x80
Airport	0x07	0x80
Shopping Center	0x08	0x80
Marina	0x09	0x80
University	0x0a	0x80
Hospital	0x0b	0x80
Industrial	0x0c	0x80
Reservation	0x0d	0x80
Airport Runway	0x0e	0x80
National park	0x14	0x80
National park	0x15	0x80
National park	0x16	0x80
City Park	0x17	0x80
Golf	0x18	0x80
Sport	0x19	0x80
Cemetery	0x1a	0x80
State Park	0x1e	0x80
State Park	0x1f	0x80
Ocean	0x28	0x80
Sea	0x32	0x80
Lake	0x3c	0x80
Lake	0x3d	0x80
Lake	0x3e	0x80
Lake	0x3f	0x80
Lake	0x40	0x80
Lake	0x41	0x80
Lake	0x42	0x80
Lake	0x43	0x80
Lake	0x44	0x80
River	0x46	0x80
River	0x47	0x80
River	0x48	0x80

River	0x49	0x80
Intermittent River/Lake	0x4c	0x80
Glaciers	0x4d	0x80
Woods	0x50	0x80
Wetland	0x51	0x80
Flats	0x53	0x80

Dla uporządkowania:

0x10 – POI

0x20 – zwykle punkty, miasta itp

0x40 – ulice, strumienie, linie kolejowe, promowe, poziomnice, linie energetyczne i inne linie pojedyncze

0x80 – obszary – jeziora, tereny miejskie, zabudowania, parki, lasy, inne obszary

Załącznik 2

Stosowane dotychczas w UMP reguły przydzielania kodów rodzaju i typu obiektów:

Major HWY - thick 0x01 autostrada
Principal HWY - thick 0x02 droga główna duża
Principal HWY - medium 0x03 droga główna średnia
Arterial Road - medium 0x04 średnia droga przelotowa
Arterial Road - thin 0x05 mała droga przelotowa
Road - thin 0x06 zwykła droga
Unpaved Road - thin 0x0a droga nieutwardzona
Alley - thin 0x07 aleja
Trail - 0x16 szlak

Uwaga.

Ostatnio rozważamy zastosować dla map przedstawiających TYLKO szlaki turystyczne kody dróg do oznaczenia szlaków. Pomysł wziął się stąd (JurekR), że dla dróg można zmieniać wielkość liter w opisie w odbiorniku i są czytelniejsze niż jasnoszare lub przerywane linie o niższych ważnościach.

Dodatkowo poziomice:

Land Contour - thin 0x20
Land Contour - medium 0x21
Land Contour - thick 0x22

oraz linie głębokości
Depth Contour - thin 0x23
Depth Contour - medium 0x24
Depth Contour - thick 0x25

Załącznik 3

Opis parametrów sekcji IMG ID (na razie w języku), w oparciu o informacje od GpsMapperpera. Od wersji 0.045 parametry wpisane kursywą są ignorowane:

```

;[IMG ID]
;#####
;
;ID      = (required) integer number, id number of map - must be unique for each map
;        may be written in decimal format or special hex format with 'I' letter :
;        DEC    HEX
;        11000204 I00A7D98C
;#####
;Datum   = (optional) code name of source datum, default value is W84
;        ex. Datum=NAS-A
;#####
;POIIndex = (optional-full version only) Y/N - if set to 'Y' all RGN10 elements will be
;        indexed (to activate button 'find' in GPS) - without this option only
;        with provided additional info (ex. street/description) would be indexed
;        default is 'N'
;#####
;POINumberFirst = (optional-full version only) Y/N - if set to 'Y', then the house number
;        (for POI) will be BEFORE street name
;        default is 'Y'
;#####
;POIZipFirst = (optional-full version only) Y/N - if set to 'Y', then the zip-code (for POI)
;        will be BEFORE street name
;        default is 'Y'
;#####
;AlignMethod = (optional) -1,0,positive real value
;        Chosen method for coordinate alignment -
;        -1 - no special alignment - should not be used!
;        0 - grids in all regions will be aligned
;        positive value - maximum error between given coordinates and
;        calculated coordinates for chosen region - if error exceed
;        the parameter then other region will be chosen or new region
;        will be created.
;        If two nodes belonging to different elements (except RGN10, RGN20)
;        will be nearer than the AlignMethod distance - then during import both of them will be
;        connected
;        suggested value : between 0.001 and 0.000001
;#####
;Name     = (required) map name - character string
;#####
;TreSize  = (optional) maximum allowed region size - in grids - the bigger value, the
;        newly created region may be bigger, but performance of map might
;        be worst, in other hand, the smaller region, the more regions
;        must be created...
;        suggested value : between 2000 and 4000
;#####
;TreMargin = (optional) additional margin for newly created region (in degree)
;        suggested value : 0.02
;#####
;LevelFill = (optional) approximate percentage of element existing in Layer compared to Layer+1
;        if during creation of new map there would be not enough space
;        in certain region, this value may be increased
;        suggested value : 75
;#####
;LevelLimit = (optional) limit of element for a single region (value less or equal to 255)
;        this value is used only during creating new regions on other
;        layer than the most detailed one
;        Decreasing this value can help if some elements could not be imported
;        suggested value : 200
;#####
;RgnLimit = (optional) limit for number of elements in one region
;        can be any value between ~50 - 1024 (value less than 50 doesn't make sense)
;        suggested value : 700
;#####

```

```

;WorldMap = (optional) Y,N
;   if set to 'Y' then special procedure will be used to create
;   regions on layer 1 - more regions will be created
;#####
;Levels = (required) number of layers for map, last layer - in this example Level3
;   is always empty - no objects will be imported into
;#####
;Level0 = (required) grid for each layer, layer 0 is the most detailed one
;   grid value is used to calculate level grid in layer - in degree value
;   standard value for most detailed level is 22.
;   To choose best grid values you can use the following table :
;
;   Grid value also defines visibility in GPS!
;
;
; Grid   Degree Grid       Visibility for normal settings in GPS
; 24    2.14576721191406E-05  0m - 120m
; 23    4.29153442382812E-05  200m - 300m
; 22    8.58306884765625E-05  500m
; 21    0.000171661376953125  800 - 1.2km
; 20    0.00034332275390625  2km
; 19    0.0006866455078125  3km
; 18    0.001373291015625  5 - 8km
; 17    0.00274658203125  12km
; 16    0.0054931640625
; 15    0.010986328125
; 14    0.02197265625
;#####
;Zoom0 = (required-used only by MapSource) zoom level for which given layer should be visible in MapSource
;   ex. Zoom1 = 4 means Layer1 will be visible in zoom range between 30km and 12km
;   WARNING! zoom values for successive layer must be ascending!
;
; Zoom   Approximate visibility in MapSource
; 7      starting from 300km zoom till 120km
; 6      starting from 120km zoom till 50km
; 5      starting from 50km zoom till 30km
; 4      starting from 30km zoom till 12km
; 3      starting from 12km zoom till 8km
; 2      starting from 8km zoom till 3km
; 1      starting from 3km zoom till 1.2km
; 0      starting from 1.2km zoom
;#####
; LIMITATIONS :
; - Size of a single element cannot exceed maximum allowed size of region
;   (maximum size of region can be calculated from TreSize*DegreeGrid for
;   given level)
; - maximum number of regions is limited to 65535 - optimum number of regions
;   is about 50 for small maps and 200-300 for big ones
;#####

```

Załącznik 4

Definicja warstw – ich dokładności i widzialności (w oparciu o dokument JurkaR). Wg. opinii Jurka z września 2002 jest przestarzały (he he), ale na razie zostaje w celach dydaktycznych.

Mapa składa się z warstw, które leżą na różnych piętrach. Oglądając mapę widzimy w danym momencie tylko elementy z jednej warstwy. Której - zależy od stopnia powiększenia. Od piętra zależy dokładność z jaką elementy z warstwy przybliżają rzeczywistość. Elementami mapy są łamane (zamknięte - lasy, jeziora, miasta itd lub otwarte - drogi, koleje, strumienie itd.) oraz punkty.

Elementy z jednej warstwy są podzielone na regiony (prostokątne obszary) przy czym każdy element musi być całkowicie w jakimś regionie. Regiony mogą się nakładać. W każdym regionie wyznaczona jest prostokątna siatka o wielkości oczka zależnej od piętra, na którym znajduje się region (zawierająca go warstwa). Dokładniej: siatki z piętra n mają wielkość oczka równą $360 \cdot 2^{-n}$ (mierzone w stopniach!!) tzn. dla piętra 14 jest to 0.02197265625, dla piętra 18 jest to 0.001373291015625, dla piętra 21 jest to 0.000171661376953125, dla piętra 24 jest to 0,000021457672119140625 (to najwyższe piętro).

Oczko siatki jest (w normalnych współrzędnych) "prostokątem" (o krawędziach będących łukami). W Polsce proporcje takiego prostokąta to mniej więcej 1:0.6 (nasz równoleżnik jest tyleż razy krótszy od południka). Dla piętra 24 wymiary oczka (u nas) to ok 2.5m na 1.5m, dla kolejnych pięter niższych trzeba to mnożyć przez 2, 4, 8 itd. Punkty oraz wierzchołki łamanych z danego regionu (jak też wierzchołki regionu) mogą być tylko w punktach kratowych (w przecięciach linii siatki). Siatki z różnych regionów z tej samej warstwy (zachodzących na siebie) nie zawsze się nakładają (ale wielkość oczka mają taką samą!!), co może sprawiać kłopoty przy przedłużaniu linii.

Nie wszystkie piętra są wykorzystane, zwykle tylko kilka. Dodatkowo każdy region z danej warstwy ma swój rejon ojcowski z warstwy niższej bezpośrednio (co może oznaczać kilka pięter niżej). Rejon jest całkowicie zawarty w rejonie ojcowskim.

Piętro, na którym jest warstwa określa też przy jakiej skali widzimy tę warstwę na ekranie GPS. Podaję za GpsMapperem:

```
; Grid      Degree Grid      Visibility for normal settings in GPS
; 24        2.14576721191406E-05  0m - 120m
; 23        4.29153442382812E-05  200m - 300m
; 22        8.58306884765625E-05  500m
; 21        0.000171661376953125  800 - 1.2km
; 20        0.00034332275390625  2km
; 19        0.0006866455078125  3km
; 18        0.001373291015625  5 - 8km
; 17        0.00274658203125  12km
; 16        0.0054931640625
; 15        0.010986328125
; 14        0.02197265625
```

Powyzsza tabela oznacza, że jeśli rozmieścimy warstwy następująco:

Rozmieszczenie		Widzialność	
Warstwa	Piętro	Skala	Widzimy
A	24	0-120m	Warstwę A
B	22	200m-500m	... B
C	18	800m-8km	... C
PUSTA	17	12km	Naszą mapę

Warstwa PUSTA mówi przy jakiej skali przestajemy widzieć naszą mapę (a dokładnie pusty prostokąt o wymiarach naszej mapy) i przechodzimy np. na mapę bazową.

Poniżej kilka pytań, które pojawiły się po wypuszczeniu wersji 1.03, wraz z odpowiedziami:

1. Czy warstwa i piętro są z sobą ściśle związane?

Nie. To zależy od nagłówka np:

Level0=23

Level1=21

Level2=19

Level3=17

Level4=15

Warstwa0 -piętro 23

Warstwa1 -piętro 21

itd

Warstwa PUSTA -piętro 15

2. Czy File0= plik.plt jst zawsze umieszczone na piętrze 24?

Nie. Na warstwie 0, a piętro definiuje nagłówek w sekcji IMG

3. Dlaczego w tabelce Piętru 22 przypisano skalę 200-500 m, a powyżej Grid 22 ma przypisane 500 m?

Skala 200-500 jest przyporządkowana warstwie a nie piętru. Jeśli na jakimś piętrze nie umieszczamy żadnej warstwy (to robimy w nagłówku), to przy skali odpowiadającej temu piętru widzimy warstwę z najbliższego zapełnionego piętra poniżej.

4. Czy grid i piętro to to samo?

Nie. Ale są powiązane. Grid do dokładność z jaką możemy umieszczać elementy na danym piętrze.

5. W tabelce warstwie PUSTA odpowiada, że widzimy naszą mapę (ja rozumiem, że np. UMP), a z tekstu poniżej, że przechodzimy na mapę bazową - wygląda to na niekonsekwencję.

„Widzimy Naszą mapę” oznacza że widzimy pusty prostokąt o wymiarach naszej mapy

4. Dlaczego w nagłówku deklaruje się nie kolejne piętra, lecz co drugie, a mimo to kolejne warstwy są widoczne?

Chyba wyjaśniłem wyżej. Można wypełniać dowolne piętra ale na ogół nie warto. Zmiana dokładności dokonuje się na zdefiniowanych piętrach, a na pośrednich wyświetlane są informacje nie zmieniane

Załącznik 5 (Janusz)

Dwa programy używane do tworzenia map UMP, cGpsMapperper i Sendmap, są wywoływane z wiersza polecenia, co jest trochę niewygodne i może być dla niektórych użytkowników nieco onieśmielające. Opisana poniżej procedura pozwoli korzystać z tych programów przez przeciąganie myszą ikon, tak jak normalnie w Windows, bez jawnego wywoływania okna DOS.

1. Na Pulpicie (lub w innym wybranym folderze) tworzymy skrót do programów cGpsMapperper i Sendmap.
2. Klikamy prawym przyciskiem skrót cGpsMapperper, wybieramy Właściwości i edytujemy je następująco:
 - a) w polu {Element docelowy} dopisujemy na końcu opcje: `ac -p` w wyniku tego pole to wygląda następująco (uwaga na spację przed -p):
`".....\cGpsMapperper.exe" ac -p`
 gdzie oznacza ścieżkę (typowo C:\Program Files\GpsMapperper);
 - b) w polu {Rozpocznij w} wpisujemy katalog danych, w którym trzymamy nasze pliki map do kompilacji
3. Analogicznie, we właściwościach skrótu Sendmap:
 - a) w polu {Element docelowy} dopisujemy na końcu opcje: `COM1 -p` ; w wyniku tego pole to wygląda następująco:
`".....\Sendmap.exe" COM1 -p`
 (jeżeli do komunikacji z GPS używamy innego portu, to wpisujemy np. COM2);
 - b) w polu {Rozpocznij w} wpisujemy katalog danych jak wyżej.
4. Aby skompilować plik np. mapa.txt, przeciągamy myszą ikonę tego pliku i upuszczamy ją na skrót cGpsMapper.
5. Aby posłać plik np. mapa.img do odbiornika, przeciągamy myszą ikonę tego pliku i upuszczamy ją na skrót Sendmap.

Załącznik 6

Przykładowy wpis klucza rejestru. Jeżeli nie wiesz co to jest i do czego służy, lepiej się tym nie baw, bo możesz doprowadzić do reinstalacji swojego MS Windows

WARNING: Modifying the registry incorrectly can cause serious problems that may require you to reinstall your operating system. UMP team and personally me cannot guarantee that problems that result from incorrectly modifying the registry can be solved. You modify the registry at your own risk.

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Garmin\MapSource\Products\66]
"Loc"="E:\\Directory\\GpsMapper\\Test_Map"
"Bmap"="E:\\Directory\\GpsMapper\\Test_Map\\testmap.img"
"Tdb"="E:\\Directory\\GpsMapper\\Test_Map\\testmap.tdb"
```

Załącznik 7

Przykładowy plik generujący pliki img i tdb dla MapSource. **Ważne:** Jeżeli zmienisz wartość parametru FileName i ProductCode musisz odpowiednio zmienić treść wpisów do rejestru w opisie w rozdziale 5b (str. 18)

```
[Map]
FileName=mg_ump
MapVersion=100
ProductCode=99

Levels=5
Level0=23
Level1=21
Level2=19
Level3=16
Level4=10

Zoom0=1
Zoom1=2
Zoom2=4
Zoom3=5
Zoom4=7

MapsourceName=UMP
MapSetName=UMP-pl.rec.gps
CDSetName=Testy
Copy1=
Copy2=
[End-Map]

[Files]
img=c:\Program Files\MapSource\MG_UMP\04300224.img
img=c:\Program Files\MapSource\MG_UMP\04300325.img
[END-Files]
```

Podziękowania / Credits

Dziękuję za pomoc przy tworzeniu niniejszej dokumentacji oraz zaangażowanie przy powstawaniu pierwszych wersji UMP, walce z małym (wówczas) przyjaznym środowiskiem i poświęcaniem części swojego czasu innym.

Wsparcie merytoryczne:

GpsMapperper (Staszek) gps@gps.chrisb.org <http://gps.chrisb.org/>
autor narzędzi pakietu GpsMapper

JaC <J.M.> jac@plusnet.pl Klepnij, pajacyka w brzuszek
www.pajacyk.pl , to mu pomaga

JurekR(yll) ryll@artgraph.com.pl

p0li p0li@poczta.onet.pl

pleśniak gps@golf.pl

Wsparcie edytorskie:

Janusz S.

Inne uwagi:

Pliki UMP zawierają informacje o osobach tworzących mapy. Jeżeli udostępnisz swoje mapy osobie koordynującej projekt UMP (p0li), trafisz na tę listę ☺.