

## Zaawansowane metody przygotowania danych terenowych do symulatora szczebla taktycznego VBS2

J. KOSZELA, T. DROZDOWSKI, M. NIEDZIELA, H. OSTAP,  
R. WANTOCH-REKOWSKI  
jkoszela@wat.edu.pl

Wojskowa Akademia Techniczna, Wydział Cybernetyki,  
ul. Gen. Sylwestra Kaliskiego 2, 00-908 Warszawa

Artykuł zawiera opis procesu wytwarzania danych terenowych dla szczebla taktycznego VBS2. Zaprezentowano model terenu wykorzystywany w tym symulatorze oraz formaty danych terenowych, na podstawie których generowany może być wskazany teren. Następnie przedstawiono technologie i środowiska wykorzystane do generacji terenu ze wskazaniem najważniejszych właściwości. Zarówno w przypadku formatów danych (SHP, GDB, VPF, TDED), jak i narzędzi szczególną uwagę zwrócono na aspekt dostępności do specyfikacji dostępnych formatów danych. Artykuł kończy opis procesu generacji danych terenowych dla wybranego obszaru rzeczywistego na potrzeby VBS2.

**Słowa kluczowe:** VBS2, symulacja, dane terenowe.

### 1. Wprowadzenie

Jednym z aspektów przygotowania ćwiczeń wspomaganych komputerowo, także z wykorzystaniem symulatora VBS2, jest przygotowanie odpowiednich danych terenowych. Aspekt ten jest szczególnie ważny w przypadku symulatora VBS2, który znajduje zastosowanie na bardzo niskim szczeblu z wymaganiem na dużą dokładność zobrazowania obiektów (rysunek 1).



Rys. 1. Przykład zobrazowania w symulatorze VBS2

### 2. Podstawowe właściwości symulatora VBS2

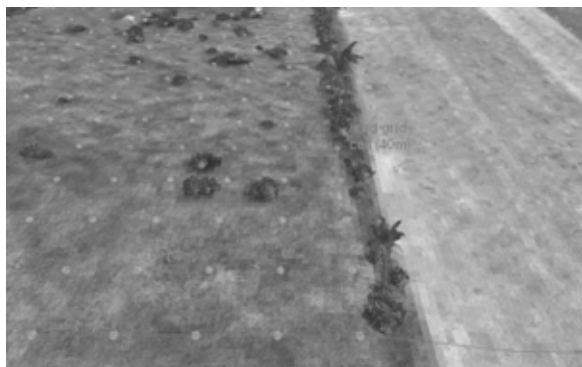
Symulator VBS2 umożliwia symulację działań pojedynczego żołnierza i pojazdu. VBS2 należy do klasy symulatorów wirtualnych z krokowym upływem czasu. Pozwala także na symulację większych grup: plutonów, kompanii. Ograniczenia w zakresie maksymalnego

szczebla symulowanych wojsk wynikają jedynie z ilości dostępnych zasobów dla symulatora, w szczególności pamięci operacyjnej. Symulacja wyglądem przypomina grę typu FPS (ang. *First-person shooter*), jednakże we wnętrzu tkwi zaawansowany silnik symulacyjny. Na wyniki symulacji mają między innymi wpływ takie aspekty jak pogoda, rodzaj użytej broni i amunicji, zmęczenie, wyszkolenie, morale, rozpoznanie. Symulator VBS2 umożliwia równoległy udział w symulacji wielu graczy, zarówno współdziałających ze sobą, jak i działających po przeciwnych stronach konfliktu. W trakcie trwania symulacji administrator może na bieżąco modyfikować scenariusz, dostosowując go do potrzeb szkoleniowych. Rozbudowane moduły edytora misji oraz analizy posymulacyjnej dopełniają funkcjonalność niezbędną do użycia VBS2 w szkoleniu. Dodatkowo VBS2 umożliwia rozbudowę swojej funkcjonalności poprzez dostępne API i bardzo elastyczny język skryptowy.

### 3. Podstawowe elementy modelu terenu symulatora VBS2

Symulator VBS2 generuje trójwymiarowe zobrazowanie symulacji, co wymaga od formatu zapisu terenu w nim wykorzystywanego niezwyklej szczegółowości, co docelowo ma zapewnić pozytywne wrażenia użytkowników symulatora. Teren w symulatorze VBS2 przechowywany jest w skompresowanych

plikach (.pbo). Zawartość pliku (.pbo) stanowią: plik z ukształtowaniem terenu (.wrp), tekstury (.png), pliki z opisem materiałów (.rvmat), modele 3D obiektów występujących na mapie (.p3d), skrypty i pliki konfiguracyjne (.lbt), (.cpp). Model terenu w VBS2 ma strukturę warstwową. Największą objętościowo i najważniejszą warstwą jest warstwa ukształtowania terenu, zawierająca siatkę punktów wysokościowych. Punkty siatki rozmieszczone są równomiernie, a odległość pomiędzy nimi określa dokładność odwzorowania terenu i determinuje wielkość zajętej pamięci w komputerze. Mapa VBS2 o rozmiarach 10 km x 10 km, z dokładnością 10 m, zajmuje około 50 MB pamięci.



Rys. 2. Fragment terenu w VBS2 z zaznaczonym podziałem na komórki

Na rysunku 2 widać punkty tworzące siatkę punktów wysokościowych, rozmieszczonych co 10 m. Na teren nałożona jest tekstura na podstawie zdjęcia satelitarne, które przy takiej dokładności mapy ma dokładność 1 texel/1 m. Na siatce punktów wysokościowych zbudowane są komórki terenu, które pełnią kluczową rolę w symulatorze, gdyż są wykorzystywane do wyznaczania dróg dla przemieszczających się obiektów. Służą również w procesie wyznaczania kolizji pomiędzy obiektami. Komórki te mogą mieć także przypisane dodatkowe atrybuty, jak np. muzyka tła.

Kolejną warstwą jest tekstura nałożona na cały teren. Wykorzystuje się ją głównie w sytuacji, kiedy obserwator znajduje się w dużej odległości, np. obserwator jest na pokładzie samolotu. Następna warstwa związana jest z pokryciem terenu. Określa ona, na jakich obszarach znajdują się jakie rodzaje podłoża. Ma to znaczenie przy przemieszczaniu się pododdziałów i przy wyświetlaniu terenu, gdyż rodzaj podłoża determinuje użycie konkretnej tekstury. Kolejną warstwą jest warstwa dróg, która podobnie jak pokrycie terenu tworzona jest na podstawie maski, najczęściej tworzonej na bazie zdjęć satelitarnych. Warstwa dróg przed

wyświetleniem na ekranie podlega jeszcze dodatkowej obróbce, zakręty są zaokrąglane, tworzone są skrzyżowania itp. Ostatnią warstwę tworzą obiekty terenowe, dla których oprócz położenia i orientacji w przestrzeni potrzebne są modele 3D. Symulator sam ostatecznie dokonuje umiejscowienia obiektu w terenie, dzięki czemu np. drzewa nie wiszą w powietrzu albo nie rosną pod ziemią (rysunek 3). Obiekty takie jak budynki mogą posiadać modele wnętrza. Dodatkowo możliwe jest definiowanie interakcji z elementami obiektów, co pozwala np. otwierać drzwi czy okna. Aktualnie VBS2 jest zdolny do obsługi terenu o wielkości 150 km x 150 km. Docelowo wielkość terenu ma przestać być ograniczona, ze względu na wprowadzany mechanizm stronicowania (multi-map). Aktualnie głównym ograniczeniem dla wielkości terenu są zasoby komputera, na którym ma być prowadzona symulacja, oraz 32-bitowa architektura symulatora. W planie jest również wydanie wersji VBS2 w 64-bitowej architekturze. Bardzo ważnym aspektem wpływającym na wielkość terenu, jaką zdolny jest obsłużyć symulator, jest sposób umieszczenia obiektów w terenie. Umieszczenie ich na stałe w terenie zmniejszy znacznie maksymalną wielkość terenu, ponieważ symulator na starcie ładuje cały teren do pamięci, dlatego też obiekty i drogi umieszcza się w plikach skryptowych, które ładowane są dynamicznie przez symulator w razie konieczności, dzięki czemu oszczędzamy zasoby komputera.

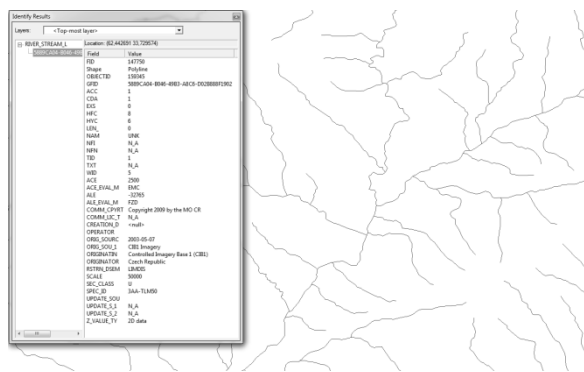


Rys. 3. Fragment widoku 3D terenu w VBS2

#### 4. Opis podstawowych formatów danych terenowych w kontekście ich wykorzystania w VBS2

##### SHP

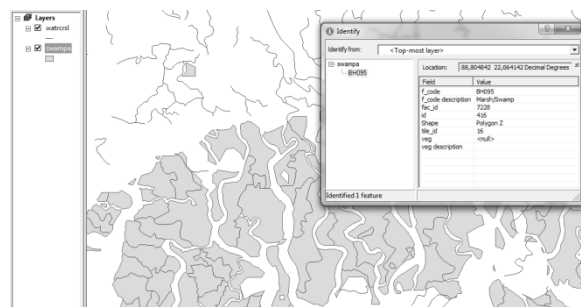
Format zapisu danych wektorowych, który został wprowadzony do użytkowania przez firmę ESRI na początku lat 90. ubiegłego wieku. Pierwotnie format zamknięty, dziś jego specyfikacja jest dostępna publicznie. Umożliwia przechowywanie kształtów, to jest: punktów, zbiorów punktów, linii, łamanych, zbiorów łamanych, wieloboków umiejscowionych w przestrzeni, wraz z opisującymi je atrybutami. Jeden „shapefile” może przechowywać kształty tylko jednego rodzaju, podobnie wszystkie kształty w „shapefile” opisują te same atrybuty, dlatego też jeden „shapefile” przechowuje zazwyczaj dane o podobnych obiektach, np. o jeziorach, natomiast dane o sieci drogowej będą umieszczone w kolejnym „shapefile”. Pracując z wieloma „shapefile”, najczęściej każdy z nich traktuje się jako osobną warstwę na mapie (rysunek 4). Na każdy „shapefile” składają się co najmniej trzy pliki. Plik (.shp) przechowujący dane o kształtach, plik (.shx) zawierający indeks przestrzenny, którego wykorzystanie znacznie przyspiesza proces wyszukiwania kształtów w (.shp), oraz plik (.dbf) zawierający dane o atrybutach kształtów, będący de facto bazą danych w formacie dBase IV. Format ten jest powszechnie wykorzystywany do przechowywania informacji w systemach GIS, dzięki czemu istnieje cała gama narzędzi open source i komercyjnych używających tego formatu. Dostępne są również API w wielu językach programowania (C++, C#, Java, Python) umożliwiające przygotowanie aplikacji wykorzystujących ten format.



Rys. 4. Przykładowy „plik” SHP zawierający dane na temat strumieni i rzek zobrazowany w programie ArcCatalog

##### VPF

VPF to format zapisu danych wektorowych powszechnie stosowany w wojskowych i publicznych systemach GIS. Został on szczegółowo opisany w dokumencie MIL-STD-2407 wydanym przez Ministerstwo Obrony USA w 1996 roku, który jest udostępniony publicznie. Podobnie jak format SHP, VPF służy do przechowywania danych o kształtach, choć tu mówi się raczej o obiektach i klasach, umiejscowionych w przestrzeni, z tą jednak różnicą, że cała mapa zawiera się w jednym „pliku” VPF, co z kolei wymagałoby wielu „plików” SHP. Sam model terenu zastosowany w formacie VPF przypomina geodatabazę; każdy obiekt powiązany może być z kształtem, a pomiędzy obiektami mogą występować relacje. Informacje geolokacyjne mogą w ramach plików VPF być dzielone na macierze, co ma zwiększyć szybkość dostępu do danych i szybkość wyszukiwania obiektów. Bezpośrednio z tych cech formatu wynika jego skomplikowana struktura. Dane na temat obiektów z podobnych klas przechowywane są w tych samych folderach, np. dane na temat hydrologii, na które będą składały się dane jezior, rzek, oczek wodnych itp. VPF jest formatem dużo mniej popularnym niż SHP i dużo mniej narzędzi potrafi poprawnie obsługiwać ten format (rysunek 5). Dostępne są API (Java) pozwalające wykorzystać ten format we własnych aplikacjach.



Rys. 5. Przykładowe dane hydrologiczne (rzeki i bagna) zapisane w formacie VPF, prezentowane w narzędziu ArcMap

##### GDB

GDB to format bazy danych geograficznych wykorzystywany przez oprogramowanie firmy ESRI. Łączy on w sobie możliwości przechowywania informacji geoprzestrzennych oraz typowo bazodanowych. Format ten został wykorzystany w trakcie przygotowania danych do VBS2, aby ułatwić pracę z wieloma plikami

SHP, VPF i DTED. Przy wykorzystaniu API od firmy ESRI pozwoliło to ujednolicić i uprościć proponowane rozwiązanie, które korzystało jedynie z zasobów geobazy danych. Jedną geobazą danych jest katalogiem, w którym umieszczone są wszystkie jej zasoby. Format zapisu danych w GDB jest własnością firmy ESRI i nie jest udostępniony publicznie, stąd decydując się na ten format, należy korzystać z oprogramowania tej firmy bądź z API przez nią udostępnianych (C++, Java, C#, Python). Odpowiednie API można pozyskać ze strony producenta.

## DTED

DTED to siatkowy format zapisu danych wysokościowych powszechnie stosowany w wojskowych systemach GIS. Format ten został szczegółowo opisany w dokumencie MIL-PRF-89020B wydanym przez Departament Obrony USA, który finalną postać uzyskał w 2000 roku. Podobnie jak w przypadku poprzednich formatów opracowanych przez Departament Obrony USA jego specyfikacja jest dostępna publicznie. Dane wysokościowe zapisane w tym formacie przechowywane są w postaci tablic danych wysokościowych, a podane są w metrach nad poziomem morza.



Rys. 6. Przykładowe zobrazowanie danych wysokościowych zapisanych w formacie DTED, poziom 2

Format DTED przewiduje trzy poziomy szczegółowości: 0, 1, 2. Choć w ostatnim czasie zostały zaproponowane trzy kolejne, to nie doczekały się one standaryzacji. Tabela 1 przedstawia odległość pomiędzy punktami wysokościowymi oferowaną przez poszczególne poziomy.

Tabela 1. Gęstość punktów wysokościowych dla różnych poziomów DTED

Poziom DTED	Odległość pomiędzy sąsiednimi punktami wysokościowymi
0	~1 km
1	~100 m
2	~30 m
3	~10 m
4	~3 m
5	~1 m

W zależności od szerokości geograficznej odległość pomiędzy punktami wyrażona w stopniach zmienia się, i tak np. dla DTED2 w okolicach równika wynosi ona jedną sekundę kątową, a w pobliżu bieguna 6 sekund kątowych.

W zależności od poziomu struktura plików zapisu danych wysokościowych w formacie DTED różni się, jednak same dane wysokościowe zapisywane są w postaci tabel. Jedną tabelą z danymi ma szerokość i wysokość jednego stopnia kątowego. Taka komórka zapisana jest w jednym pliku i może zawierać dane 3601 punktów wysokościowych, dla DTED2, a dla DTED1 1201 punktów. Dane wysokościowe punktów brzegowych każdej z komórek powtarzane są w sąsiednich komórkach. W jednej tabeli dane wysokościowe zapisane są wierszami, a wiersze ułożone są zgodnie z szerokością geograficzną i odpowiadają, w kolejności od najmniejszej do największej.

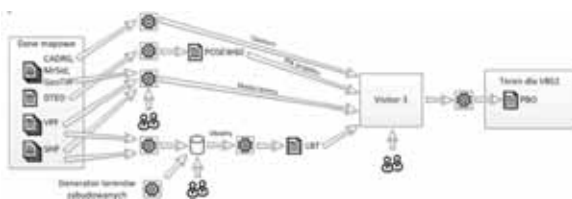
Dokładność danych wysokościowych dla poziomu DTED2 odpowiada szczegółowości konturów na mapie w skali 1:50000 (rysunek 6), natomiast DTED1 odpowiada szczegółowością mapy w skali 1:250000.

## 5. Generowanie terenu z wykorzystaniem różnych zasobów informacyjnych

Ze względu na bardzo dużą popularność symulatora VBS2 w zastosowaniach wojskowych na całym świecie dostępne są dla niego narzędzia i środowiska służące do wspomaganie wytwarzania danych terenowych, takie jak TerraTools firmy TerraSim. Są to jednak rozwiązania komercyjne i zawierają pewne ograniczenia.

Producent VBS2 udostępnia wraz z symulatorem zestaw narzędzi pomocnych przy generacji map, które z dodatkowym wspomaganie w postaci własnego oprogramowania umożliwiają budowanie zaawansowanych map. Rysunek 7 przedstawia

proponowany sposób przygotowania danych terenowych dla symulatora VBS2 z wykorzystaniem różnych źródeł danych mapowych oraz własnych i dostarczanych przez producenta narzędzi [7].



Rys. 7. Schemat procesu generacji danych terenowych dla symulatora VBS2

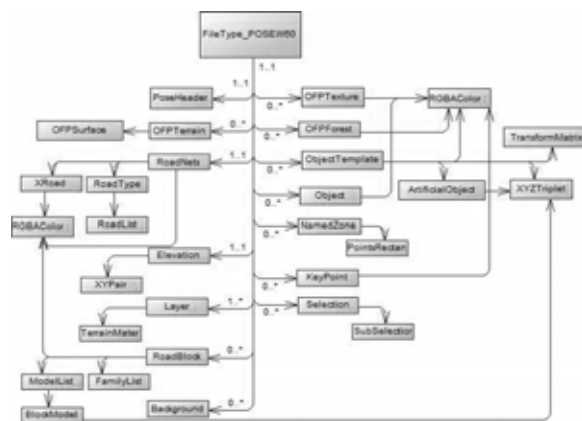
W przypadku generacji terenu dla VBS2 na niektórych etapach istnieje potrzeba ręcznej weryfikacji. Wynika to z faktu, że teren musi być dość szczegółowo odwzorowany, a dane geograficzne często pozbawione są wielu istotnych z punktu widzenia VBS2 elementów, np. takich jak: orientacja budynków, rozkład wnętrza budynków, tekstury obiektów itp.

Na wstępie użytkownik musi zdefiniować obszar, dla którego generowany będzie teren. Korzystając z aplikacji wytworzonej na WAT, użytkownik na mapie zaznacza interesujący go teren, podaje również rozmiar mapy i jej poziom szczegółowości (rysunek 8).



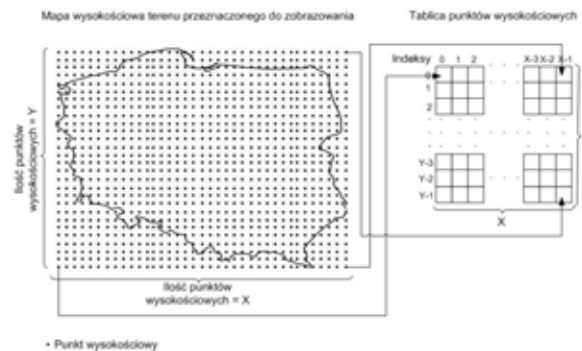
Rys. 8. Przykładowe umiejscowienie terenu do generacji mapy dla VBS2

Aplikacja dokonuje wszystkich niezbędnych obliczeń oraz tworzy plik projektu terenu w formacie POSEW60 (rysunek 9) wraz z niezbędnymi plikami konfiguracyjnymi, który w efekcie końcowym jest przekształcany do formatu 8WVR używanego przez silnik gry VBS2 – Virtual Reality (rysunek 10).



Rys. 9. Diagram klas opisujący pełną strukturę pliku w formacie POSEW60 [1]

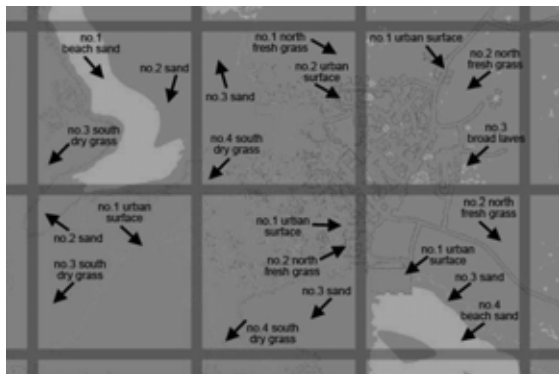
Następnie na podstawie warstwy wysokościowej bazującej na danych w formacie DTED aplikacja tworzy mapę wysokościową i zapisuje ją bezpośrednio do pliku z terenem. Na tym etapie może zająć konieczność wykonania przez aplikację interpolacji danych wysokościowych, zwłaszcza w sytuacji kiedy dane wysokościowe będą miały zbyt małą rozdzielczość w stosunku do tej wybranej przez użytkownika. W tym momencie można zastosować interpolację bilinearną lub najbliższego sąsiada.



Rys. 10. Sposób zapisu wartości punktów wysokościowych w tablicy pliku mapy formatu 8WVR [1]

W kolejnym kroku na podstawie wybranej warstwy z danymi mapowymi, np. w formacie CADRG, generowana jest tekstura terenu, ustalenie rozmiaru odbywa się automatycznie. Tekstura w pierwszej kolejności zapisywana jest do pliku (.png), a później ładowana przez użytkownika do pliku projektu terenu za pomocą aplikacji VBS2 Visitor3. Następnie, wykorzystując dane zapisane rastrowe w formatach CADRG, MrSID, GeoTiff oraz dane wektorowe w formatach SHP i VPF, tworzony jest przez użytkownika plik maski

terenu (rysunek 11). Można w tym celu wykorzystać program Global Mapper, który standardowo dołączany jest do VBS2. Wraz z plikiem maski tworzony jest plik legendy oraz uzupełniane są wpisy w plikach konfiguracyjnych dotyczące terenu i jego właściwości mających wpływ na symulację, np. trudność pokonywania, zapiaszczenie, tekstura, jaka ma zostać użyta. Użytkownik musi przed przystąpieniem do tego etapu przygotować zestaw danych terenowych, które znajdują odniesienie w tworzonym pliku maski.



Rys. 11. Przykładowy plik maski terenu z zaznaczonymi rodzajami terenu i podziałem mapy na segmenty [2]

Po utworzeniu pliku maski tworzona jest baza danych o obiektach terenowych. Zawartość bazy danych zostaje wygenerowana na podstawie danych w formatach SHP i VPF. Sama baza danych została tak zaprojektowana, aby nie było konieczności wykonywania kolejnych konwersji i jej schemat odpowiada formatowi plików (.lbt), za pomocą których obiekty będą dołączane do danych terenowych dla symulatora. Użycie tych plików skutkuje możliwością dodania do terenu dużo większej liczby obiektów, w stosunku do dodania ich bezpośrednio do danych terenowych, dzięki temu, że dane w nich zawarte będą mogły być przez VBS2 ładowane dynamicznie, kiedy będą potrzebne. Ze względu na różny zakres danych informacyjnych przechowywanych w formatach SHP, VPF i (.lbt) niezbędne jest na tym etapie uzupełnienie zawartości bazy danych. Dane geoprzestrzenne nie zawierają standardowo takich informacji jak orientacja obiektów w przestrzeni czy też ich wielkość, natomiast takie dane wymagane są przez symulator do poprawnego renderowania obiektów. W opisywanym rozwiązaniu brakujące dane generowane były losowo w oparciu o cechy charakterystyczne dla wybranego obszaru. Należy również dodać, że na tym etapie dla danych obszarowych, np. dla lasów, generowane

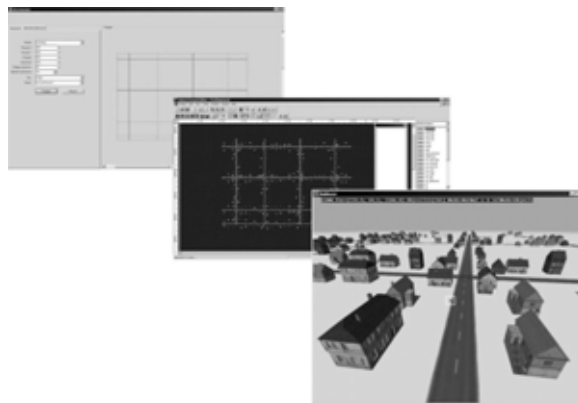
są pojedyncze obiekty, które losowo wypełniają wskazany obszar. Ważnym elementem na tym etapie jest również biblioteka modeli obiektów, gdyż każdemu dodawanemu obiektowi musi zostać przyporządkowany model 3D. Biblioteka obiektów musi zostać przygotowana przez użytkownika, który powinien na tym etapie uwzględnić charakter terenu, np. występowanie określonych gatunków drzew czy też typową dla rejonu zabudowę (rysunek 12).

W tym momencie niezbędna jest ingerencja użytkownika w zawartość bazy danych w celu poprawienia danych zawartych w bazie. Krok ten jest konieczny ze względu na losowanie w poprzednim kroku pewnych parametrów obiektów, co często może prowadzić do niewłaściwych rezultatów, np. budynki tworzące osiedle będą zorientowane każdy w inną stronę. Dokonanie niezbędnych modyfikacji ułatwia generowany podgląd terenu.



Rys. 12. Podgląd terenu w trakcie edycji zawartości bazy danych o obiektach terenowych [1]

Kolejny krok jest opcjonalny. Istnieje możliwość uzupełnienia obiektów pozyskanych z plików SHP i VPF. Można w tym celu posłużyć się szeregiem programów. Jednym z nich jest program opracowany na WAT umożliwiający generowanie losowych terenów zabudowanych (rysunek 13).



Rys. 13. Przykład dodania do terenu dla VBS2 obszaru zabudowanego przy wykorzystaniu generatora opracowanego w WAT [3]

Po wykonaniu poprawek i uzupełnieniu zawartości bazy danych na jej podstawie generowane są pliki (.lbt).

Na tym etapie użytkownik, korzystając z programu Visitor3 (lub nowszego Visitor4), scala wszystkie elementy wytworzone w trakcie realizacji procesu. Ma w tym miejscu również możliwość dokonania niezbędnych korekt.

Ostatnim krokiem jest spakowanie dołączonym do VBS2 narzędziem plików terenowych i zapisanie ich w formacie (.pbo).

## 6. Wnioski

W opracowaniu przedstawiono sposób przygotowania danych terenowych na potrzeby wysokorozdzielczej symulacji działań taktycznych realizowanych w środowisku symulatora wirtualnego VBS2. Na potrzeby tego rodzaju symulacji wymagane jest bardzo szczegółowe odwzorowanie terenu działań na podstawie map cyfrowych standardu GIS (m.in.: VPF, DTED, SHP, CADRG, MrSID, GeoTiff). Efektywność procesu przekształcania danych terenowych zapisanych w formatach standardu GIS w dużej mierze zależy od jakości, szczegółowości i poprawności tych danych. Dokładność odwzorowania dla symulacji wysokorozdzielczej jest bardzo istotna dla realizacji symulacji na poziomie taktycznym, a w szczególności dla działań asymetrycznych i w terenie zabudowanym. Przedstawione w opracowaniu rozwiązanie zostało praktycznie wykonane i zastosowane do symulacji wielorozdzielczej działań konfliktowych i kryzysowych.

## 7. Bibliografia

- [1] M. Niedziela, praca dyplomowa pt. *Mechanizm generacji danych o ukształtowaniu terenu na potrzeby symulacji działań militarnych wysokiej rozdzielczości*, Warszawa, WAT, 2011.
- [2] <http://community.bistudio.com>
- [3] T. Szczepański, praca dyplomowa pt. *Projekt generatora elementów infrastruktury urbanistycznej w środowisku VBS2*, Warszawa, WAT, 2011.
- [4] R. Antkiewicz, W. Kulas, A. Najgebauer, D. Pierzchała, J. Rulka, Z. Tarapata, R. Wantoch-Rekowski, "Modelling and simulation of C2 processes based on cases in the operational simulation system for CAX", *1st Military Communication and Information System Conference MCC'2006*, Gdynia, 2006.
- [5] R. Antkiewicz, W. Kulas, A. Najgebauer, D. Pierzchała, J. Rulka, Z. Tarapata, R. Wantoch-Rekowski, "Some aspects of designing and using deterministic and stochastic simulators for military trainings and CAX'es", *Proceedings of the Military Communications and Information Systems Conference MCC'2008*, September 2008, Cracow, Poland.
- [6] R. Antkiewicz, W. Kulas, A. Najgebauer, D. Pierzchała, J. Rulka, Z. Tarapata, R. Wantoch-Rekowski, M. Chmielewski, R. Kasprzyk, J. Koszela, „Symulacyjny model działań bojowych szczebla operacyjnego i taktycznego”, w: *Badania operacyjne i systemowe a zagadnienia społeczeństwa informacyjnego, bezpieczeństwa i walki*, str. 253–266, IBS PAN, Warszawa, 2008.
- [7] J. Koszela, M. Mańko, M. Niedziela, H. Ostap, T. Tarnawski, "Preparation of terrain data for the needs of multi-resolution battle space simulation system", in: *Concepts and Implementations for Innovative Military Communications and Information Technologies*, Warszawa, 2010.

## **Advanced methods of terrain data preparation for VBS2 simulator**

J. KOSZELA, T. DROZDOWSKI, M. NIEDZIELA, H. OSTAP,  
R. WANTOCH-REKOWSKI

The paper contains a description of the terrain data preparation for the virtual simulator VBS2. A model of terrain use in the simulator was presented. The technology and the software used to generate terrain data with the most important properties was presented. The specification of the main data formats was presented (SHP, GDB, VPF, TDED). The article concludes with a description of the process of generation of terrain data for the selected area.

**Keywords:** VBS2, simulation, terrain data.