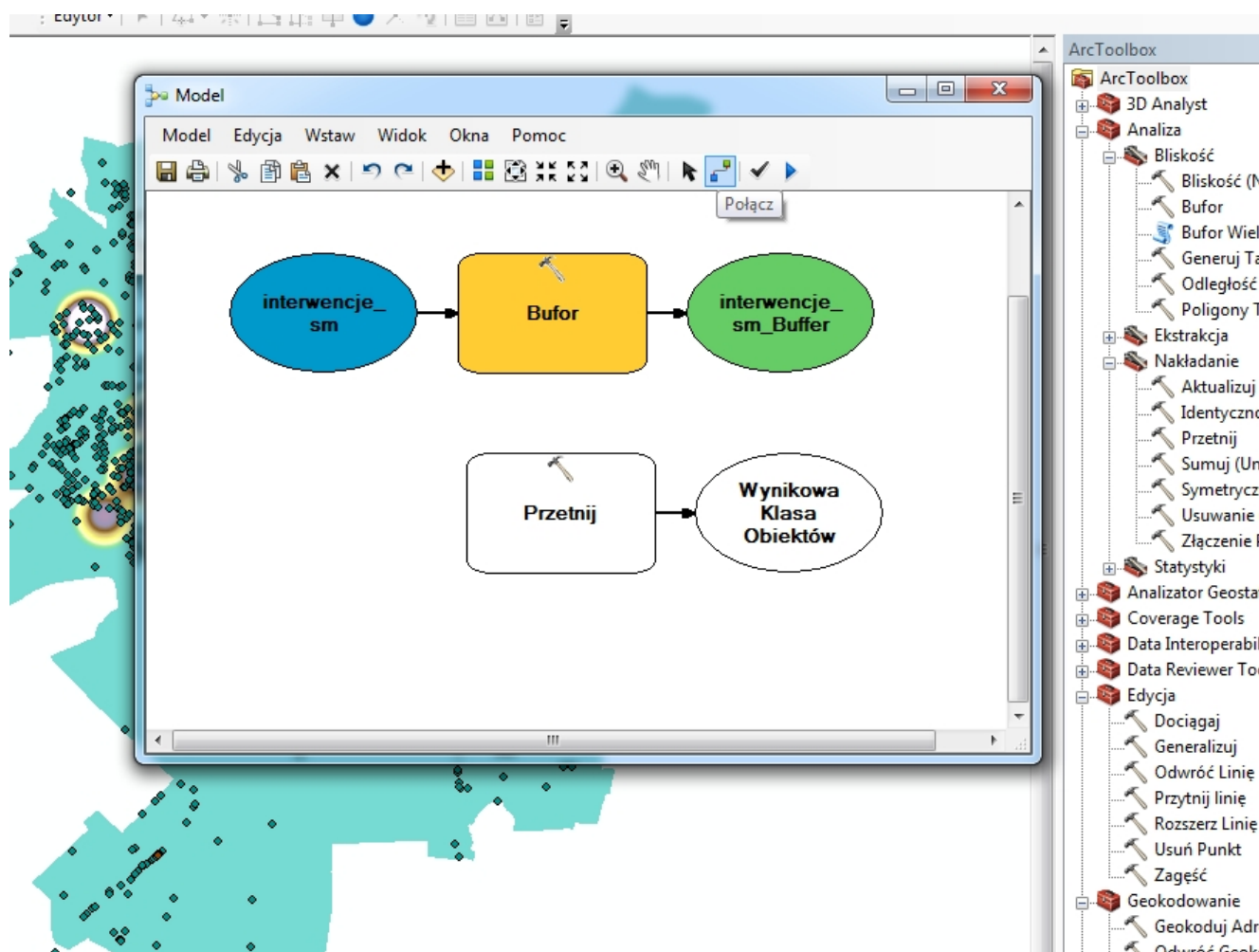


# ModelBuilder i skrypty Pythona w codziennej pracy z ArcGIS

Przygotowanie danych czy ich obróbka bywają nużące, ponieważ wykonywane są mechanicznie, przy znikomym zaangażowaniu użytkownika. Jeśli od początku widać, że praca będzie przebiegała etapami, a nasz wkład ograniczy się do wybierania kolejnych narzędzi, warto rozważyć użycie mechanizmów wspierających automatyzację pracy, dostępnych w oprogramowaniu Esri.

Automatycznie mogą być wykonywane pewne działania podejmowane wielokrotnie i w bardzo podobnej formie, np. codzienny *backup* zrealizowanych zadań czy dodanie nowych obiektów do bazy danych. Proces ten jest dodatkowo wspomagany przez mechanizmy systemu operacyjnego, pozwalające na samoczynne wywoływanie skryptów o ustalonych porach – ArcGIS pozwala wykorzystywać swoje atuty w skryptach napisanych w języku Python.

Drugim narzędziem wspierającym automatyzację jest ModelBuilder – wizualny edytor pozwalający na zarządzanie obiegiem pracy wykonywanej przez narzędzia geoprzetwarzania. Połączenie funkcjonalności obu tych instrumentów daje możliwość dobrego zorganizowania procedur pracy w [ArcGIS](#) i odciążenia użytkownika z obowiązku wielokrotnego wykonywania tych samych zadań.

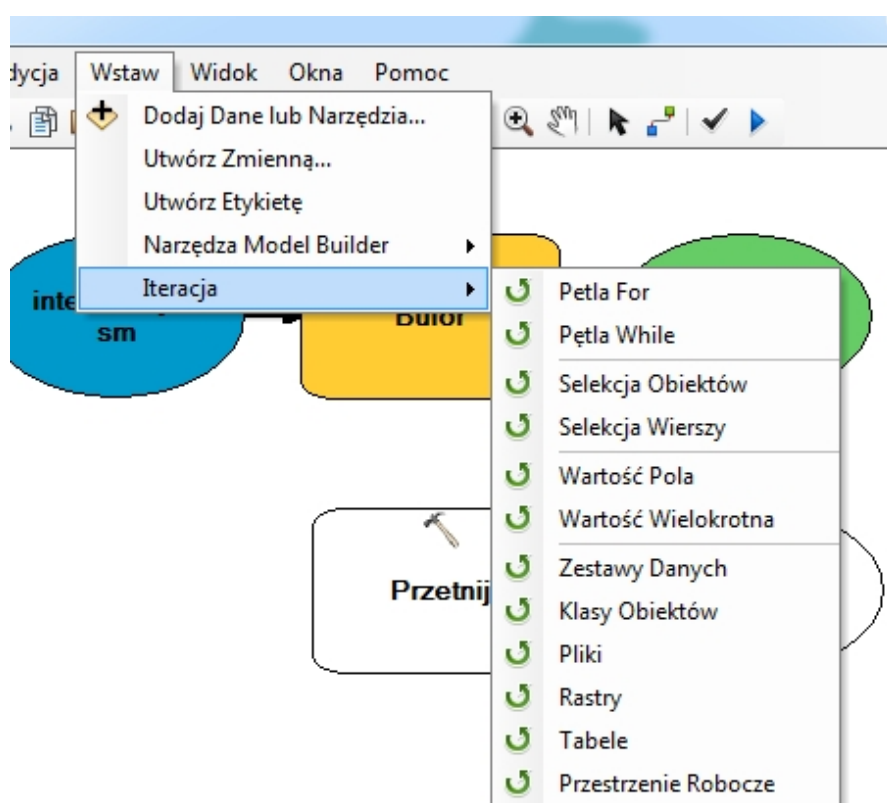


Rys. 1. W trakcie budowania modelu.

## Narzędzia na tapecie

ModelBuilder to narzędzie wbudowane w pakiet ArcGIS for Desktop. Pozwala ono na tworzenie modeli, czyli obiegów pracy zbudowanych z narzędzi geoprzetwarzania, i skryptów, przedstawionych graficznie w formie zależności między poszczególnymi elementami, oraz umożliwia ich edycję. Podczas sekwencyjnego uruchamiania wielu narzędzi z ArcGIS Toolbox bardzo ważne są zapewnienie każdemu z nich odpowiednich danych wejściowych oraz świadomość tego, co pojawi się jako wynik działania. Tylko mając pełną informację o tym, co dzieje się w każdym „kroku” modelu, można przewidzieć nieoczekiwane sytuacje i przypadki graniczne. Graficzna prezentacja kolejnych etapów realizacji modelu ułatwia pracę, co jest bardzo ważne dla mniej zaawansowanych użytkowników. Takie wykorzystanie ModelBuildera często zachęca do wejścia w świat

skryptów, który wymaga większego doświadczenia. Model jest też dobrą metodą budowania nowych rozwiązań, szczególnie tych bardziej złożonych. Wybierając kolejne „kroki” realizujące rozwiązanie, warto wspomagać się wizualnym edytorem, służącym do przekazywania parametrów czy wyników pracy pomiędzy narzędziami.



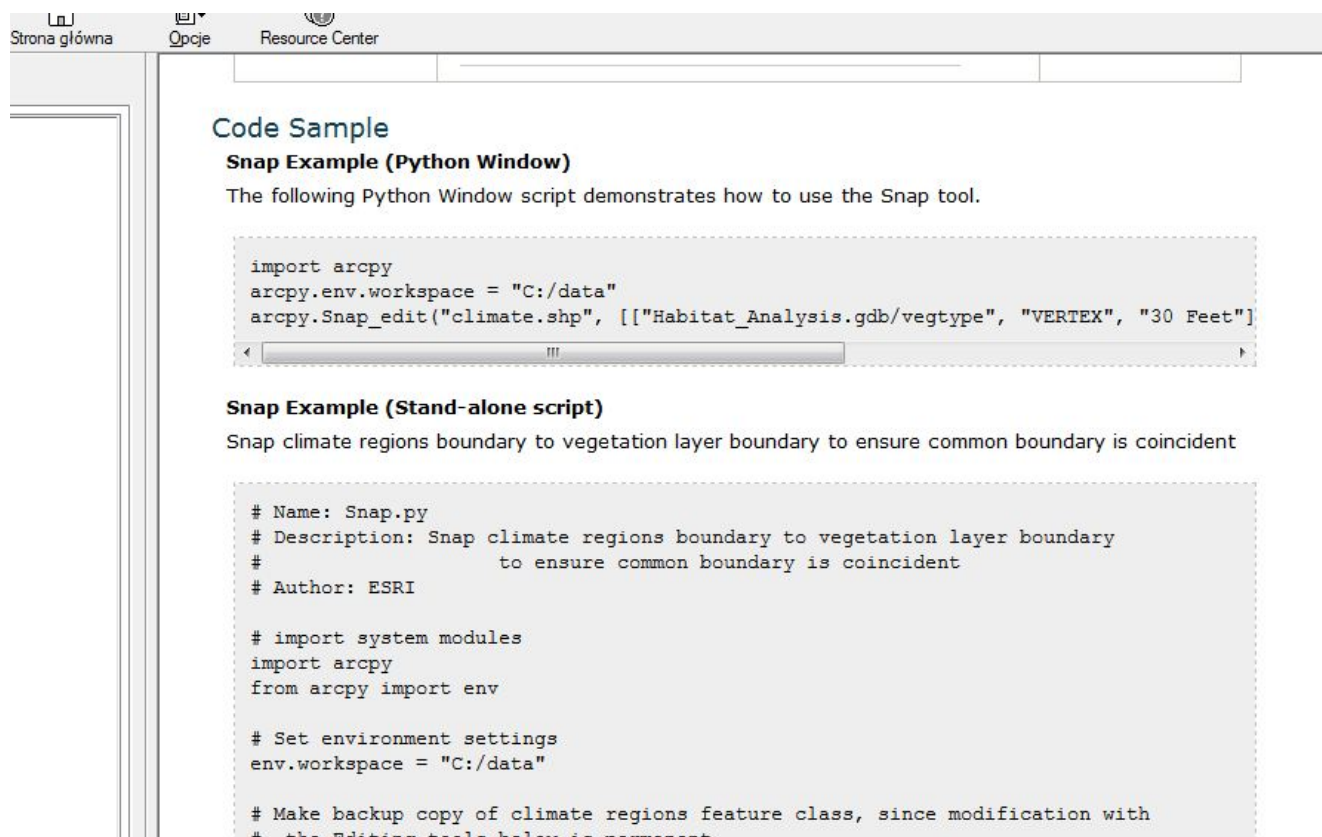
Rys. 2. Narzędzia dodatkowe ModelBuildera.

Aby rozpocząć pracę z ModelBuilderem, wystarczy kliknąć przycisk w interfejsie użytkownika ArcMap lub ArcCatalog, po czym pojawi się okno tworzenia nowego modelu. Dodawanie kolejnych narzędzi odbywa się poprzez przeciągnięcie ich z ArcToolbox. Pojawiają się wówczas opisany nazwą prostokąt z zaokrąglonymi narożnikami, reprezentujący narzędzie, i owal, reprezentujący wynik pracy tego narzędzia. Po określeniu koniecznych parametrów elementy modelu zostają wypełnione odpowiadającymi im kolorami – świadczy to o gotowości danego elementu do pracy. Obiekty z tabeli zawartości można wprost przeciągać do modelu, nie tracąc czasu na wskazywanie tabeli

czy klasy obiektów na dysku. ModelBuilder zawiera zestaw narzędzi wspomagających pracę i rozszerzających możliwości geoprzetwarzania. Gotowy model może zostać dodany do zbioru narzędzi ArcToolbox (w ramach własnej skrzynki narzędziowej, podczas gdy systemowe skrzynki narzędziowe są tylko do odczytu), dzięki czemu będzie dostępny z interfejsu ArcGIS również do użycia w innych modelach czy skryptach. Przed dodaniem warto zadbać o to, aby model był interaktywny, a więc przyjmował parametry wejściowe w postaci argumentów, zamiast odnosić się do sztywno wskazanych danych. Osiąga się to, oznaczając wybrane owale jako parametry modelu.

## **Nie taki Python straszny**

Platforma ArcGIS wykorzystuje język skryptowy Python już od kilku edycji. Z każdą kolejną wersją integracja tej platformy z Pythonem jest większa – udostępnionych zostaje więcej komend i funkcji. Do wersji 9.3.1 dostęp do funkcjonalności ArcGIS z poziomu Pythona realizowany był przez moduł ArcGISsripting. Wersja 10. przyniosła małą rewolucję – firma Esri przygotowała bibliotekę ArcPy, która oprócz dostarczenia zbioru klas typowych dla danych przestrzennych i funkcji do pracy z nimi, poszerzyła możliwości użycia skryptów o tworzenie kompozycji map i pogłębiła integrację z ArcGIS. Aby zagwarantować działanie rozwiązań przygotowanych przez użytkowników starszych wersji, zapewniono zgodność wstecz – z ArcGIS 10 działają zarówno skrypty wykorzystujące ArcPy, jak i te bazujące na module ArcGISsripting.



Rys. 3. Okno pomocy narzędzia – widoczny przykład użycia w kodzie.

Python to nowoczesny, bardzo elastyczny skryptowy język programowania. Programistom nie jest narzucany styl programowania (może to być programowanie strukturalne, obiektowe lub funkcyjne). Dostępne są też w pełni dynamiczny system typów i mechanizm zarządzania pamięcią, dzięki czemu łatwo pracować nawet osobom niezaznajomionym z programowaniem. Rozsądnie prowadzony kod jest czytelny, a użycie biblioteki Esri – intuicyjne. Do dyspozycji są wszystkie narzędzia z ArcToolbox (dostępne na aktualnym poziomie licencyjnym), które wykorzystuje się w sposób analogiczny do tego z poziomu okna dialogowego, określając wymagane parametry lub wskazując ścieżki do danych. Sposób użycia każdego narzędzia geoprzetwarzania w skrypcie jest pokazany w jego dokumencie pomocy – zaprezentowana zostaje tam kolejność parametrów oraz lista argumentów koniecznych i opcjonalnych.

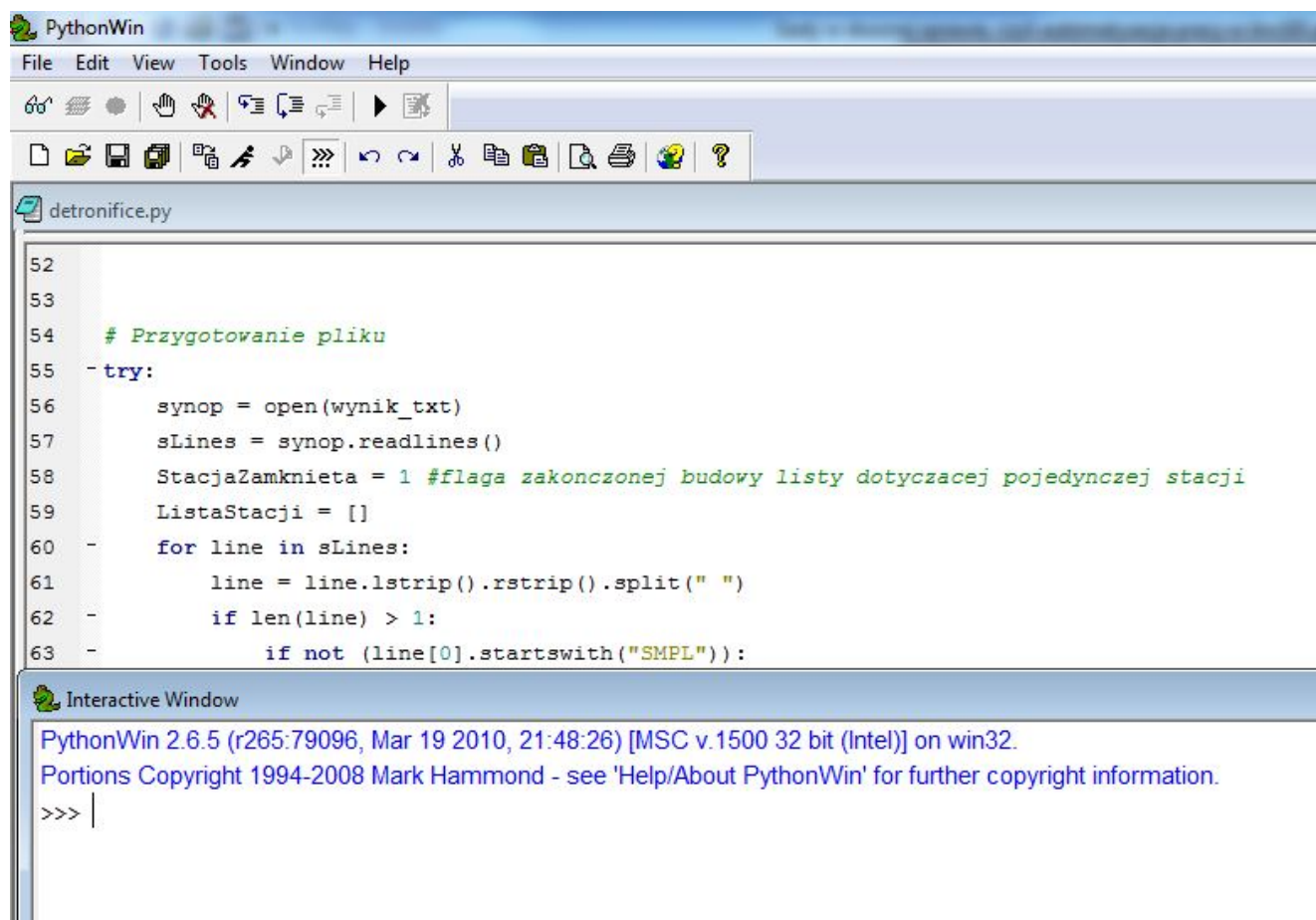
## Gdzie pisać

Swoje pierwsze polecenia Pythona można wykonać już w ArcMap. W

głównym interfejsie dostępny jest przycisk uruchamiający okno Pythona, w którym da się wpisać pojedyncze polecenia lub krótkie bloki oraz wywołać narzędzia. Jest to wygodne dla osób dobrze zaznajomionych z geoprzetwarzaniem w Pythonie i często z niego korzystających. Polecenie można szybciej wpisać (szczególnie dzięki temu, że działa sprawne podpowiadanie składni), niż wyszukać w gąszczu narzędzi ArcToolbox. Nie na tym jednak polega automatyzacja przy użyciu skryptów.

Kod Pythona jest interpretowany w czasie wykonywania, nie ma więc potrzeby kompilowania go – dlatego do pisania wystarczy prosty edytor tekstowy. Dużo wygodniejsze jest jednak kodowanie w środowisku programistycznym. Przy większych projektach staje się to wręcz koniecznością, ponieważ skuteczne mechanizmy debugowania, kolorowanie słów kluczowych, autouzupełnianie słów czy wsparcie prowadzenia treści bardzo ułatwiają pracę (dlatego warto od początku zaczynać swoją przygodę z Pythonem w takim narzędziu).

Na rynku istnieje wiele rozwiązań. Firma Esri postawiła na PythonWin – mały darmowy edytor dołączany na płycie z ArcGIS. Po uruchomieniu widoczny jest prosty pasek narzędziowy z przyciskami do edycji treści kodu i uruchomienia go oraz z narzędziami debuggera (służącymi do poszukiwania błędów w działaniu skryptu). Poniżej wyświetlają się dwa okna – okno skryptu, w którym tworzona jest właściwa treść, oraz okno Interactive Window, w którym można komunikować się bezpośrednio z interpreterem Pythona, wydając mu pojedyncze polecenia. Esri nie narzuca PythonWin. Obsługiwane są również inne środowiska.



Rys. 4. PythonWin – interfejs i fragment kodu z widocznym formatowaniem.

ArcPy umożliwia użycie narzędzi geoprzetwarzania, a także dostarcza bardziej zaawansowanych narzędzi. Dzięki obiektom *Cursor* w skryptach można uzyskać dostęp do pojedynczych obiektów z klasy obiektów lub wierszy w tabeli, edytować je, usuwać i tworzyć nowe, również na poziomie geometrii obiektów – można odczytywać punkty tworzące obiekt, czy tworzyć nowe obiekty. Dostępne są obiekty opisujące dane; informacje te można przechowywać w listach i iterować po nich, uzyskując prawdziwie automatyczną pracę na zbiorze. ArcPy współpracuje z mechanizmem Data Driven Pages, wspomagając tworzenie serii map i komponowanie z nich zbiorczych dokumentów. Gotowy skrypt, podobnie jak model, można dodać do ArcToolbox jako własne narzędzie i wykorzystywać go w innych skryptach czy modelach. Może on być dostępny do użycia z poziomu dialogowego okna wywołania, wygodnego dla mniej zaawansowanych użytkowników.

**Czy to przydatne?**



Warto znać mechanizmy automatyzacji pracy i tworzenia modeli. W tym artykule możliwości, jakie daje ArcGIS łącznie z ArcPy, zostały tylko zarysowane – tak, aby zachęcić do zapoznania się z nimi we własnym zakresie. Szczególnie godny polecenia jest Python, ponieważ z każdą edycją jego możliwości się poszerzają, a producent oprogramowania zapowiedział, że od wersji 10.1 (która ma się pojawić już w tym roku) potencjał skryptów jeszcze wzrośnie. Będzie możliwe tworzenie wtyczek (Add-In) do ArcGIS Desktop. To bardzo duży krok naprzód i gest w kierunku średnio zaawansowanych użytkowników. Umożliwi on rozbudowę funkcjonalności platformy bez sięgania po narzędzia bardziej skomplikowane, wymagające większej wiedzy informatycznej, takie jak Java czy Microsoft.Net. Python to dobra droga do wejścia w świat tworzenia własnych rozwiązań i narzędzi GIS.