

Paweł Pędzich
Krzysztof Cybulski

Instytut Fotogrametrii i Kartografii
Politechnika Warszawska
(00-661 Warszawa, pl. Politechniki 1)

Próba wykorzystania Digital Cartographic Studio (DCS) do tworzenia grafiki mapy *

W artykule opisano możliwości wykorzystania oprogramowania firmy Intergraph o nazwie Digital Cartographic Studio (DCS) w procesie redakcji mapy na podstawie danych przechowywanych w systemie informacji geograficznej. Artykuł składa się z dwóch części, pierwsza zawiera opis zasad tworzenia znaków umownych, w drugiej przedstawiono zasady tworzenia i używania w DCS reguł stosowanych przy opracowaniu grafiki mapy.

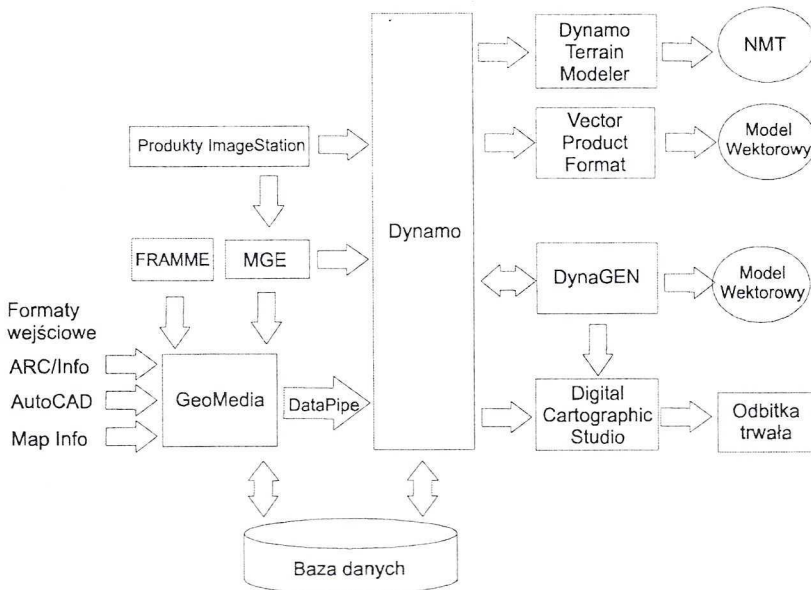
Działanie DCS oparte jest na środowisku Dynamo, który jako jedyny system GIS pozwala na zachowywanie ciągłej topologii obiektów. Jest to możliwe dzięki zastosowaniu w Dynamo formatu pliku o rozszerzeniu *.os (object space). W pliku tym przechowywana jest zarówno geometria jak i atrybuty obiektów. Do środowiska tego można importować gotowe dane z istniejących projektów MGE (geometria i atrybuty), MicroStation (geometria i atrybuty, przy czym atrybutami są numer warstwy, barwa i styl obiektów) lub za pośrednictwem środowiska GeoMedia z innych systemów GIS. Środowisko Dynamo posiada też liczne rozszerzenia pozwalające na generalizację danych, ilościową i jakościową z użyciem modułu DynaGen, obróbkę numerycznego modelu terenu z wykorzystaniem Terrain Modeler lub przetworzenie danych do formatu VPF (Vector Product Format) z użyciem VPFP (Vector Product Format Production System).

Środowisko DCS zbudowane jest z następujących modułów:

– DCS Administrator – moduł przeznaczony do tworzenia biblioteki znaków i reguł symbolizacji.

* Redakcja udostępnia polskim czytelnikom technologie DCS jako przykład możliwości warsztatowego tworzenia grafiki mapy w sposób elastyczny i efektywny

- DCS Cartographer – moduł przeznaczony do tworzenia grafiki mapy w oparciu o dane szkieletowe przechowywane w pliku *.os. Wykorzystuje się do tego celu bibliotekę znaków i reguły zdefiniowane przez użytkownika. W wyniku tych działań otrzymujemy plik z rozszerzeniem *.dcs, który można poddawać dalszej obróbce tak jak plik *.os. W module tym istnieje możliwość jednostkowej edycji obiektów i znaków umownych.
- DCS Typographer – moduł, który na podstawie reguł użytkownika automatycznie umieszcza etykiety i napisy dla obiektów punktowych, liniowych i powierzchniowych. Aplikacja pozwala na wykrycie nakładania się tekstu na znaki i automatyczne przemieszczanie tekstu. Treść opisu można tworzyć na podstawie wartości atrybutu obiektu.
- DCS Geodesist – moduł przeznaczony do tworzenia siatki kartograficznej i kilometrowej. Można określić styl linii dla każdej z siatek.
- DCS Marginalia – pozwala na automatyczne tworzenie marginałów mapy.



Rys. 1. Miejsce DCS w środowisku Dynamo

DCS jest bardzo rozbudowanym oprogramowaniem przeznaczonym do celów kartograficznych. DCS przyspiesza proces produkcji mapy z danych w formie szkieletowej przechowywanych w systemach informacji przestrzennej, dzięki dużym możliwościom tworzenia znaków, automatycznego ich łączenia z odpowiednimi obiektami i dużym możliwościom rozwiązywania tzw. konfliktów, polegających głównie na nakładaniu się na siebie znaków kartograficznych. Zasady łączenia obiektu ze znakiem, wykrywania konfliktów i ich rozwiązywania zawarte są w tzw. regułach symbolizacji tworzonych przez

użytkownika w specjalnym języku programowania DCS. W dalszej części artykułu zostaną przedstawione zasady budowania biblioteki znaków i tworzenia reguł symbolizacji.

1. Tworzenie znaków umownych

Na mapie spotykamy się z następującymi typami znaków:

- punktowe, reprezentują obiekty rzeczywistości postrzegane jako punktowe (np. punkty osnowy geodezyjnej) lub obiekty, które ze względu na rozmiary nie dają się przedstawić w skali mapy (np.: sygnatury budynków i budowli lub obiektów gospodarczych);
- liniowe, reprezentują obiekty o charakterze liniowym jak: drogi, koleje, linie przesyłowe, wąskie cieki wodne, granice;
- powierzchniowe, reprezentują obiekty mające charakter powierzchniowy, dające się przedstawić w skali mapy (np.: roślinność, uprawy, grunty, budynki i budowle, drogi, wody).

Oprócz wyżej wymienionych typów znaków nierozłącznie związane z mapą są napisy, które przy automatycznej symbolizacji bazy danych powinny być postrzegane jako czwarty typ znaku. W znaku tym zmienia się treść a pozostałe parametry jak justowanie, odstęp pomiędzy liniami, czcionka jej styl, rozmiar pozostają stałe.

Grafikę znaku można rozbić na elementy składowe, moduły graficzne – components. Pozwala to na wykorzystanie tych samych modułów przy budowaniu różnych znaków. Przykładem użycia wielokrotnego tego samego modułu jest znak linii kolejowej, którą można rozbić na dwa moduły liniowe: linia czarna o grubości 0,7 mm, linia biała przerywana o grubości 0,5 mm, długość przerwy 10,0 mm, długość elementu białego 5,0 mm z nałożonymi na obie linie modułami punktowymi – kreskami jedną, dwiema lub trzema w zależności od liczby torów. Nakładając na siebie właściwe moduły w odpowiedniej kolejności otrzymujemy żądane znaki linii kolejowej.

1.1 Moduły graficzne

Do budowy znaków wykorzystuje się następujące rodzaje modułów graficznych:

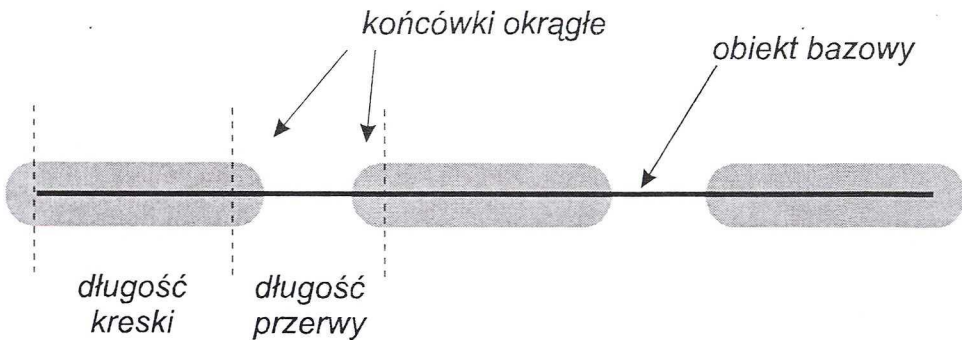
- linia – line
- powierzchnia – area
- kontur znaku – point lines
- wypełnienie znaku – point areas
- tekst – text
- deseń powierzchni – area pattern
- deseń linii – line pattern
- równoległe przesunięcie linii – offset line generator
- generator linii ze znaków – line point generator

1.1.1 Linia

Moduł graficzny liniowy jest używany do budowania znaków liniowych i konturów obiektów powierzchniowych. Dla modułu określa się barwę, szerokość i rodzaj linii. Barwa modułu wybierana jest spośród barw dostępnych w *Tabeli Barw*. Tabelę tę należy zbudować dla potrzeb konkretnej mapy tak by zawierała wszystkie możliwe barwy występujące na tworzonej mapie. Istnieje też możliwość importu tabeli ze środowiska MicroStation. Szerokość linii jest wyrażana w milimetrach, przy czym możliwa jest budowa linii o zmiennej szerokości. W takim przypadku należy podać szerokość początkową, końcową i długość, na jakiej ma nastąpić zmiana. Poprzez rodzaj linii definiujemy czy linia jest ciągła, czy przerywana. Dla linii przerywanej buduje się cyklicznie powtarzany moduł poprzez określenie długości następujących po sobie elementów kreskowych i przerw. Kolejne parametry określają sposób przedstawienia początku i końca całej linii i dla linii przerywanej zakończenie elementów kreski od strony przerwy. Możliwe zakończenia linii są następujące:

- okrągłe,
- przedłużone ścięte,
- ścięte,
- docięte (tylko dla początku i końca linii).

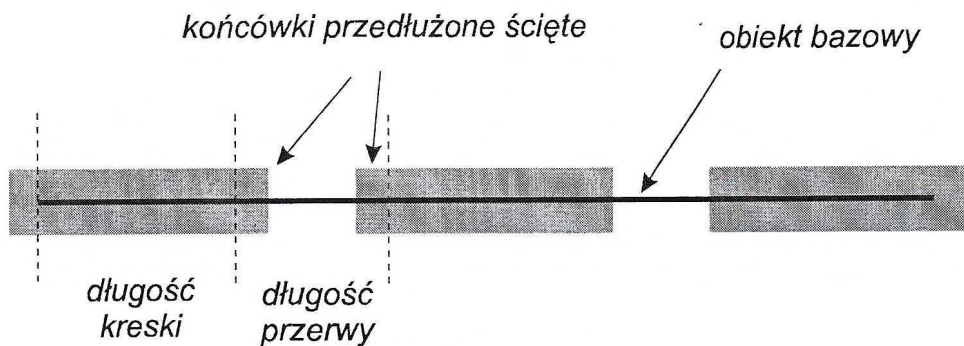
Zakończenie okrągłe powstaje w wyniku dorysowania na końcu kreski okręgu o promieniu równym połowie szerokości linii, w wyniku czego długość kreski wydłuża się. Zakończenie takie przedstawione jest na rys. 2.



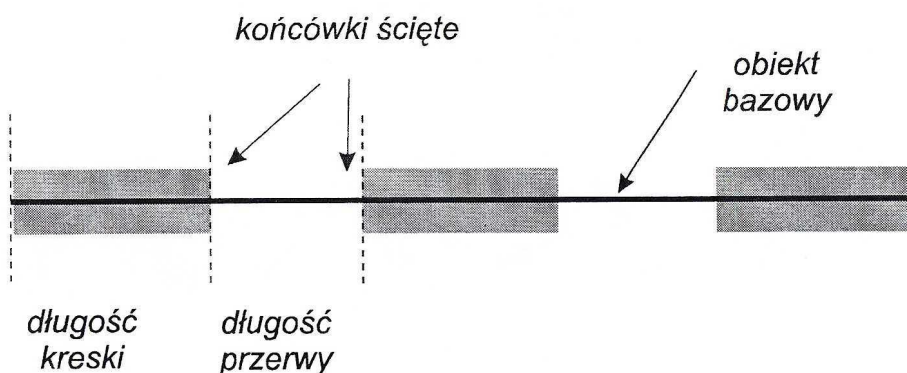
Rys. 2. Przykład okrągłego zakończenia kreski

Zakończenie przedłużone ścięte jest kolejnym zakończeniem, które zmienia rzeczywistą długość kreski i polega na przedłużeniu kreski o długość równą połowie szerokości linii. Zakończenie to przedstawione jest na rys. 3.

Zakończenie ścięte jest najprostszym typem zakończenia kreski, w którym kreska nie jest przedłużana. Wygląd tego typu końcówki przedstawiony jest na rys. 4.

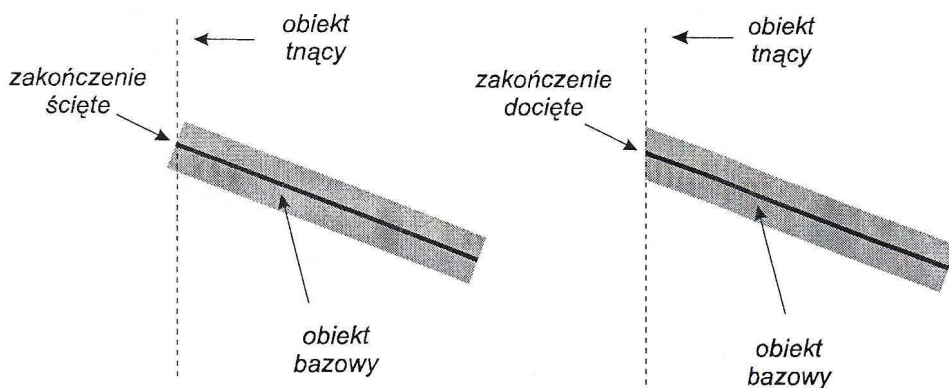


Rys. 3. Przykład przedłużonego ściętego zakończenia kreski



Rys. 4. Przykład ściętego zakończenia kreski

Zakończenie docięte jest zakończeniem, które może być zastosowane jedynie na końcu i początku całej linii. Polega ono na docięciu końcówki linii do elementu, z którym dana linia się styka. W wyniku takiego sposobu przedstawienia zakończenia unikniemy sytuacji, w której dana linia będzie kończyła się za wcześnie i wtedy powstanie przerwa lub będzie przeciągnięta za linię docelową np. ramkę mapy. Zakończenie to przedstawione jest na rys. 5.



Rys. 5. Przykład dociętego zakończenia kreski

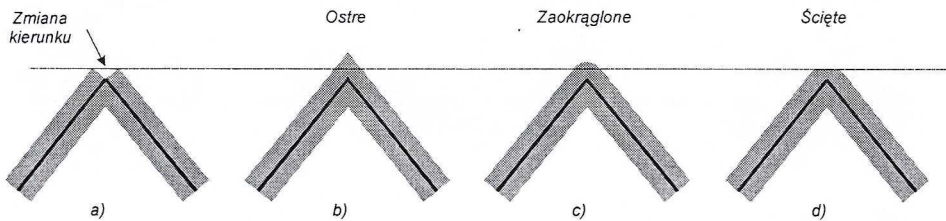
Poza określeniem sposobu zakończenia linii można określić sposób łączenia odcinków linii łamanej przy zmianie kierunku linii. Dostępne są trzy sposoby przedstawienia połączenia:

- ostre,
- zaokrąglone,
- ścięte.

Ostre połączenie (rys. 6b) polega na przedłużeniu zewnętrznych krawędzi linii do punktu przecięcia, przy czym przy małym kącie wewnętrznym połączenia takiego nie uzyskamy.

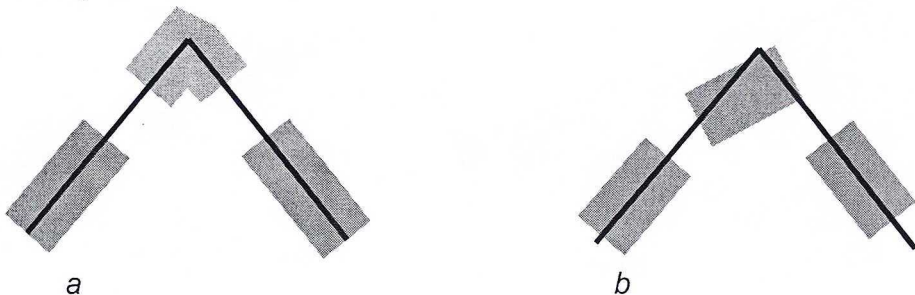
Zaokrąglone połączenie (rys. 6c) jest połączeniem, w którym w punkcie zmiany kierunku rysowany jest okrąg.

Ścięte połączenie (rys. 6d) jest połączeniem, w którym obcinany jest wierzchołek występujący przy połączeniu ostrym.



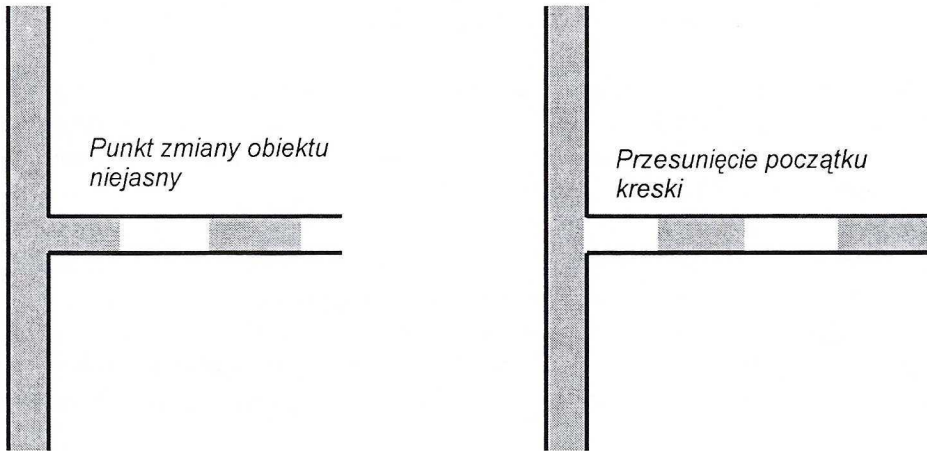
Rys. 6 Przykłady przedstawienia połączeń linii, a) bez żadnych zmian b) połączenie ostre c) połączenie zaokrąglone d) połączenie ścięte

W przypadku szerokiej przerywanej linii wyżej opisane sposoby graficznego przedstawiania zmiany kierunków linii łamanej mogą nie dać estetycznych rozwiązań. Szczególnie, gdy obiekt posiada wiele załamań może się zdarzyć, że końcówka z jednej strony załamania jest krótka. Sytuacja taka przedstawiona jest na rysunku 6a. Rozwiązaniem jest rysowanie kresek linii przerywanej bez ich załamywania z jednoczesnym małym przesunięciem położenia znaku w stosunku do położenia linii bazowej. Na rys. 7 zestawiono sposób przedstawienia linii bez zmiany położenia elementów tworzących 7a i ze zmianą 7b.



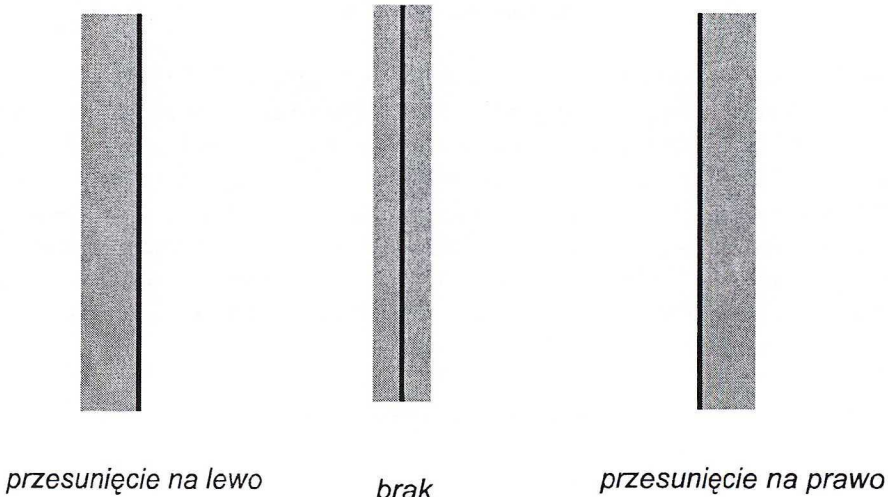
Rys. 7. Przykład zmiany położenia elementów linii przy zmianie kierunku

Kolejną ciekawą właściwością modułu liniowego jest możliwość przesunięcia początku kreski dla linii przerywanej na określoną odległość (rys. 8). Przykładowo, jeżeli moduł liniowy zbudowany jest z elementów kreski o długości 10 mm i przerwy o długości 10 mm, ustawienie parametru na 5 mm spowoduje, że linia będzie zaczynała się przerwą o długości 5 mm.



Rys. 8. Przykład przesunięcia początku kreski

Ostatni parametr możliwy do ustawienia we właściwościach modułu liniowego nosi nazwę *przesunięcie* – offset i daje on możliwość rysowania znaku na prawo, lewo lub centralnie w stosunku do obiektu bazowego (rys. 9).



Rys. 9. Ilustracja parametru przesunięcie

1.1.2. Powierzchnia

Moduł graficzny powierzchniowy odpowiada za wypełnienie obiektów powierzchniowych i w porównaniu z modułem liniowym właściwości jego są bardzo proste, ponieważ wybieramy tylko albo barwę wypełnienia (barwę jednolitą) albo raster procentowy danej barwy.

1.1.3. Kontur znaku

Moduł graficzny wykorzystywany przy budowie znaków punktowych odpowiada za kontur znaku. Aby go zbudować należy znać geometrię znaku i ustawić parametry linii konturu, takie same jak dla modułu liniowego. Dodatkowo na etapie budowania znaku można przeskalować i zmienić orientację geometrii.

Geometrię określającą kształt znaku możemy stworzyć bezpośrednio w DCS Administratorze lub „zaimportować” w formie szkieletowej z komórki MicroStation. Import gotowej biblioteki znaków z MicroStation jest mniej czasochłonny niż tworzenie jej w środowisku Dynamo. Ostatnią czynnością przed wykorzystaniem nowo utworzonej geometrii do budowy modułów punktowych jest określenie punktu głównego znaku, w odniesieniu, do którego rozwijany będzie znak przy symbolizacji obiektów punktowych.

1.1.4. Wypełnienie znaku

Moduł ten buduje wypełnienie znaku punktowego i podobnie jak dla konturu znaku przy budowie tego modułu wykorzystuje się gotową geometrię znaku, dla której określa się barwę wypełnienia.

1.1.5. Generator linii ze znaków

Moduł za pomocą, którego znaki punktowe umieszczane są wzdłuż obiektu liniowego według określonego wzorca. Odstęp między kolejnymi znakami określone są przez parametry linii przerywanej (długość kreski i przerwy). Dodatkowo istnieje szereg parametrów, które pozwalają na orientację i położenie znaku względem pojedynczej kreski i przerwy. Parametry te pozwalają na umiejscowienie znaku na początku, w środku lub na końcu kreski i przerwy. Można też orientować znak prostopadłe do obiektu liniowego lub zmieniać orientację znaku o zadaną wartość kąta w stosunku do obiektu bazowego. Dostępne są też parametry, które wymuszają pojawienie się znaku na początku i/lub na końcu linii lub w punkcie zmiany kierunku linii.

1.1.6. Równoległe przesunięcie linii

Moduł wykorzystywany do równoległego przemieszczenia znaku liniowego w stosunku do odpowiadającego mu obiektu bazowego. Przesunięcie jest wykonywane na określoną odległość i w zadanym kierunku. Dodatkowo można zdefiniować typ połączenia przy

zmianie kierunku linii. Moduł ten nie odpowiada parametrowi offset definiowanemu przy module liniowym. Przy definiowaniu modułu *równoległe przesunięcie linii* jako element wejściowy wymagany jest moduł liniowy.

1.1.7. Deseń powierzchni

Do symbolizacji obiektów powierzchniowych z deseniem jest wykorzystywany moduł w określonej barwie. Do utworzenia tego modułu potrzebna jest geometria desenia.

1.1.8. Tekst

Utworzenie tego modułu niezbędne jest, aby móc umieszczać wszelkiego rodzaju napisy na mapie. Aby można było użyć danej czcionki w DCS należy ją wcześniej „zarejestrować” w programie. W czasie „rejestracji” czcionki określa się jej krój, rozmiar oraz charakter stosowanych liter (np. duże, małe litery, znaki specjalne). Czcionki z przynajmniej jednym różnym parametrem traktowane są jako różne. Można rejestrować czcionki typy TrueType 8 i 16 – bitowe, przy czym muszą być to czcionki dostępne na poziomie systemu operacyjnego. W jednym module tekstowym można zarejestrować do ośmiu różnych czcionek. Na poziomie tego modułu określamy barwę, sposób justowania – który określa punkt główny napisu, odstęp między liniami. Możliwa jest również zmiana położenia napisu poprzez podanie przesunięcia i kąta obrotu.

Możliwe jest stworzenie trzech odmian modułu tekstowego dla:

- napisów o treści z góry określonej, które mogą być wykorzystywane np. dla elementów legendy;
- napisów, których treść będzie wnoszona przez operatora w chwili ich umieszczania na arkuszu mapy lub będzie automatycznie generowana z atrybutów z bazy danych w module DCS Typographer;
- napisy mieszane, które mają część stałą a część jest wnoszona interaktywnie przez operatora.

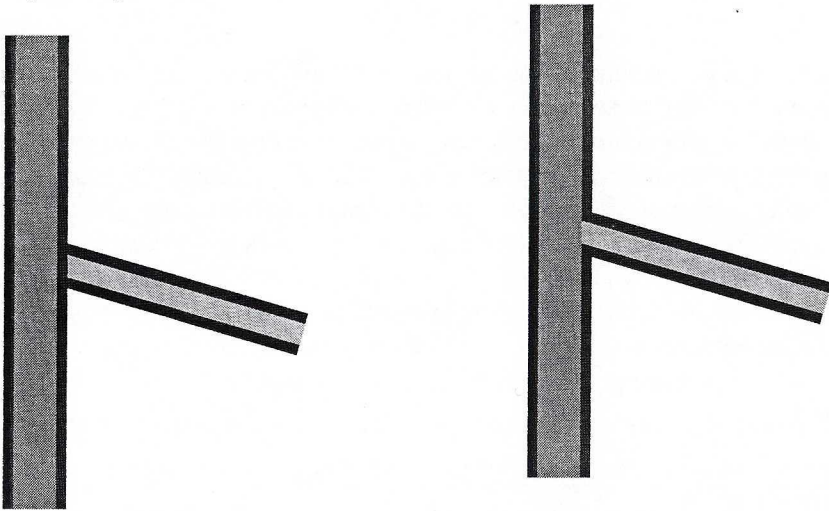
1.2. Budowanie znaków z modułów graficznych

1.2.1. Hierarchia modułów

Budowanie znaku polega na wskazywaniu kolejnych modułów graficznych, które go tworzą, z jednoczesnym określeniem dla każdego modułu priorytetu wyświetlania. Parametr ten pozwala na ustawienie kolejności, w jakiej moduły będą wyświetlane i drukowane. Moduły z niższą wartością parametru będą przykrywane przez te z wyższą, zarówno w obrębie znaku jak i całej biblioteki symboli. Najniżej w hierarchii wyświetlania leżą znaki przedstawiające obiekty przyrodnicze (znaki przedstawiające lasy, łąki, wody). Przykrywane są one znakami rzeźby terenu, które z kolei znikają pod znakami obiektów

antropogenicznych. Najwyżej w hierarchii znajdują się nazwy i opisy objaśniające. Oprócz kontroli globalnej hierarchii można kontrolować takie szczegóły jak np. sposób łączenia dróg różnych kategorii, co przedstawiono na rys. 10.

Podczas tworzenia arkusza mapy mogą się jednak zdarzyć sytuacje, gdy globalna hierarchia modułów nie spełni swojego zadania. Przykładem może być most dwupoziomowy drogowo-kolejowy, na którym w zależności od tego czy droga leży nad koleją czy kolej nad drogę na arkuszu mapy powinna znaleźć się odpowiednio droga lub kolej. Rozwiązaniem takiej sytuacji jest możliwość ręcznej edycji hierarchii wybranych modułów przez operatora.



Rys. 10. Przykład różnego przedstawienia skrzyżowania uzyskany dzięki różnym ustawieniom parametru priorytet poszczególnych modułów graficznych

1.2.2. Znaki liniowe

Do budowy liniowych znaków umownych możemy wykorzystać następujące komponenty:

- linia,
- równoległe przesunięcie linii,
- generator linii ze znaków.

Na polskiej mapie topograficznej w skali 1:10 000 najwięcej znaków liniowych budowanych jest w oparciu o moduły liniowe w połączeniu z modułami typu *generator linii ze znaków*. W skład tej grupy wchodzi znak: linii kolejowych szeroko- i normalnotorowych, linii tramwajowych, kolei wiszącej, wyciągu narciarskiego, nasypu, wykopu, ściany oporowej, ogrodzenia trwałego, muru historycznego, suwnicy, linii energetycznych, rurociągów naziemnych, linii telefonicznej, brzegu umocnionego i urwistego, jazu stałego, falochronu, wału, grobli, terenów rozkopanych. W przeważającej liczbie przypadku *generator linii z symboli* tworzy linie prostopadłe lub skośne do osi obiektu liniowego.

Drugą pod względem liczby znaków jest kategoria znaków budowanych jedynie z modułów liniowych. Do grupy tej zaliczają się znaki: linii kolejowej wąskotorowej, toru stacyjnego, dróg wszystkich kategorii, szlaków turystycznych, cieku stałego o szerokości 1 m i mniejszej, i o szerokości 5 m i mniejszej, cieku okresowego i suchego rowu, poziomic, (jeżeli kreski spadu będą stanowiły osobny znak), taśmociągu.

1.2.3. Znaki powierzchniowe

Znaki powierzchniowe składają się z modułów graficznych powierzchniowych, *powierzchnia* i *deseń powierzchni*, które odpowiadają za wypełnienie obiektu powierzchniowego i/lub z modułów liniowych, które odpowiadają za przedstawienie linii konturu obiektu powierzchniowego. Budowanie znaków obiektów powierzchniowych jest łatwym zadaniem.

1.2.4. Znaki punktowe

Znaki budowane są na bazie dwóch modułów graficznych: *kontur symbolu* i *wypełnienie symbolu*. W polskiej edycji mapy topograficznej 1:10 000 drugi z wymienionych komponentów może być użyty przy składaniu kilku znaków, z których najciekawszym jest znak oczyszczalni ścieków, z powodu występowania wypełnienia na części znaku. Pozostałe znaki to znaki budynku i szklarni nie dających się przedstawić w skali mapy i znaki z czarnym wypełnieniem jak np. latarnia, głąz narzutowy.

2. Zasady redakcji mapy w Digital Cartographic Studio

Redagowanie treści mapy opiera się na pewnych zasadach, które określają m.in.:

- jakim znakiem ma być przedstawiony dany obiekt terenowy,
- minimalną odległość pomiędzy rysowanymi znakami,
- zmianę położenia znaku w przypadku, gdy minimalna odległość pomiędzy znakami zostanie naruszona,
- który ze znaków ma zostać zmieniony i w jaki sposób, gdy występuje ich nakładanie się.

Rozwiązywanie tego typu problemów tradycyjnymi metodami i narzędziami jest zadaniem bardzo pracochłonnym i trudnym. Digital Cartographic Studio daje możliwości zautomatyzowania tego typu zadań poprzez wykorzystanie tzw. *reguł symbolizacji* (symbolization rules). Reguły symbolizacji są to zasady redakcji mapy zapisane przez użytkownika w specjalnie do tego celu stworzonym języku programowania DCS. W szczególności w DCS wyróżniono następujące typy reguł: symbolizacja klas obiektów bazowych, czyli przypisanie odpowiedniego znaku klasom obiektów występujących w bazie danych, obrót znaku, zmiana orientacji jednego znaku względem drugiego, zmiana barwy znaku, wykrywanie i rozwiązywanie konfliktów. Wykrywanie konfliktów oznacza tu wyszukiwanie niedopuszczalnych relacji pomiędzy dwoma obiektami lub pomiędzy dwoma znakami (np. nakładanie się dwóch znaków). Natomiast rozwiązywanie konfliktów

oznacza użycie odpowiedniej metody eliminującej tego typu błędy (np. usunięcie znaku lub jego części, modyfikacja znaku, przesunięcie znaku). Poszczególne typy reguł zostaną szerzej omówione w dalszej części artykułu.

Tworzenie reguł w DCS przypomina pisanie programów komputerowych. Do pisania reguł można wykorzystać dowolny edytor tekstu. Składnia reguł symbolizacji jest zbliżona do języka naturalnego.

2.1. Składnia reguł symbolizacji

Podstawowa struktura reguły ma następującą postać:

If

(warunek)

Then

(akcja)

Spełnienie warunku oznacza wykonanie akcji.

Najprostszym warunkiem jest sprawdzenie czy dana klasa obiektów istnieje w bazie danych. Wykonanie reguły powoduje przeszukanie i sprawdzenie każdej klasy obiektów występującej w bazie danych. Składnia warunku istnienia może mieć następującą postać:

If

a ROAD_L exists

Then

(akcja)

ROAD_L jest przykładową klasą obiektów liniowych.

W sekcji warunek można również sprawdzać wartości atrybutów obiektów. Dostępne są następujące operatory:

equal to (=)

greater than (>)

less than (<)

greater than or equal to (>=)

less than or equal to (<=)

not equal to (<>).

Dopuszczalne jest użycie powyższych operatorów w dwóch postaciach, np. „>” lub „greater than”.

Przykładowy warunek dotyczący atrybutów może mieć następującą postać:

If

a ROAD_L has a WID equal to 6

Then

(akcja)

Wyrażenie WID jest atrybutem obiektu ROAD_L.

Można również łączyć warunki za pomocą operatorów „or” lub „and”. Przykładowy warunek tego typu może mieć następującą składnię:

If

```
a ROAD_L has a WID equal to 6 or 8 or
a LTN greater than or equal to 2 and
a WTC equal to „All Weather” Then
(akcja)
```

Jeżeli warunek jest spełniony wykonywane są działania umieszczone po słowie kluczowym „then”, czyli tzw. „akcja”. Działania te polegają przede wszystkim na doborze odpowiednich znaków, zmianie orientacji znaków, modyfikacji znaku itp. DCS dostarcza szereg funkcji, które pozwalają na automatyczne wykonywanie tych czynności. Dużym ułatwieniem jest możliwość wykorzystywania zamiast oryginalnych nazw funkcji tzw. synonimów. Można tworzyć własne nazwy funkcji i potem je używać pisząc reguły. Synonimy są definiowane w nagłówku pliku, w którym zapisane są reguły. Umieszcza się je po słowie kluczowym SYNONYMS, np:

```
SYNONYMS: „initialize”      = „GSY7C_initialize_transaction”
           „assign_symbol”   = „GSY7C_associate_sd”
           „symbolize”      = „GSY7C_symbolize_feature_component”
           „rotate_sd”      = „GSY7C_rotate_symbol”
```

W powyższym przykładzie po lewej stronie operatorów „=” znajdują się synonimy, a po prawej odpowiadające im nazwy funkcji używane przez DCS.

Najczęściej używanymi funkcjami są: „GSY7C_initialize_transaction”, „GSY7C_associate_sd”, „GSY7C_symbolize_feature_component”.

Te trzy funkcje powodują przypisanie odpowiedniego znaku do obiektu z bazy danych. Przykładowa reguła wykorzystująca te funkcje może mieć następującą postać:

R: roads

If

```
a ROAD_L has a WTC equal to „All Weather” and
a RST equal to „Loose/Light”
then
initialize the ROAD_L using 0
assign_symbol the ROAD_L using „AWLS_RD”
symbolize the ROAD_L
```

Po słowie kluczowym „Rule” (lub krótko „R”) występuje nazwa reguły.

Synonim „initialize” przywołuje funkcję „GSY7C_initialize_transaction” dla obiektu ROAD_L. Funkcja ta powoduje czyszczenie pamięci programu dla zdarzenia lub grupy zdarzeń związanych z obiektem ROAD_L. Wyrażenie „using 0” oznacza, że nie wprowadza się dodatkowych argumentów dla funkcji „initialize”.

Synonim „assign symbol” przywołuje funkcję „GSY7C_associate_sd”, która łączy klasę bazową z odpowiednim znakiem, w przykładzie nazwa znaku to „AWLS_RD”. Nazwa jest identyfikatorem znaku i musi być zgodna z nazwą występującą w bibliotece znaków DCS.

Synonim „symbolize” przywołuje metodę „GSY7C_symbolize_feature_component”, która łączy odpowiedni znak z każdym obiektem występującym w bazie danych.

2.2. Przykłady reguł symbolizacji

Wśród reguł symbolizacji można wyróżnić reguły powodujące

- obrót znaku o określony kąt,
- zmianę barwy znaku,
- zmianę orientacji znaku względem innych znaków,
- oraz grupę reguł powodujących tzw. wykrywanie i rozwiązywanie konfliktów.

2.2.1. Obrót znaku

W przypadku reguł powodujących obrót znaku, wartość kąta obrotu może być odczytana z bazy danych, jeżeli występuje jako atrybut obiektu. Przykładowa reguła powodująca obrót znaków punktowych reprezentujących budynki może mieć następującą postać:

```
If
  a BUILDING_P exists
Then
  initialize the BUILDING_P using 0
  assign_symbol the BUILDING_P using „POINT_BUILDING”
  build a text_string CELL_ROTATION of the BUILDING_P
  rotate_sd the BUILDING_P using text_string
  symbolize the BUILDING_P
```

W regule tej wyrażenie „build a text_string” powoduje zamianę wartości numerycznej atrybutu CELL_ROTATION w łańcuch znaków. Synonim „rotate_sd” powoduje wywołanie odpowiedniej funkcji powodującej obrót znaku.

Można także bezpośrednio podać wartość kąta obrotu wpisując ją w miejsce wyrażenia „tekst_string”.

2.2.2. Zmiana barwy znaku

Reguła zmiany barwy znaku może być użyteczna wówczas, gdy występują znaki o tym samym kształcie różniące się jedynie barwą. Nie trzeba wówczas tworzyć dwóch znaków. Wystarczy napisanie odpowiedniej reguły powodującej nadanie odpowiedniej barwy dla odpowiedniego znaku. Reguła zmiany barwy znaku punktowego budynku może mieć następującą postać:

```

If
  a BUILDING_P exists
Then
  initialize the BUILDING_P using 0
  assign_symbol the BUILDING_P using „POINT_BUILDING”
  set_color the BUILDING_P using 10000, „POINT_BUILDING”, „125”
  symbolize the BUILDING_P

```

Synonim „set_color” wywołuje odpowiednią funkcję powodującą zmianę barwy obiektu.

Wyrażenie „using 10000” oznacza, że nastąpi zmiana barwy całego znaku (możliwa jest zmiana poszczególnych modułów tworzących znak). Liczba „125” oznacza kod koloru. Jest to wielkość definiowana przez użytkownika w plikach zawierających tabele barw używanych przez DCS do wyświetlania lub do wydruku mapy.

2.2.3. Orientacja znaku

Reguły tego typu ustalają orientację jednego znaku względem drugiego.

Przykładowa reguła powodująca ustawienie znaku kierunku biegu rzeki równoległe do rzeki może mieć postać:

```

If
  a FLOW_P exists and the FLOW_P coalesces using „RIVER_L”, 0, 1, 0.5,
  „WHOLE”,
  „WHOLE” a set of RIVER_L
then
  send orient symbol to the FLOW_P using the set of RIVER_L,
  „FLOW_ARROW”, „WHOLE”, 0.0, 1, 1, 1

```

W powyższym przykładzie synonim „coalesces” wywołuje funkcję, która bada odległość jednego znaku względem drugiego, w tym wypadku minimalna odległość wynosi 0,5 mm. Poniżej tej wartości następuje wykonanie akcji. Dwie pozostałe wartości w warunku oznaczają geometrię znaku (0 – punkt, 1 – linia). Słowo „WHOLE” oznacza, że każda część znaku będzie rozpatrywana podczas badania warunku. Synonim „orient symbol” wywołuje funkcję, która powoduje zmianę orientacji znaku. Podstawowymi parametrami tej metody są kierunek, względem którego dokonuje się obrotu znaku oraz kąt orientacji.

Działanie powyższej reguły przedstawia rys. 11.



Rys. 11. Przykład działania reguły powodującej zmianę orientacji znaku

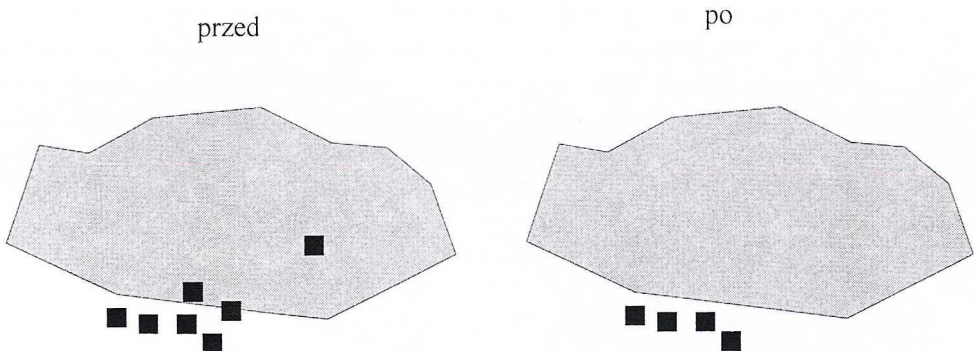
2.2.4. Wykrywanie i rozwiązywanie konfliktów

W tego typu regułach, w sekcji warunek, następuje wykrywanie konfliktów. Występują dwa podstawowe sposoby wykrywania konfliktów. Pierwszy z nich polega na pomiarze odległości pomiędzy znakami. W tym przypadku konflikt występuje gdy odległość pomiędzy symbolami jest mniejsza od założonej minimalnej odległości. Drugi sposób polega na określeniu relacji pomiędzy obiektami, np. częściowe lub całkowite pokrywanie się obiektów. Przykładem takiego konfliktu może tu być pokrywanie się granicy gminy z rzeką.

Po spełnieniu warunku następuje wywołanie odpowiedniej funkcji powodującej rozwiązanie danego typu konfliktu. Wyróżnia się cztery podstawowe sposoby rozwiązywania konfliktów:

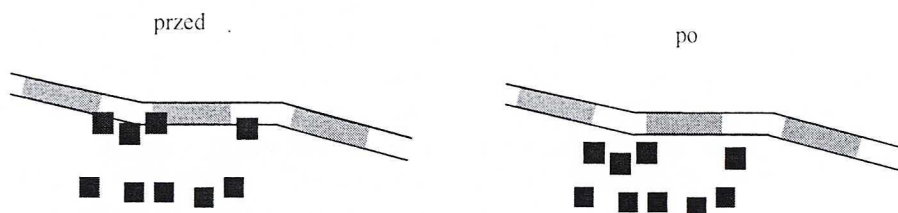
- usunięcie znaku,
- zmiana położenia znaku,
- modyfikacja znaku – resymbolization,
- umieszczenie obiektów w tzw. kolejce.

Pierwszy sposób polega na usunięciu całego znaku lub pojedynczego modułu tworzącego znak, używana jest wówczas funkcja „GSY8314_suppress_symbol”. Możliwe jest również usunięcie części znaku, w przypadku gdy dwa znaki pokrywają się ze sobą, np. usunięcie fragmentu znaku granicy działki wówczas gdy ten fragment pokrywa się ze znakiem rzeki. Do usuwania części znaku używa się funkcji „GSY8315_suppress_section”. Poniżej pokazano przykład działania reguły, w której następuje wykrywanie pojedynczych znaków budynków nachodzących na znak obszaru zabudowy. Znaki budynków będące w konflikcie zostały usunięte.



Rys. 12. Przykład działania reguły powodującej usunięcie znaku

Drugi sposób polega na odsunięciu jednego znaku od drugiego w przypadku gdy są one w konflikcie, np. gdy znak budynku pokrywa się z znakiem linii kolejowej, wówczas można



Rys. 13. Przykład działania reguły powodującej zmianę położenia znaku

spowodować, że nastąpi automatyczne odsunięcie znaków budynków. Wykorzystywana jest wówczas funkcja o nazwie „GSY8310_displace_symbol”.

Kolejny sposób polega na zmianie całego znaku lub jego części na inny, np. gdy granica obrębu administracyjnego pokrywa się z biegiem rzeki, można wówczas dokonać modyfikacji fragmentu znaku granicy znajdującego się w konflikcie, zwiększając w ten sposób czytelność mapy. Wykorzystywane funkcje to „GSY312_resymbolize” i „GSY8313_resymbolize_section”.



Rys. 14. Przykład działania reguły powodującej modyfikację znaku

Ostatni sposób rozwiązywania konfliktów polega na utworzeniu tzw. „kolejki”. Wówczas istnieje możliwość interaktywnego rozwiązywania zaistniałych problemów. Ten sposób jest szczególnie użyteczny w przypadkach nietypowych, pozwala na przejście kontroli nad rozwiązaniem danego konfliktu przez kartografa wykonującego prace redakcyjne.

3. Wykorzystanie DCS w procesie opracowania map topograficznych

Digital Cartographic Studio jest narzędziem, które w znacznym stopniu ułatwia i przyspiesza proces opracowywania map. Podstawową zaletą tego oprogramowania jest możliwość częściowego zautomatyzowania prac redakcyjnych, głównie dzięki możliwości tworzenia reguł symbolizacji w specjalnie stworzonym do tego celu języku programowania. W żadnym innym oprogramowaniu tego typu nie występuje taka możliwość.

Zalety DCS szczególnie dają się zauważyć w procesie opracowywania map topograficznych. Mamy wówczas do czynienia z dużą ilością arkuszy map, a więc możliwość wielokrotnego wykorzystania tych samych reguł symbolizacji jest tu szczególnie użyteczna. Poza tym mapy topograficzne są wykonywane według określonych zasad zawartych w instrukcjach i wytycznych technicznych. Przełożenie tych zasad na język DCS jest, jak to pokazano w artykule, dość łatwe. DCS posiada również rozbudowane narzędzia do tworzenia i modyfikowania znaków. W bibliotekach znaków DCS można przechowywać wszystkie typy znaków, zarówno punktowe, liniowe jak i powierzchniowe. Możliwości tworzenia znaków w DCS są znacznie większe niż w innych oprogramowaniach tego typu. W DCS możemy tworzyć w prosty sposób skomplikowane znaki linii kolejowych, nasypów i wykopów, linii energetycznych. Również łatwo daje się tworzyć znaki powierzchniowe o skomplikowanych deseniach.

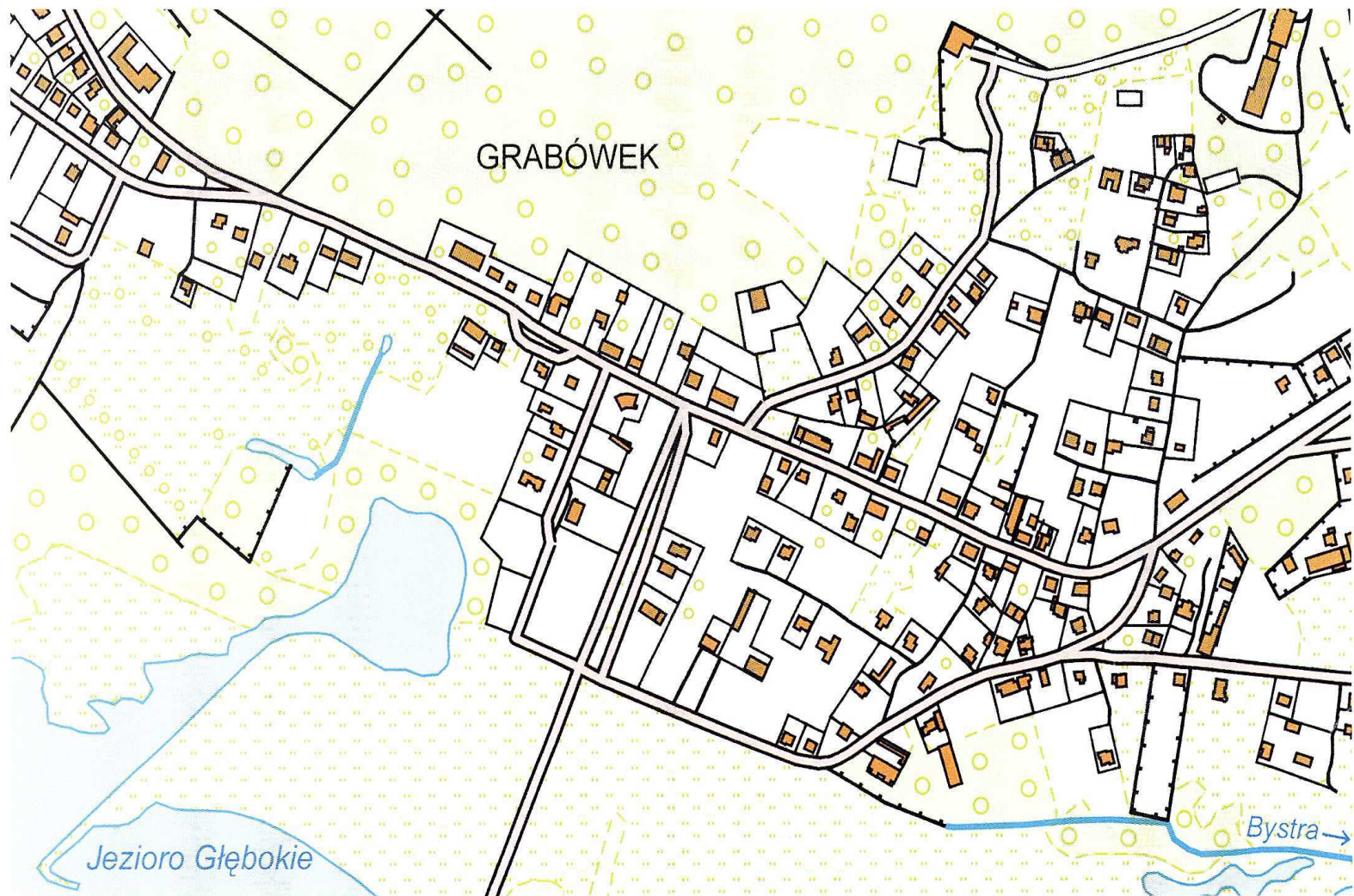
Na rysunku 15 zaprezentowano fragment mapy topograficznej w skali 1:10000 opracowanej w DCS z wykorzystaniem danych pochodzących z systemu informacji przestrzennej. W przykładzie zastosowano kilka podstawowych reguł symbolizacji. Otrzymana mapa wymaga jeszcze dalszych prac redakcyjnych.

Na zakończenie spróbujemy dokonać porównania procesu technologicznego wydania map w środowisku Dynamo i DCS z procesem opartym na oprogramowaniu z grupy MGE poczynając od MGE Finishera, przez InterPlot i na programie Map Publisher kończąc. Tworząc takie zestawienie można zauważyć, że w środowisku Dynamo i DCS tak jak w MGE Map Finisher możemy:

- generalizować elementy treści (MGE Map Finisher w stopniu dużo bardziej uproszczonym niż np. w DynaGen),
- wykonać finalną symbolizację (w Map Finisher musimy mieć biblioteki krojów pisma, symboli, deseni, linii, w DCS realizowane jest to za każdym razem, gdy wykonujemy symbolizację pliku),
- generować ramki, siatkę, opisy pozaramkowe, legendę mapy.

Analizując procesy realizowane przez InterPlot w połączeniu z Map Publisherem możemy zauważyć, że rasteryzacja produktów wektorowych odbywa się w oparciu o tabele obiektów a sposób ich rozwinięcia w odpowiednie symbole dokonuje się na podstawie tabeli deseni, barw, linii, czcionek. Następnie generowane są pliki wydawnicze do naświetlania diapozytywów w oprogramowaniu Map Publisher. Natomiast w środowisku DCS jest to realizowane poprzez definiowanie na samym początku biblioteki symboli i hierarchii wyświetlania poszczególnych modułów graficznych tworzących znaki. Rezultaty pracy DCS możemy zapisać zarówno do pliku rastrowego w standardzie tif jak i do pliku wykorzystywanego przez naświetlarki dzięki możliwości zdefiniowania w specjalnie do tego przeznaczonym pliku tzw. Color Resolution Table. W tabeli tej przypisuje się danej barwie używanej w środowisku DCS procent tonu, nazwę biblioteki siatki rastrowej i nazwę konkretnej siatki. Możliwe do wykorzystania są wszystkie dostępne biblioteki siatek rastrowych oferowane przez firmę Intergraph.

Natomiast to czego nie ma w procesie opartym na MGE, a występuje w DCS to możliwość symbolizacji obiektów w zależności od sąsiedztwa innych obiektów dzięki zastosowaniu reguł symbolizacji.



Rys. 15. Fragment mapy topograficznej opracowanej w skali 1 : 10000 wykonanej w DCS

LITERATURA

- [1] Daniels P., *Digital Cartographic Studio – product training course*, Materiały szkoleniowe firmy Intergraph, Huntsville, Alabama, 2002
[2] *Zasady redakcji mapy topograficznej w skali 1:10 000. Wzory znaków*, Instrukcja techniczna, 1998

Received December 15, 2002

Accepted July 15, 2003

Paweł Pędzich
Krzysztof Cybulski

An attempt to utilise Digital Cartographic Studio (DCS) for creation map graphics

S u m m a r y

The paper discusses possibilities to utilise Intergraph's software, the Digital Cartographic Studio (DCS) in the process of map editing, basing on data stored in a Geographic Information System. The paper consists of two parts. The first part discusses principles concerning creation of conventional signs. Principles concerning creation and utilisation of rules applied for development of map graphics in the DCS are discussed in the second part of the paper.

Павел Пендзих
Кышчтоф Цыбульски

Попытка использования Digital Cartographic Studio (DCS) для создания графики карты

Р е з ю м е

В статье представлены возможности использования состава программ Digital Cartographic Studio (DCS) фирмы Intergraph в процессе изготовления карты на основе данных собранных в системе географической информации. Статья состоит из двух частей. Первая часть содержит описание правил создания условных знаков, а во второй представлены правила создания и использования в DCS правил принимаемых в процессе обработки графики карты.