

Tak jak wspomniałem wcześniej, pozostałe elementy obrazu prościej jest narysować od podstaw. Używając obrazu rastrowego jako „podkładki”, narysuj jedną łopatę śmigła, a następnie skopiuj ją i obróć o 120° , by uzyskać dwie pozostałe. Tło, wraz z wewnętrznymi kręgami, narysuj jako zespół kół, z których niektóre mają białe obwódki. Napis „T.M. REG. U.S. PAT. OFF.” bardzo ładnie można odwzorować za pomocą jakiejś prostej czcionki bezszeryfowej — nawet standardowej **Arial**.

Gorzej z białymi napisami: „CONSTANT SPEED”, „SELECTIVE PITCH”, „FEATHERING”. Użyto do nich bardzo podobnej czcionki, co w napisie „Electric Propellers”. Nie udało mi się znaleźć podobnej, a wektoryzacja tego kształtu nie wyszłaby zbyt dobrze — za dużo byłoby „babrania” się z poprawianiem kątów ostrych. Ostatecznie zdecydowałem się użyć innego kroju pisma. Choć nie jest takie samo, to na pierwszy (a może i drugi) rzut oka nie razi, szczególnie gdy nigdzie obok nie widać oryginału¹ (Rysunek 12.43.7):



Rysunek 12.43.7 Efekt finalny — całkowicie wektorowy obraz rastrowego pierwowzoru

¹ Swoją drogą — wygląda na to, że datujący się z początku XX w. krój markowego napisu „Coca-Cola” jest wykonany w bardzo popularnym w tym czasie stylu. Tyle, że inne firmy, używające w swoich logo podobnych czcionek, już zdążyły poznać.

Rozdział 13. Blender — ogólne

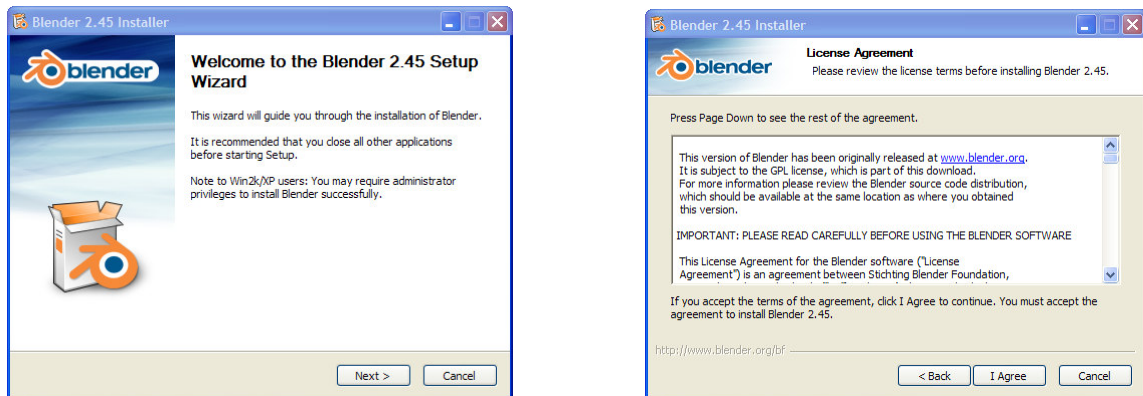
W tym rozdziale przedstawione są zagadnienia związane "ogólnie" z Blenderem. Często są to także takie tematy, których nie potrafiłem jednoznacznie przypisać do grup tematycznych, zawartych w dalszych rozdziałach.

Pominę w tym rozdziale zagadnienia, które przedstawiłem wcześniej (Rozdział 3). Chodzi tu o:

- wprowadzenie do programu. Opisałem je już w sekcji 3.1 (str. 55)
- metody zmiany projekcji (zoom, pan, obrót wokół obiektu). Opisałem je już w sekcji 3.2 (str. 62);

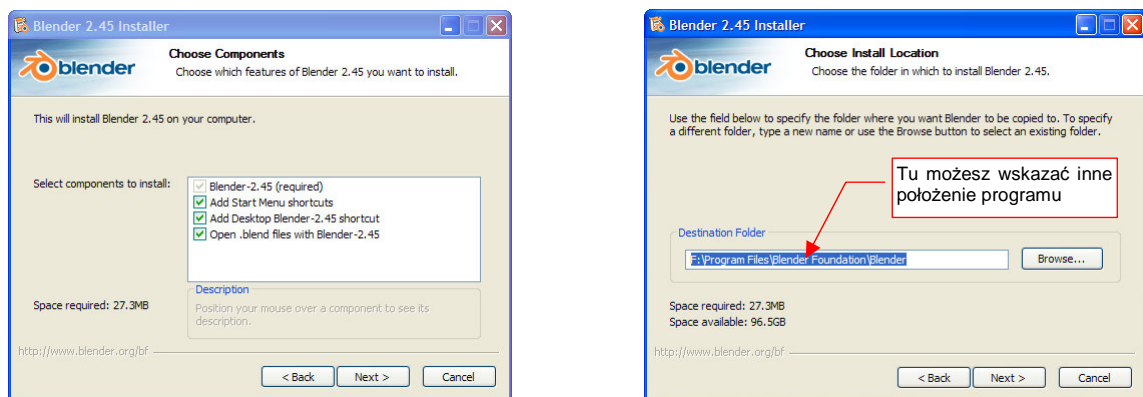
13.1 Instalacja

Ta sekcja opisuje instalację Blendera w wersji 2.47. Po uruchomieniu programu instalacyjnego pojawia się ekran "powitalny", a następnie ekran z umową licencyjną (Rysunek 13.1.1):



Rysunek 13.1.1 Instalacja Blendera - pierwsze dwa ekran

Po naciśnięciu przycisku **I Agree** na ekranie **License Agreement**, przejdziesz do dwóch kolejnych ekranów. Można tu zmienić opcje instalacji (Rysunek 13.1.2):



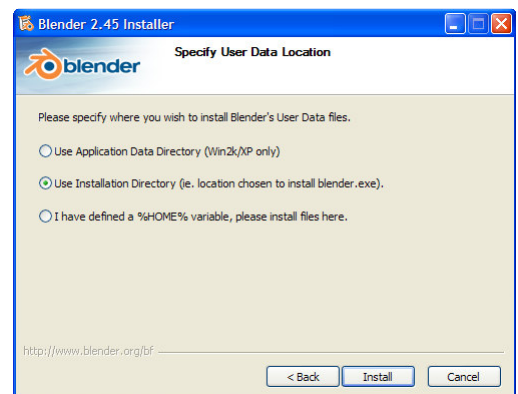
Rysunek 13.1.2 Instalacja Blendera — wybór opcji instalacji

Na ekranie **Choose Components** nie ma co, szczerze mówiąc przebierać. Za to na ekranie **Choose Install Location** możesz wskazać inny folder dla Blendera, jeżeli nie odpowiada Ci położenie domyślne.

Wreszcie na ekranie **Specify User Data Location** możesz określić, gdzie Blender ma umieścić swoje dane konfiguracyjne (Rysunek 13.1.3).

Dane, o które tutaj chodzi, to folder **.blender**, wraz z całą zawartością. Składają się na nią:

- domyślne ustawienia użytkownika (zapisane w pliku **.B.blend**);
- folder **locale**, zawierający translację programu na różne języki;
- folder **scripts**, zawierający skrypty Pythona, dostarczane wraz z Blenderem. Są to implementacje wielu dodatkowych poleceń menu.



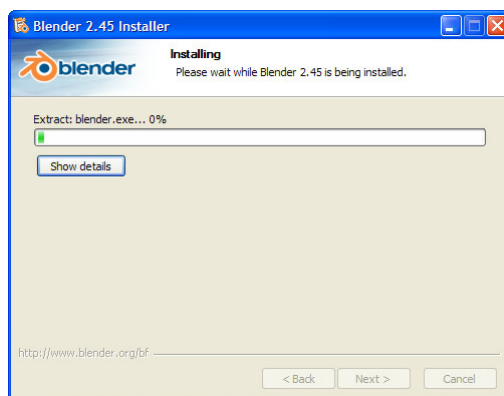
Rysunek 13.1.3 Określanie położenia danych konfiguracyjnych Blendera

Domyślnie dla programów Windows jest proponowana pierwsza pozycja (**Use Application Data Directory**). Co ta opcja oznacza? Gdy ją wybierzesz, Blender umieści folder **.blend** w Twoim katalogu użytkownika (nazywanego

w Windows także "profilem"), w folderze *Dane aplikacji*.¹ (Pełna ścieżka to *C:\Document and Settings\<Twoje id>\Dane aplikacji*.) Wiele różnych programów przechowuje tu Twoje indywidualne ustawienia. Umieszczenie tu folderu *.blender* ma swoje dobre i złe strony. Dobre — to niezależność danych konfiguracyjnych (pliku *.B.blend*) na wgranie nowej wersji programu. Złe — skrypty, implementujące dodatkowe polecenia, także od tego nie zależą. Zakładając, że każda nowa wersja coś tam dogra, będą się tylko kumulować - stare i nowe dodatki. Nie jest to pożądane: nieraz obserwowałem, jak jakaś dodatkowa funkcja, realizowana we wczesnej wersji przez skrypt, po dwóch — trzech aktualizacjach Blendera zostaje przepisana w C++ i dodana do jądra programu. Po takiej zmianie skrypt jest już niepotrzebny. Nie pojawia się już więcej w wśród plików tej nowej wersji. Jeżeli jednak folder *.blender\scripts* znalazłby się w katalogu *Dane aplikacji* — to nic, poza Twoją ręczną interwencją, go stamtąd nie usunie.

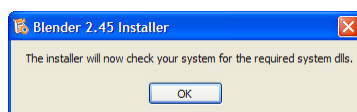
Osobiście wolę trzymać wszystko, co dotyczy konkretnej wersji Blendera w jednym miejscu. Proponuję więc wybrać z ekranu *Specify User Data Location* drugą możliwość: *Use Installation Directory*. Spowoduje to umieszczenie folderu *.blender* w folderze *Blender*. (Czyli w tym samym miejscu, co *blender.exe* i pozostałe pliki.) Jest to jednak tylko moja sugestia: jeżeli wybierzesz opcję proponowaną domyślnie, też będzie dobrze. (Pamiętaj tylko potem, gdzie szukać tych dodatkowych plików Blendera).

Po naciśnięciu przycisku **Install** rozpoczyna się proces instalacji (Rysunek 13.1.4):



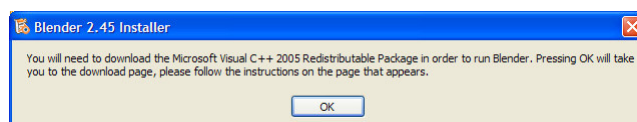
Rysunek 13.1.4 Postępy instalacji

Po wgraniu plików Blendera pojawi się komunikat o sprawdzaniu obecności wymaganych bibliotek (Rysunek 13.1.5):



Rysunek 13.1.5 Informacja o rozpoczęciu sprawdzania środowiska

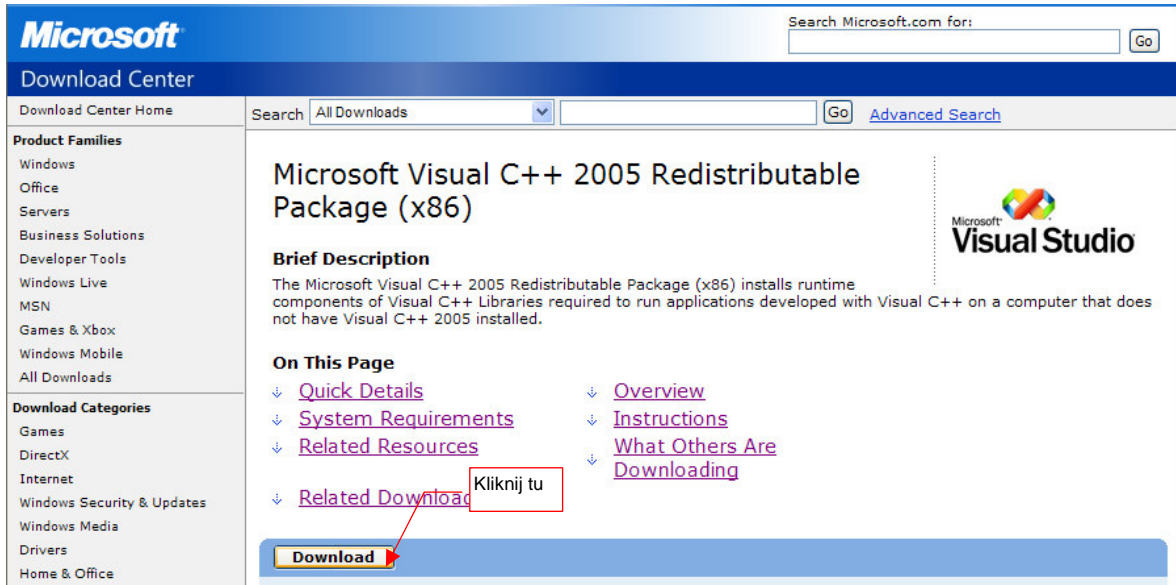
Jeżeli masz zainstalowane wszystkie wymagane dodatki (tak jest np. na Windows Vista), zobaczysz po chwili okno finalne (Rysunek 13.1.10, str. 754). Jeżeli jednak brakuje Ci komponentów wymaganych dla programów skompilowanych w MS Visual C++ 2005, pojawi się odpowiedni komunikat (Rysunek 13.1.6):



Rysunek 13.1.6 Informacja o braku bibliotek (tzw. *runtime*) dla MS Visual C++ 2005

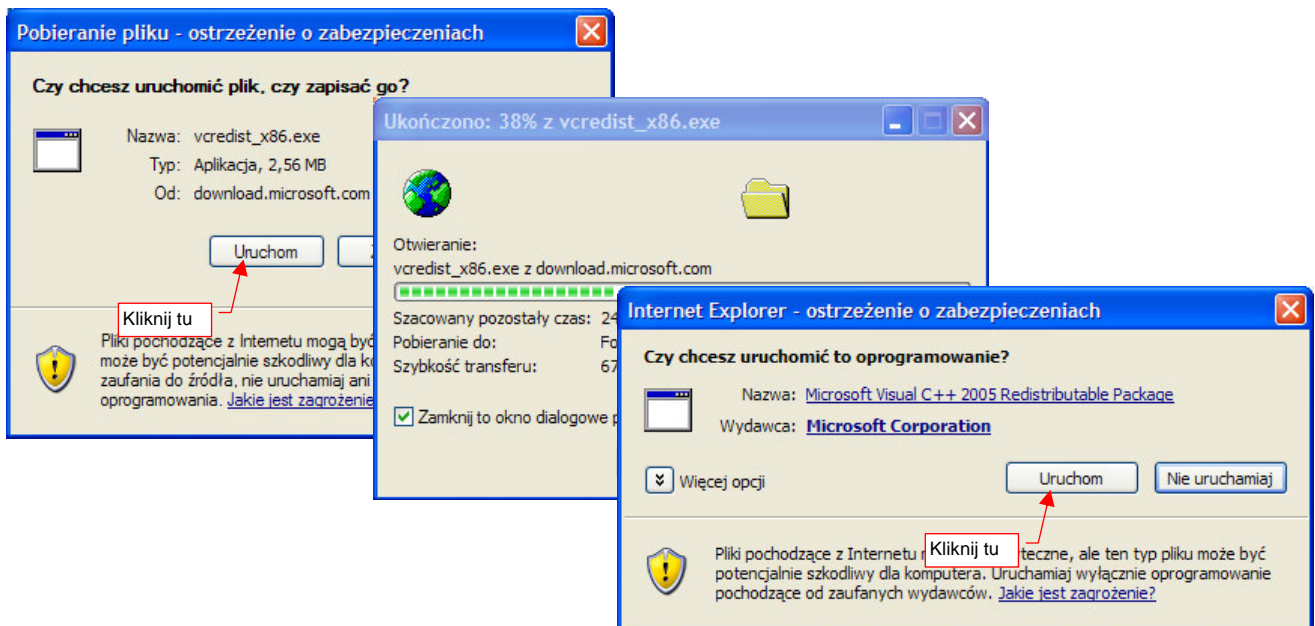
¹ Nazwa tego folderu jest spolszczona. W anglojęzycznych Windows nosi nazwę *Application Data*. (Podobnie jak angielskie *My Documents* w polskiej wersji Windows nazywa się *Moje dokumenty*.)

Brakujący komponent jest swobodnie rozpowszechniany (choć, oczywiście nie na zasadach licencji [Open Source](#)). Jeżeli w trakcie tej instalacji Twój komputer jest podłączony do Internetu, Blender po tym komunikacie sam otworzy stronę Microsoftu (Rysunek 13.1.7):



Rysunek 13.1.7 Strona, z której można pobrać program instalacyjny *runtime* MS VC++ 2005

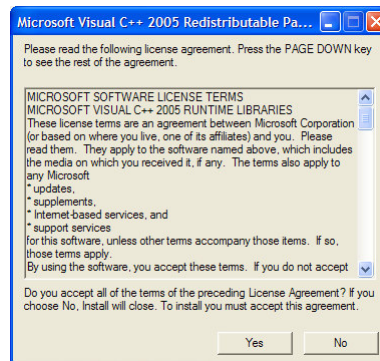
Wystarczy kliknąć w przycisk [Download](#) na tej stronie Microsoftu (Rysunek 13.1.8):



Rysunek 13.1.8 Pobieranie i uruchamianie komponentów wymaganych przez MS VC++ 2005

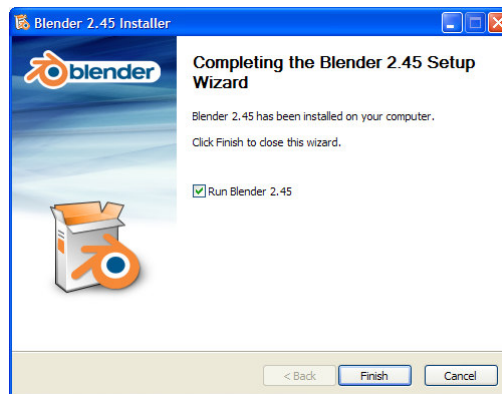
Po co się jednak tak męczyć? Wśród plików, dostarczonych wraz z tą książką, znajdziesz dokładnie ten sam plik. Pobierz i uruchom: http://www.samoloty3d.pl/downloads/wm/install/blender/vcredist_x86.exe.

Po uruchomieniu instalacji brakującego komponentu, pojawi się ekran z warunkami licencji (Rysunek 13.1.9):



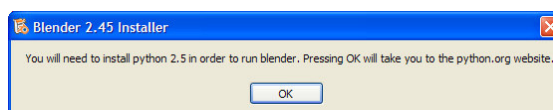
Rysunek 13.1.9 Warunki licencji na runtime MS Visual C++ 2005

Po naciśnięciu **Yes** biblioteki VC++ 2005 zostaną zainstalowane. W tym czasie instalator Blendera pokazuje już ostatni ekran:



Rysunek 13.1.10 Blender — finalne okno instalacji

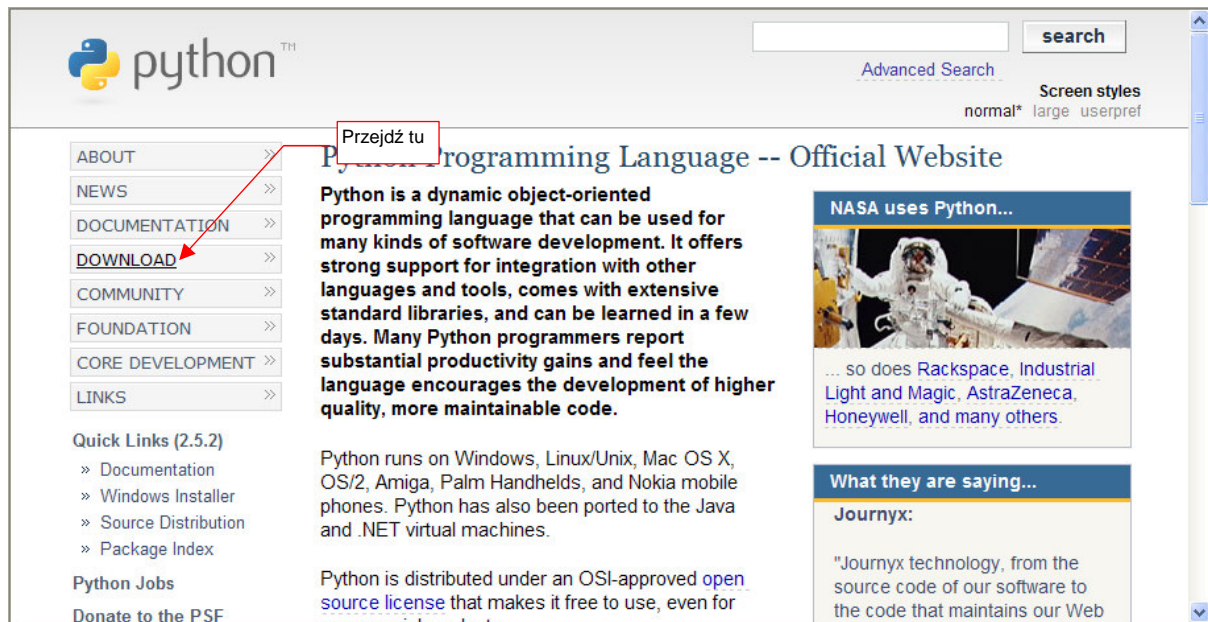
Po naciśnięciu **Finish** pojawi się sugestia, by zainstalować pełną wersję interpretera języka Python (Rysunek 13.1.11). Stanie się to jednak tylko wtedy, gdy Twój komputer będzie połączony z Internetem:



Rysunek 13.1.11 Informacja, by zainstalować Python 2.5

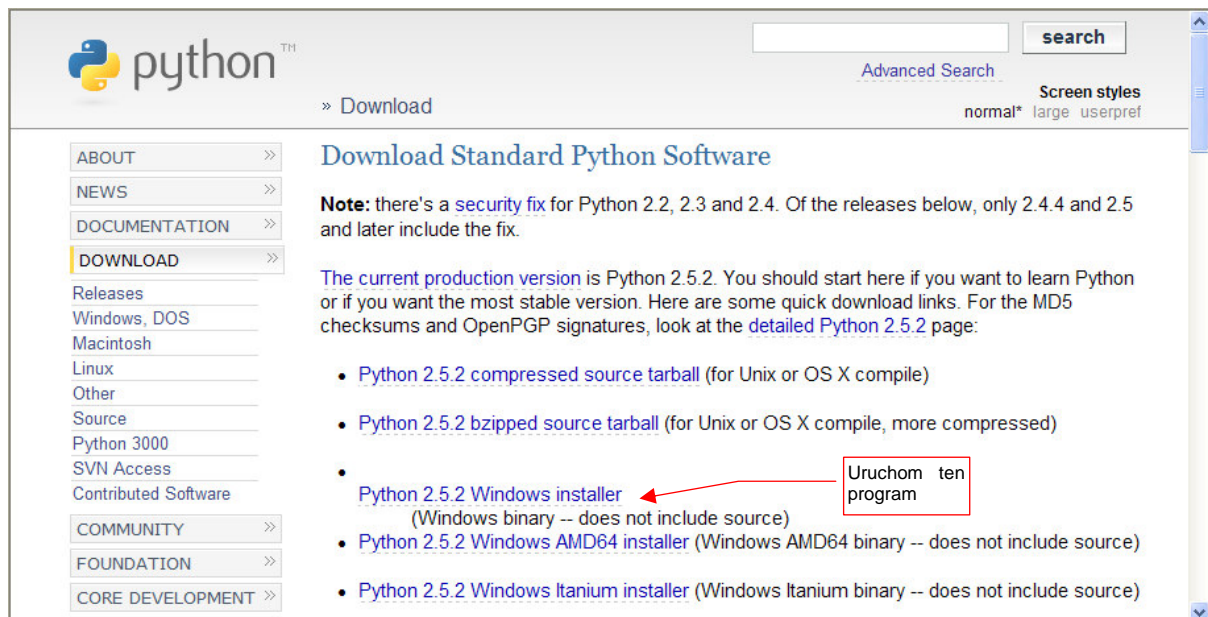
W istocie, Blender zawiera ograniczoną wersję Pythona 2.5. Krok, który tu jest proponowany, jest opcjonalny. (Wiele przydatnych skryptów, napisanych przez użytkowników Blendera, może wymagać pełnej wersji Pythona).

Jeżeli w czasie instalacji byłeś połączony z Internetem, Blender sam otworzy w przeglądarce stronę projektu Python (www.python.org — Rysunek 13.1.12):



Rysunek 13.1.12 Strona główna Pythona

Przejdź do sekcji **DOWNLOAD** (Rysunek 13.1.13):

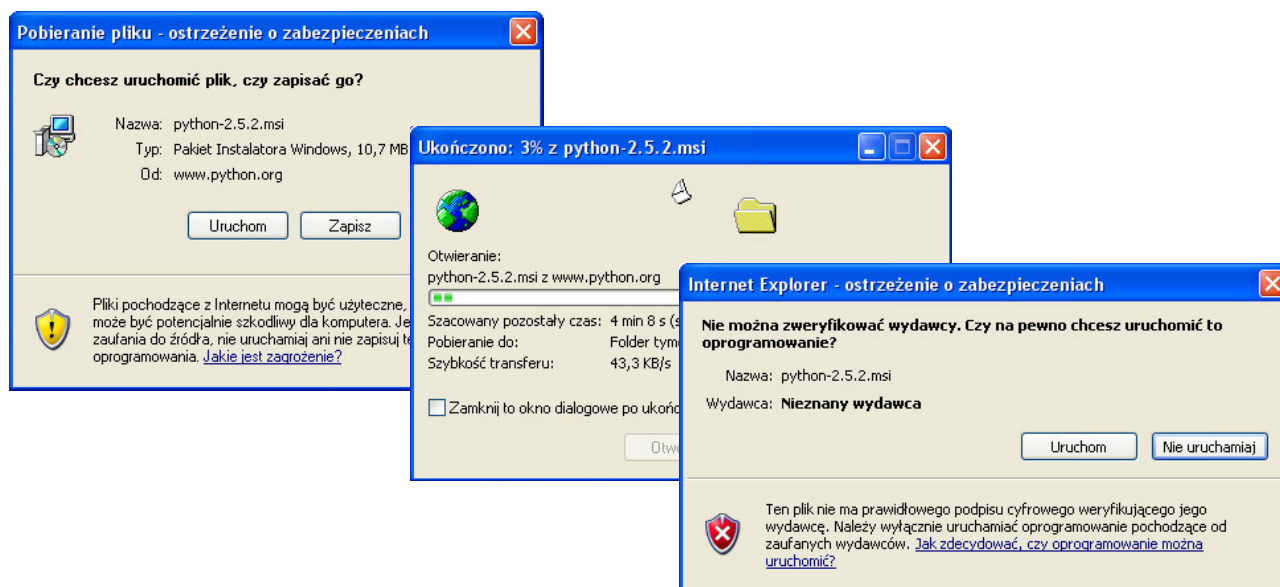


Rysunek 13.1.13 Strona, z której możesz pobrać instalator Pythona

Spośród wielu możliwości, które są na tej stronie, pobierz instalator dla Windows, bez kodu źródłowego (ten plik będzie najmniejszy).

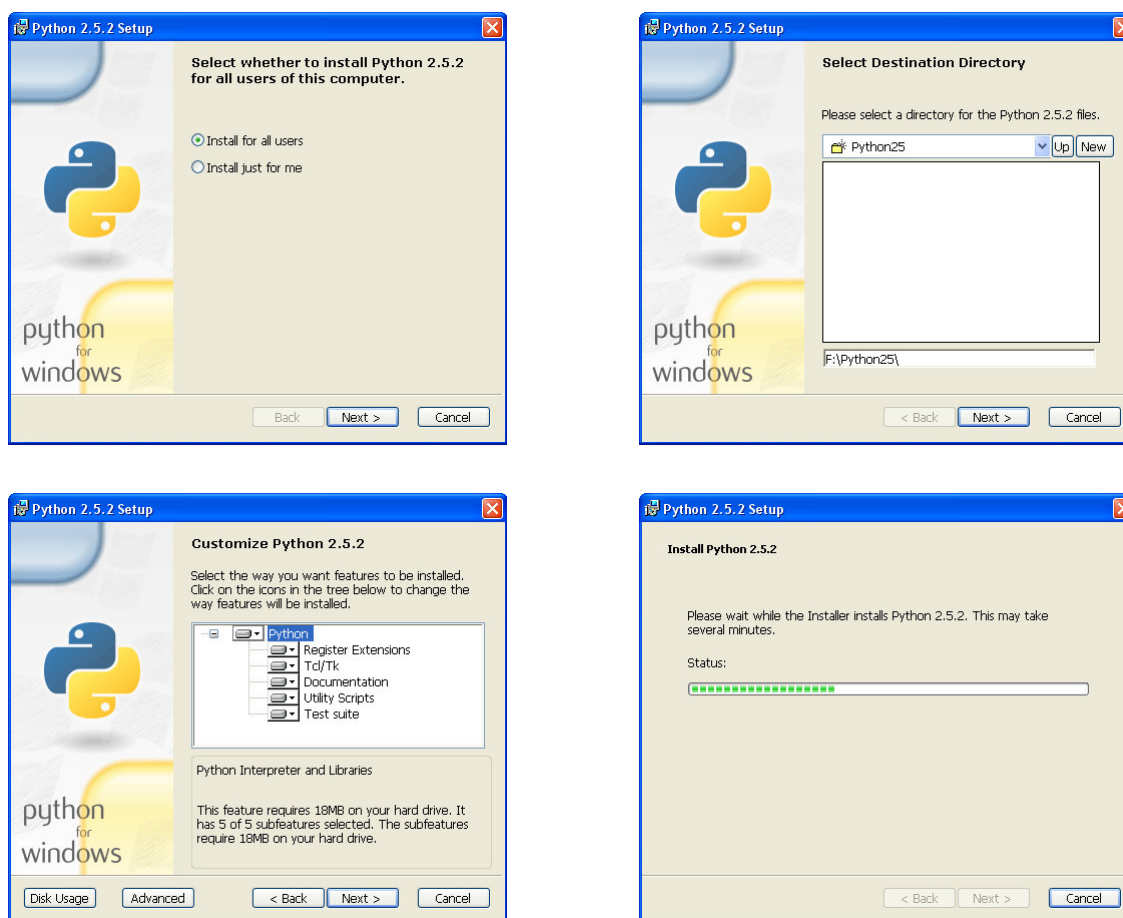
Plik instalacyjny Pythona masz także wśród plików towarzyszących tej książce (<http://www.samoloty3d.pl/downloads/wm/install/blender/python-2.5.2.msi>). Wolę jednak podać, skąd można pobrać nowsze wersje.

Jeżeli chcesz zainstalować plik wprost z Internetu — kliknij w jego skrót, i potwierdź, że chcesz go uruchomić (Rysunek 13.1.14):



Rysunek 13.1.14 Uruchomienie instalatora Pythona — wprost z Internetu

W instalatorze Pythona po prostu potwierdzaj wszystkie domyślne propozycje (Rysunek 13.1.15):



Rysunek 13.1.15 Python — instalacja pełnej wersji

Instalacja Pythona kończy się ekranem podziękowaniem dla Marka Hammonda (Rysunek 13.1.16):



Rysunek 13.1.16 Python — ekran finalny instalacji

(Zaglądałem na stronę tego autora, mogę potwierdzić, że zrobił bardzo wiele, by Python dobrze pasował do środowiska Windows).

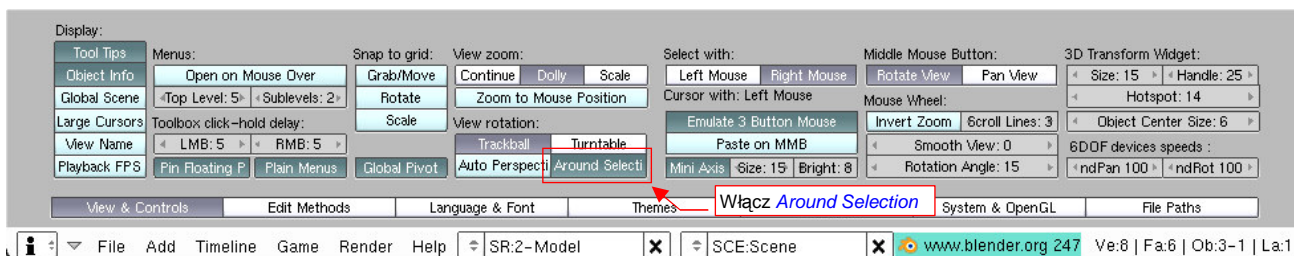
13.2 Ustawienie środowiska pracy

Wszelkie zmiany ustawień, jakich dokonasz w oknie *User Preferences*, zostaną zapamiętane tylko wtedy, gdy wywołasz polecenie **File→Save Default Settings**. Oznacza to, że te ustawienia nie są związane z jakimkolwiek aktualnym rysunkiem. Traktuj je jako konfigurację Twojej instalacji Blendera.

- Wszelkie zmiany w tej konfiguracji warto jest wykonywać po otwarciu Blendera bez żadnego konkretnego rysunku (pliku). Polecenie **File→Save Default Settings** zapisuje całą zawartość aktualnego stanu Blendera. Oznacza to, że jeżeli wywołasz je, gdy masz w oknie widoku jakiś model, to ten model będzie się od tej pory pojawiał jako domyślna zawartość każdego nowego pliku. Ustawienia, jakich teraz dokonasz w *User Preferences* i tak będą dotyczyły wszystkich rysunków, także tych, które zapisałeś już wcześniej.

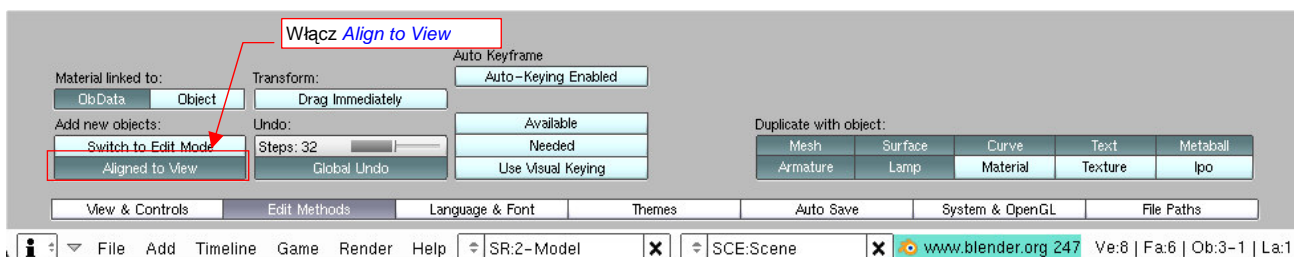
W tej sekcji pokażę, jakie domyślne ustawienia Blendera warto jest zmienić, aby ułatwić sobie pracę. Przejdę przez kolejne sekcje okna *User Preferences*, pomijając oczywiście te, w których nic nie trzeba zmieniać.

W sekcji *View & Controls* — włącz *View rotation: Around Selection* (Rysunek 13.2.1) . To ułatwi obracanie widoku wokół detali:



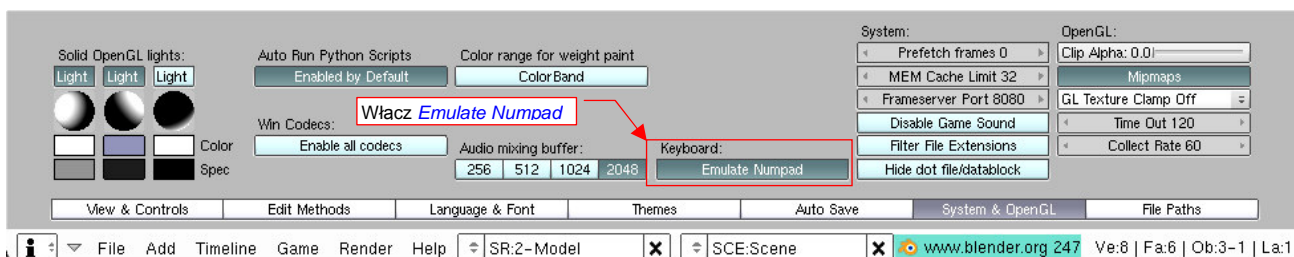
Rysunek 13.2.1 Ustawienia Blendera — sekcja *View & Controls*

W sekcji *Edit Methods* — włącz *Add new objects: Aligned to View* (Rysunek 13.2.2). Dzięki temu każdy obiekt, który stworzysz, powstanie w płaszczyźnie równoległej do aktualnego widoku. (Inaczej, aby odpowiednio zorientować nowo powstały element, miałbyś go dodatkowo obracać.)



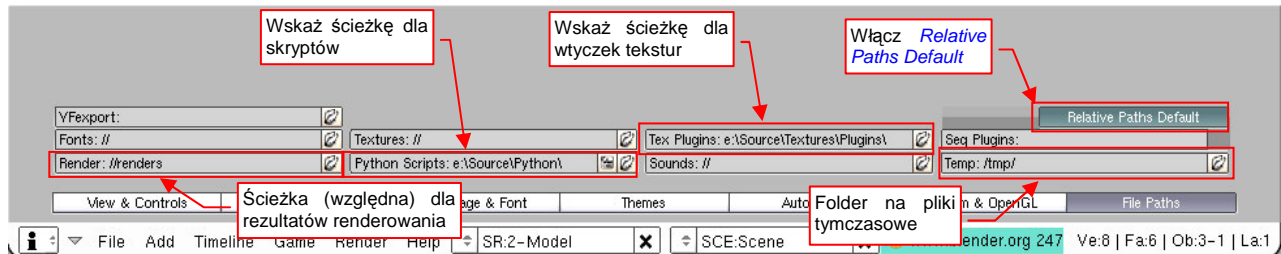
Rysunek 13.2.2 Ustawienia Blendera — sekcja *Edit Methods*

W sekcji *System & OpenGL* — włącz *Keyboard: Emulate Numpad* (Rysunek 13.2.3). Choć jest to praktyczne przede wszystkim dla notebooków, może okazać się przydatne także na tradycyjnych komputerach. (Pozwoli utrzymywać lewą rękę ponad centrum klawiatury, a nie przesuwając ponad całą jej szerokością).



Rysunek 13.2.3 Ustawienia Blendera — sekcja *System & OpenGL*

Wreszcie ustawienia domyślnych ścieżek, na których Blender ma poszukiwać różnych rodzajów plików. Służy do tego sekcja *File Paths* (Rysunek 13.2.4). Sądzę, że po pierwsze warto tu zaznaczyć opcję *Relative Paths Default*. Dzięki temu wszelkie referencje do plików obrazów (*images*) i innych danych, zostaną zapamiętane w pliku Blendera w sposób względny. Zdecydowanie to ułatwi przenoszenie całości w inne miejsca, lub tworzenie alternatywnych wersji modelu (każdej w we własnym folderze).



Rysunek 13.2.4 Ustawienia Blendera — sekcja *File Paths*

Dodatkowo — radziłbym na tym ekranie od razu ustawić kilka ścieżek. Będzie to potrzebne, aby poprawnie zadziałały dodatki do Blendera, opisane w kilku rozdziałach tej książki. Pobierz plik <http://www.samoloty3d.pl/downloads/wm/source.zip> i rozpakuj — powiedzmy, że zrobisz to do folderu *e:\Source*. Wskaż Blenderowi następujące foldery:

- *Python Scripts*: folder, w którym Blender wyszukuje dodatków napisanych w języku Python. Wskaż *e:\Source\Python*;
- *Tex Plugins*: folder, w którym Blender wyszukuje "wtyczek" pozwalających uzyskiwać specjalne efekty tekstur. Wskaż *e:\Source\Textures\Plugins*. (UWAGA: aby to ustawienie zadziałało, musisz także dopisać pełną ścieżkę do tego folderu do zmiennej środowiskowej *Path*¹);

• Na ilustracji (Rysunek 13.2.4) założyłem, że pliki, udostępnione wraz z tą książką, są na dysku *e:*².

• Jeżeli Blender nie będzie potrafił, mimo wszystko, znaleźć wtyczki w folderze *Texture Plugins* — umieść jej plik w tym samym folderze, co rysunek Blendera. (Tak się może zdarzyć, jeżeli nie dopisziesz poprawnie tego folderu do zmiennej środowiskowej *Path*.)

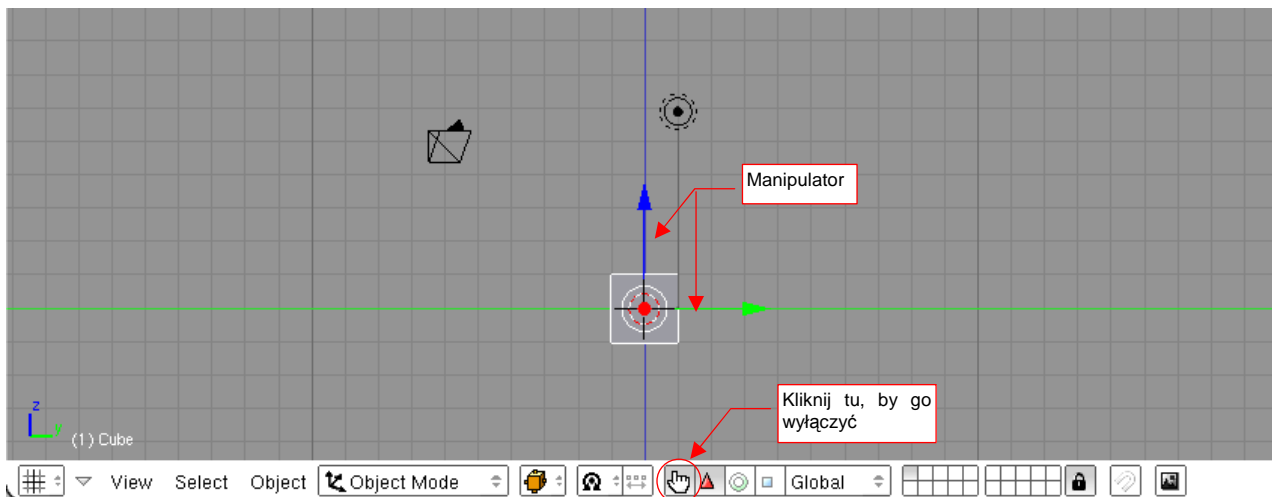
Ostatnią, opcjonalną ścieżką do ustalenia w sekcji *File Paths* jest *Renderers* (Rysunek 13.2.4). To domyślny folder, w którym zostaną umieszczone wyniki renderowania — np. pliki animacji. Zwróć uwagę, że wpisałem go w sposób względny — *//renders*. Podwójny ukośnik na początku oznacza aktualny folder rysunku. (To ustawienie sprawi, że wszystkie rezultaty renderowania będą zapisywane w podkatalogu o nazwie *renders*, utworzonym w aktualnym folderze rysunku.)

Teraz — trochę nadmiarowo — wykonajmy jeszcze parę wstępnych zmian w domyślnym oknie widoku. Nie musimy tego robić, ale często łąpałem na tym, że ciągle w ten sam sposób poprawiam zawartość nowo utworzonego rysunku. Po co to robić? Zapiszmy poprawioną postać rysunku domyślnego, i oszczędźmy sobie tej roboty.

¹ Zmienne środowiskowe to lista prowadzona przez system Windows (sam pomysł zaczerpnięto jeszcze z Unixa). Dochodzi się do nich poprzez *Mój komputer* → *Właściwości*. W oknie dialogowym *Właściwości systemu*, w zakładce *Zaawansowane* naciskasz przycisk *Zmienne środowiskowe*. W oknie dialogowym *Zmienne środowiskowe*, które się otworzy, zmienną *Path* znajdziesz w dolnej liście (*Zmienne systemowe*). Naciśnij przycisk *Edytuj* i, po średniku, dopisz do jej wartości ścieżkę *e:\Source\Textures\Plugins*. Przykład nadawania wartości zmiennej środowiskowej znajdziesz na str. 24, a zmianę *Path* — na str. 1018.

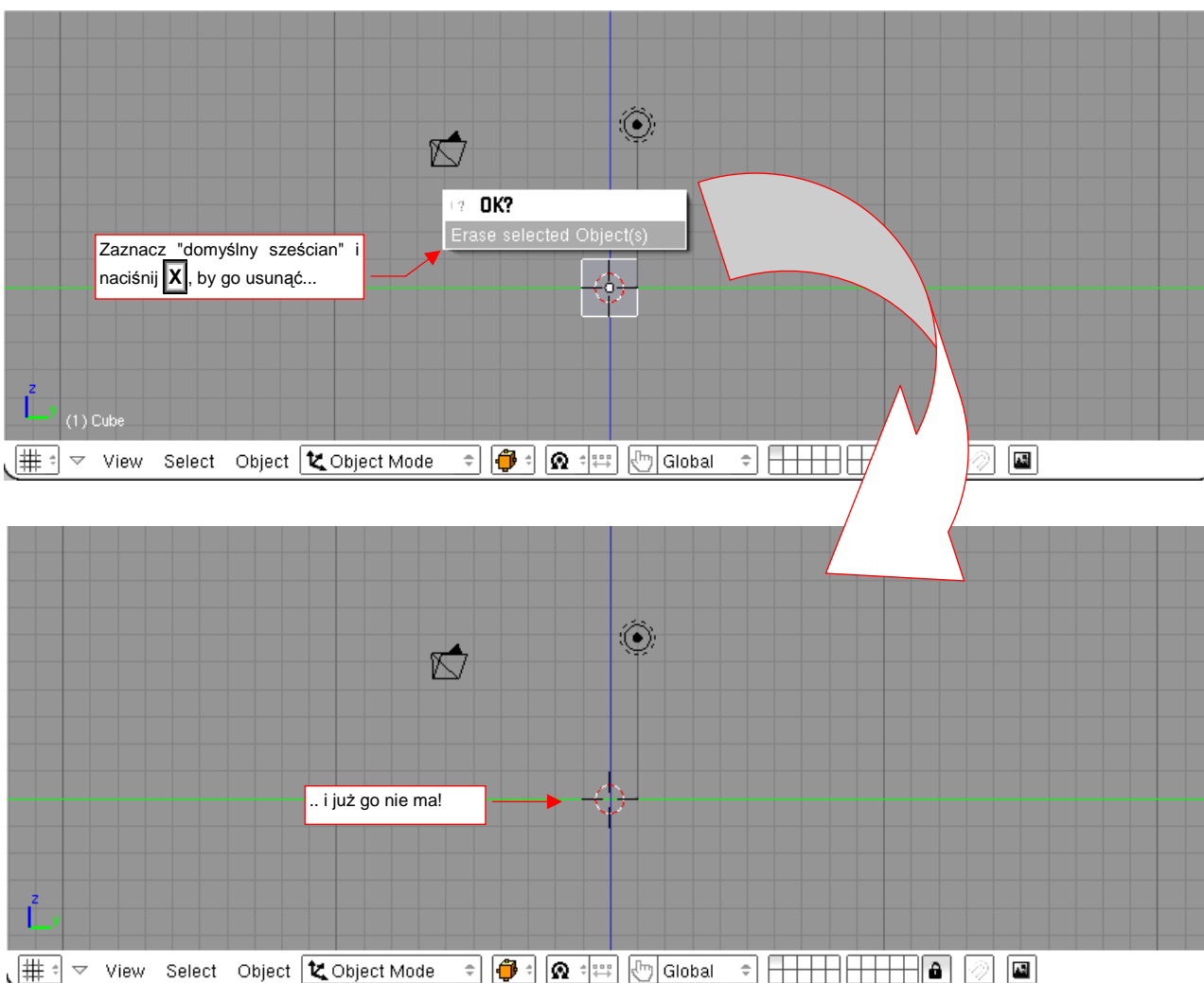
² Podstaw w miejsce tej litery ścieżkę dostępu do folderu, gdzie zapisałeś te pliki. Użyłem tu litery "e:" tylko dlatego, że to najkrótsza do zapisania ścieżka.

Zacznijmy od wyłączenia manipulatora (Rysunek 13.2.5). (Mówiąc szczerze, nigdy go nie używałem.):



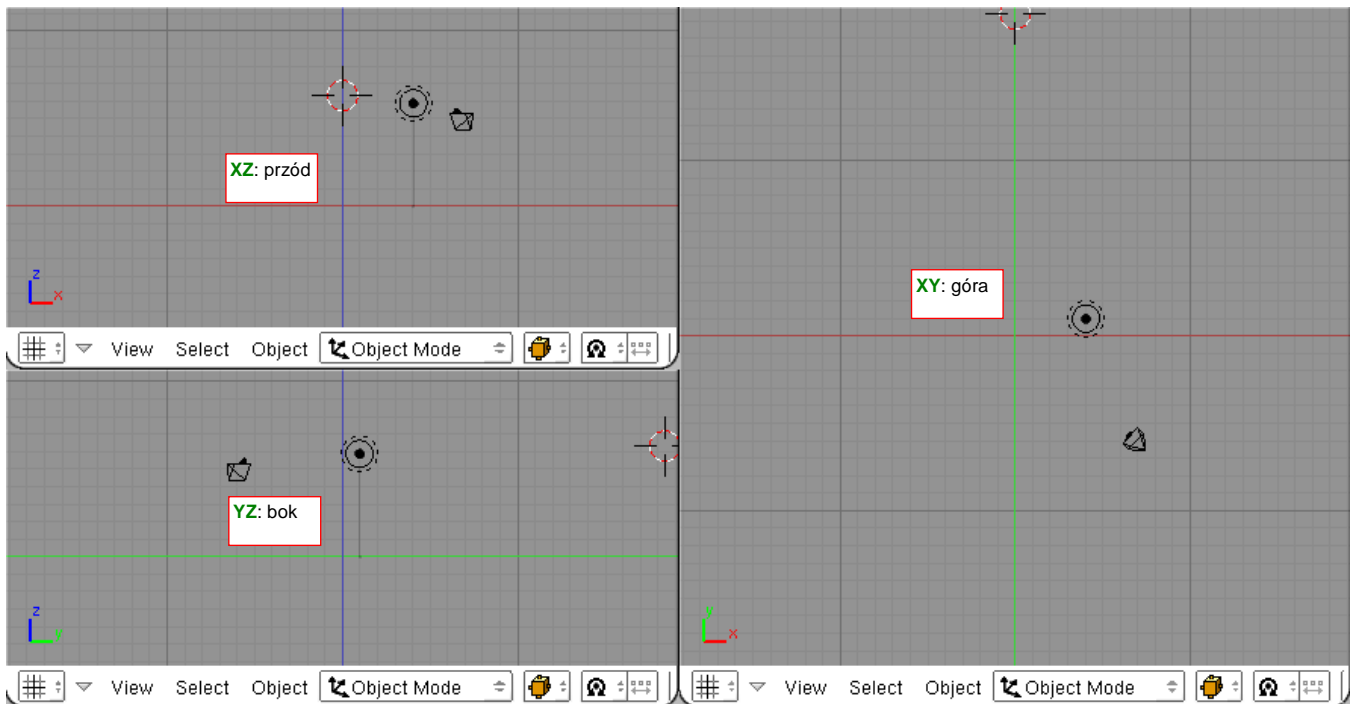
Rysunek 13.2.5 Wyłączenie manipulatora

Zróbmy coś jeszcze — usuńmy domyślne "pudełko", które Blender domyślnie umieszcza na każdym nowym rysunku. (Zawsze pierwszą rzeczą, którą robiłem, było pozbycie się tego obiektu.) Zaznacz go i naciśnij **X** (*Object* → *Delete*). W oknie dialogowym, które się pojawi, potwierdź *Erase selected Object(s)* (Rysunek 13.2.6):



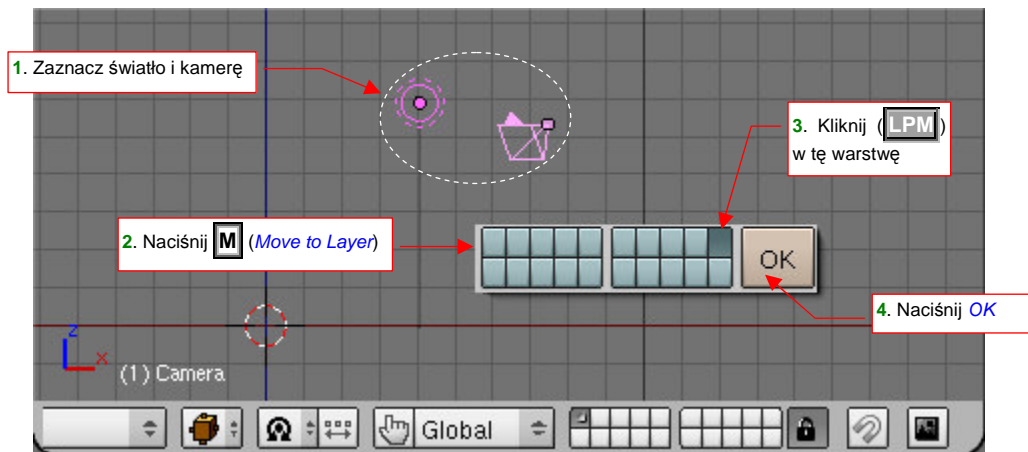
Rysunek 13.2.6 Usunięcie "domyślnego sześcianu"

Na koniec — podzieliśmy okno widoku (poleceniem [Split Area](#) — p. str. 57) na trzy okna widoku. Każde z nich z odrębnym rzutem: z przodu, z boku, i z góry (Rysunek 13.2.7):



Rysunek 13.2.7 Podział ekranu na trzy rzuty

Pozostało już tylko przesunąć kamerę i źródło światła ("lampę") na ostatnie warstwy rysunku. (To czynność opcjonalna — wypraktykowałem, że w ten sposób nie zaznaczysz ich przypadkowo w ferworze formowania samolotu). Zaznacz obydwa obiekty i naciśnij **M** ([Object→Move to Layer](#)). Rysunek 13.2.8 pokazuje, jak to masz zrobić, i o którą warstwę chodzi:



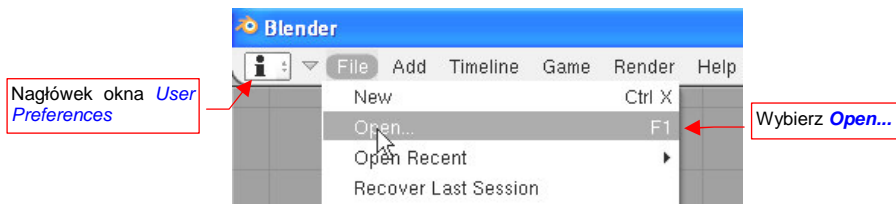
Rysunek 13.2.8 Przeniesienie kamery i źródła światła na warstwę 10

W wyniku zmiany warstwy kamera i światło znikną z rysunku (gdyż warstwa 10 ma aktualnie wyłączoną widoczność). Nie przejmuj się tym, one tam są i spokojnie czekają aż do nich wrócisz!

Pozostaje już tylko zapisać te ustawienia jako domyślne: [File→Save Default Settings](#).

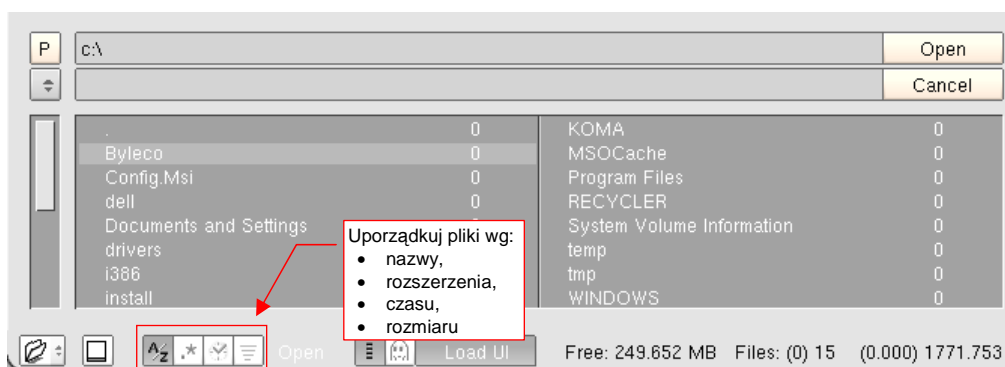
13.3 Otwieranie pliku

Wybierz z menu Blendera (nagłówek okna *User Preferences*) polecenie *File→Open* (lub naciśnij **F1**) (Rysunek 13.3.1):



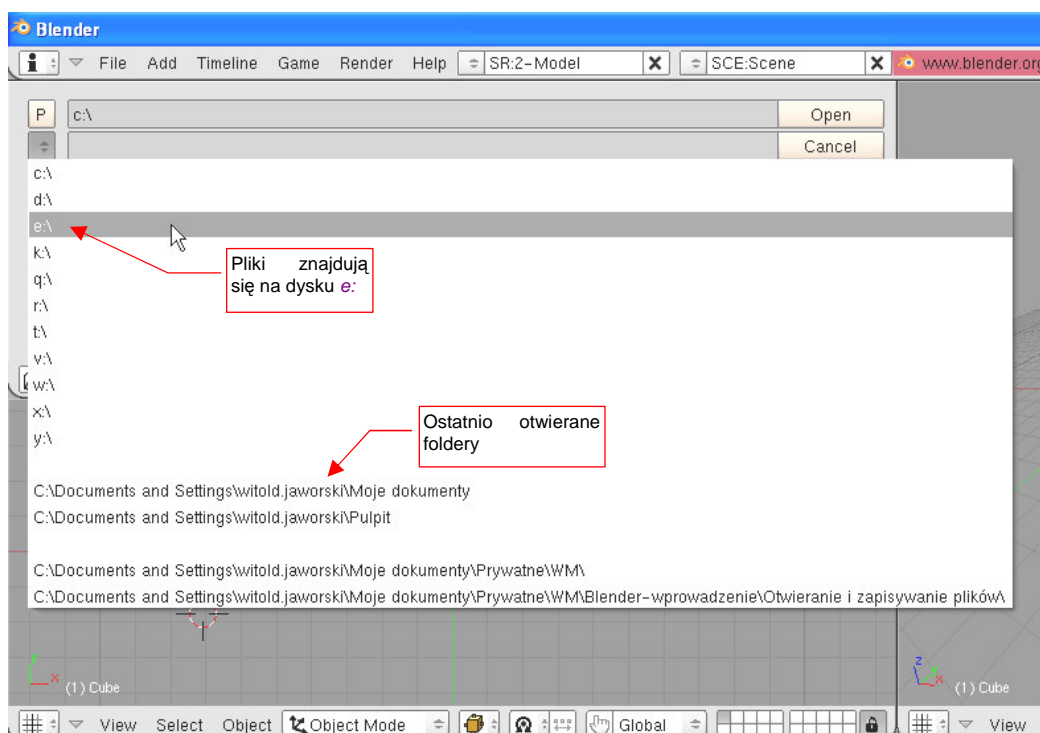
Rysunek 13.3.1 Otwieranie pliku

Spowoduje to chwilowe przekształcenie aktywnego okna w okno wyboru plików (*File Browser Window*) – Rysunek 13.3.2:



Rysunek 13.3.2 Okno wyboru pliku

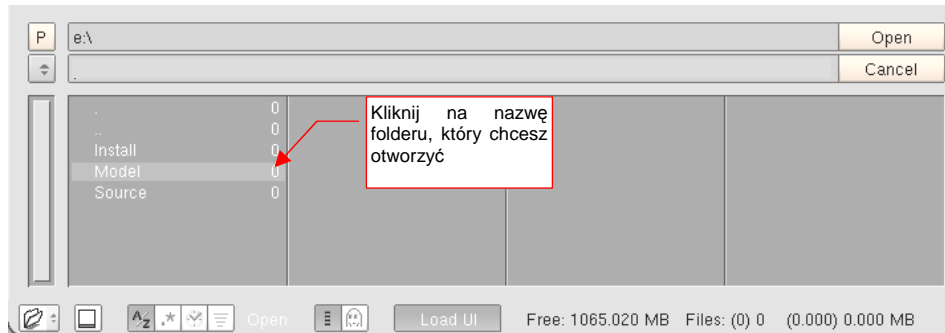
Jak widać, Blender ma własne okno do wskazywania pliku – jest to konsekwencja decyzji o zastosowaniu własnego interfejsu użytkownika. Z listy rozwijalnej po lewej stronie można wybrać napęd (lub ostatnio otwierany folder) (Rysunek 13.3.3):



Rysunek 13.3.3 Lista podpowiedzi ścieżek (dysków lub folderów)

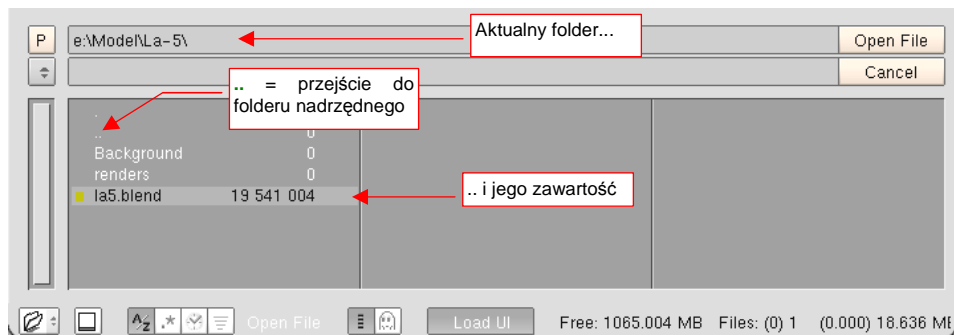
Pobierz towarzyszący tej książce plik <http://www.samoloty3d.pl/downloads/wm/model/la5.zip>. Rozpakuj jego zawartość do folderu o nazwie, powiedzmy, *Model*. Teraz otworzymy plik *Model\la5\la5.blend*.

Założmy, że na potrzeby tego przykładu pliki te umieściłeś na jakimś dysku *E:*. Otwórz w oknie wyboru plików ten napęd (Rysunek 13.3.4):



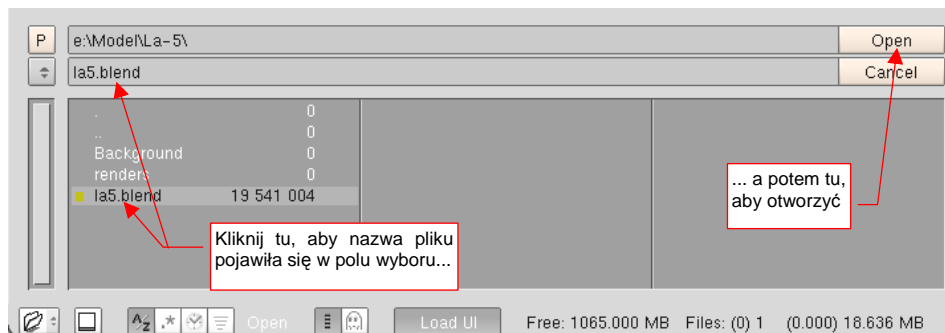
Rysunek 13.3.4 Nawigacja w oknie wyboru plików

Wystarczy kliknąć (LPM) w nazwę folderu *Model*, aby go otworzyć. W podobny sposób przejdź do folderu *La-5* (Rysunek 13.3.5):



Rysunek 13.3.5 Zawartość folderu *Model*

Kliknij w plik, który chcesz otworzyć – jego nazwa pojawi się w polu na nazwę pliku do otwarcia (Rysunek 13.3.6). (Można było, oczywiście, wpisać ją z klawiatury, ale komu by się chciało...):



Rysunek 13.3.6 Otwieranie pliku Blendera

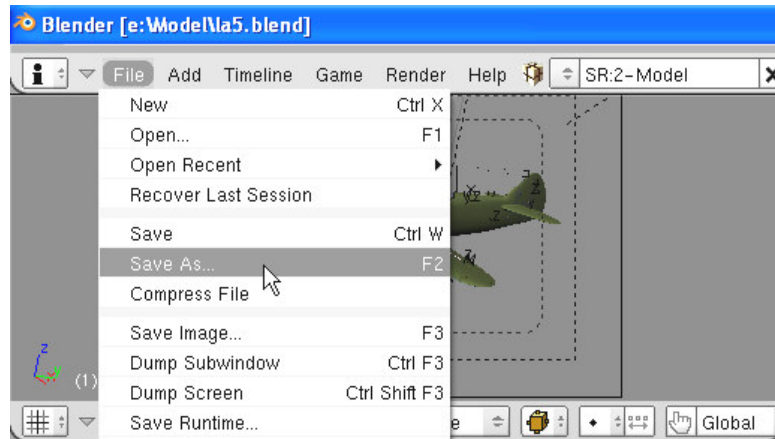
Po naciśnięciu przycisku **Open** plik zostanie załadowany. W zależności od mocy obliczeniowej Twojego komputera, może to zająć Blenderowi sekundę lub dwie.

Z innych poleceń z menu **File** (okna [User Preferences](#)), które mogą okazać się przydatne, warto wspomnieć:

- **File→Open Recent**: pozwala otworzyć jeden z dziesięciu ostatnio zapisywanych plików;
- **File→Recover Last Session**: pozwala odzyskać plik w wersji sprzed ostatniego zapisu. (Blender zawsze tworzy „cień” oryginalnego pliku – poprzednią wersję zapamiętuje w kopii o rozszerzeniu [.blend1](#)). Więcej o odtwarzaniu ostatnich wersji danych — p. str. 769;

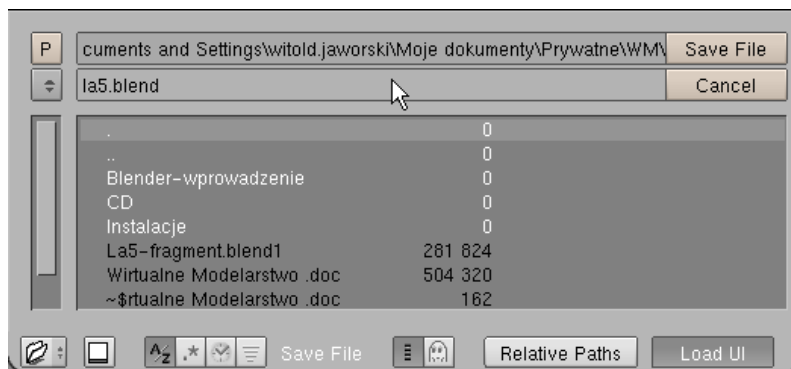
13.4 Zapisanie pliku

Aby zapisać otwarty plik pod inną nazwą, wybieramy polecenie **File→Save As** (Rysunek 13.4.1). Alternatywnie możesz nacisnąć na klawiaturze **F2** — to skrót do tego polecenia.



Rysunek 13.4.1 Polecenie - zapisanie pliku pod inną nazwą

Analogicznie jak przy otwarciu pliku, aktywne okno na ekranie zostanie na chwilę zastąpione oknem wyboru pliku (Rysunek 13.4.2). Otwierając kolejne foldery, przejdź do miejsca, gdzie chcesz zachować aktualnie otwartą pracę.



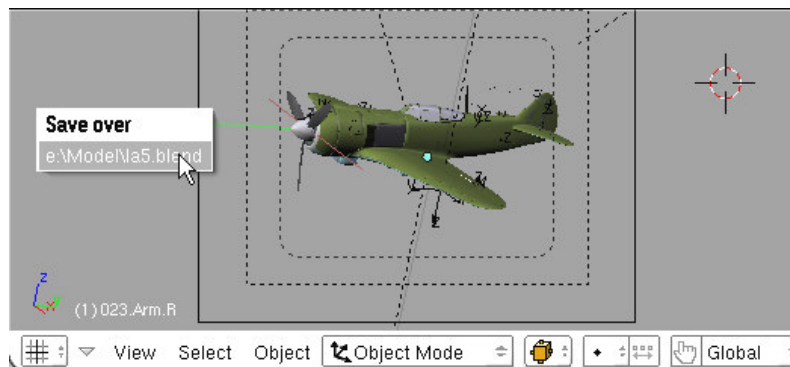
Rysunek 13.4.2 Okno wyboru nazwy pliku do zapisu

Zwróć uwagę, że w polu z nazwą do zapisania przez cały czas pozostaje oryginalna nazwa pliku. Możesz ją zmienić, wpisując nową z klawiatury (lub po prostu klikając w inny plik, jeżeli chcesz go nadpisać).

Zapis pod nową nazwą nastąpi po naciśnięciu przycisku **Save File**.

Osobiście używałem tego polecenia co najmniej raz dziennie – do zapamiętania kolejnej wersji pliku. Nazywałem to „wersją dzienną”. Takie wersje bywają przydatne. Na początku pracy z Blenderem, zdarzało mi się poprzestawiać dziesiątki parametrów i nie uzyskać oczekiwanego efektu. Czułem, że szukając rozwiązania zabrnąłem w „ślepią uliczkę”. Wycofanie dokonanych zmian do stanu początkowego wyglądało często na bardzo pracochłonne, a czasami na praktycznie niemożliwe. O wiele prościej było zacząć jeszcze raz od „czystej” wersji z poprzedniego dnia. Twórcy Blendera z myślą o takich jak ja, dodali do okna zapisu pliku (Rysunek 13.4.2) dodatkowy „gadżet”. Klawisze **+** i **-** z klawiatury numerycznej zwiększają lub zmniejszają cyfrę (numer wersji) na końcu nazwy pliku.

Aby zapisać plik bez zmiany nazwy, wybierz polecenie **File→Save** (albo **Ctrl-W**). Spowoduje to pojawienie się okna dialogowego **Save over**, w którym Blender upewnia się, że wiesz, co robisz (Rysunek 13.4.3).



Rysunek 13.4.3 Potwierdzenie zapisania pliku

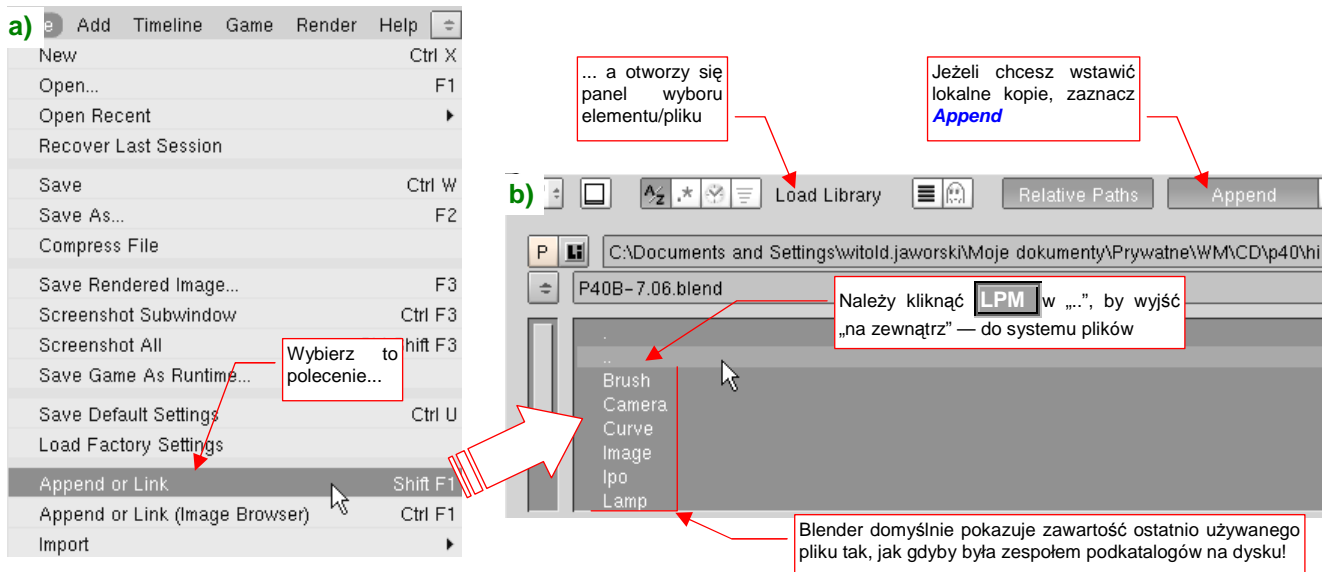
- Pamiętaj: zapis nastąpi **tylko** wtedy, gdy klikniesz myszką (**LPM**) w nazwę pliku, wyświetloną w oknie **Save over!**

Z innych poleceń z menu **File** (okna **User Preferences**), które mogą okazać się przydatne, warto wspomnieć:

- **File→Dump Subwindow**: najczęściej wywoływane nie z menu, ale za pomocą skrótu: **Ctrl-F3**. Pozwala zapisać na dysku obraz przedstawiający aktywne okno Blendera. Format obrazu (JPG, PNG, ...) – ustalamy w oknie przycisków (**Buttons Window**), zestaw **Scene**, podzestaw **Render Buttons**, panel **Format**. (patrz str. 87);
- **File→Dump Screen**: najczęściej wywoływane nie z menu, ale za pomocą skrótu: **Ctrl-Shift-F3**. Działa tak samo jak **Dump Subwindow**, ale zapisuje wszystkie widoczne okna – cały ekran Blendera
- **File→Save Rendered Image** (**F3**): zapisuje wyrenderowany (w oddzielnym oknie) finalny obraz. Podobnie jak w **File→Dump Subwindow/Dump Screen**, format obrazu jest ustalany na panelu **Format**.

13.5 Import obiektów z innego pliku Blendera

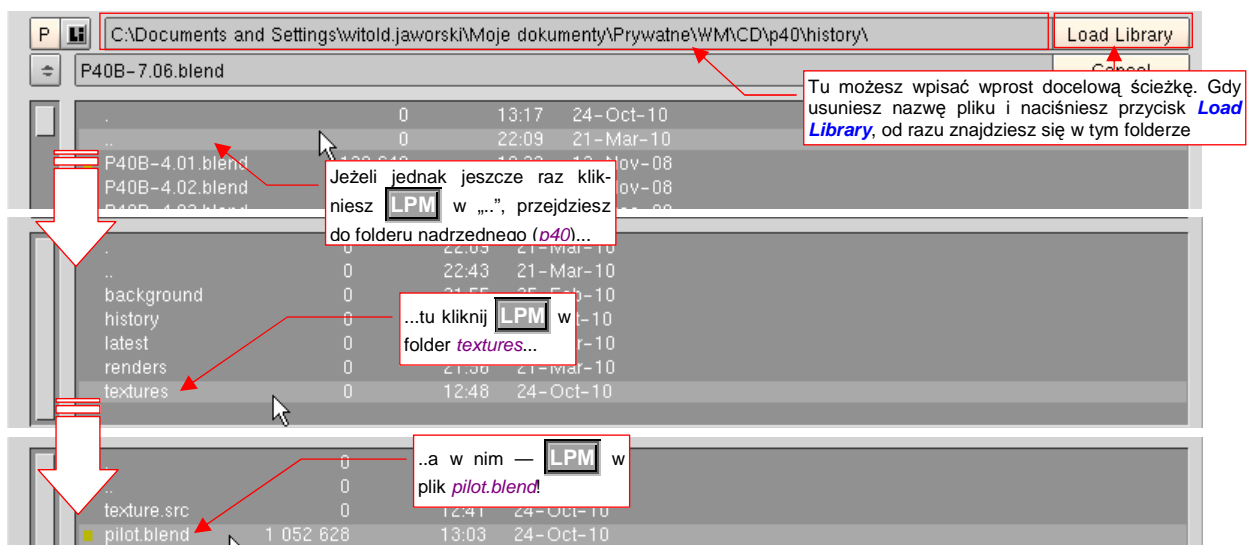
Aby wstawić do aktualnej sceny obiekty z innego pliku, wybierz polecenie **File→Append or Link** (Rysunek 13.5.1a). Alternatywnie możesz nacisnąć na klawiaturze **Shift-F1** — to skrót do tego polecenia:



Rysunek 13.5.1 Polecenie - import elementów innego pliku

Tak jak przy otwarciu pliku, aktywne okno na ekranie zostanie na chwilę zastąpione oknem **Load Library**, (Rysunek 13.5.1b) przypominającym do złudzenia okno wyboru plików. Gdy jednak przyjrzy się uważnie wyświetlonym „folderom” zauważysz, że nie są to katalogi na dysku, ale nazwy klas obiektów, przechowywanych w rysunku Blendera. To taka konwencja - zawartość każdego pliku **.blend* jest wyświetlana, jak gdyby jej struktura była przedłużeniem struktury folderów i plików na dysku.

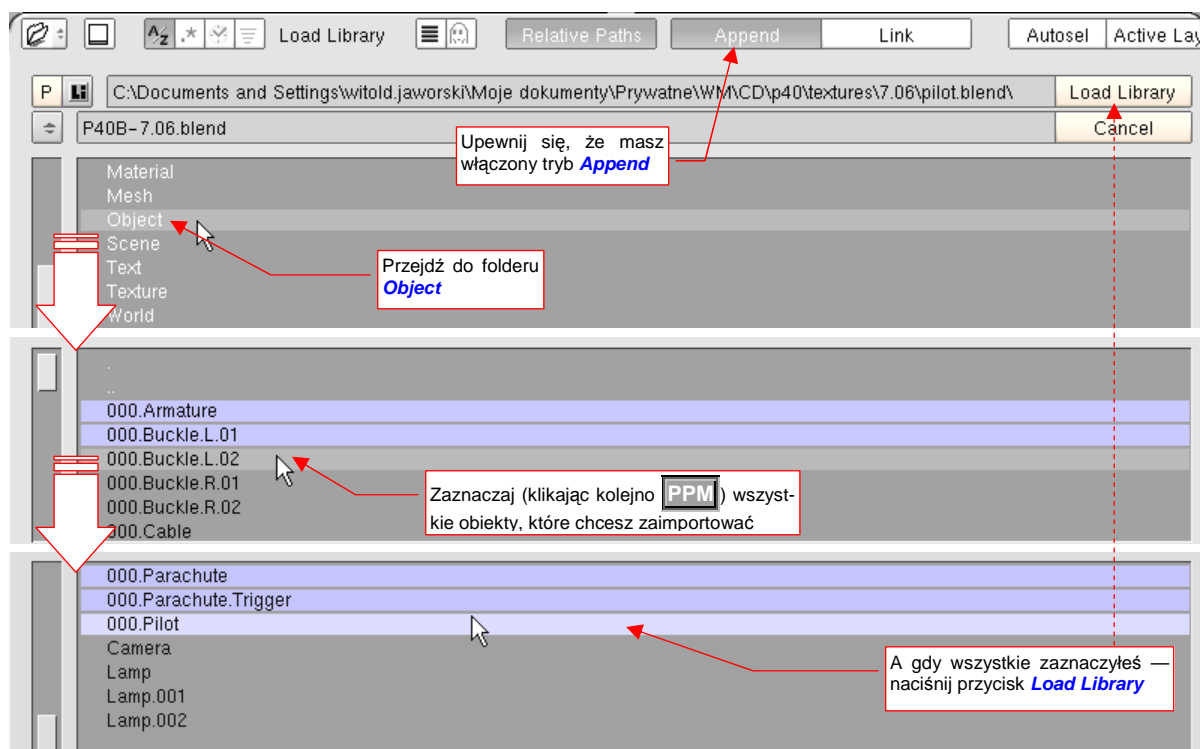
Domyślnie oknie **Load Library** wyświetlany jest ostatnio użyty plik. Załóżmy, że chcemy zaimportować figurę pilota z pliku *p40\textures\7.06\pilot.blend*. Zaczynaj więc od wyjścia z aktualnego pliku „na zewnątrz” — należy kliknąć w pseudofolder „...” (Rysunek 13.5.1b). Znajdziesz się wówczas np. w folderze *p40\history*, w którym zobaczysz inne pliki (Rysunek 13.5.2) :



Rysunek 13.5.2 Nawigacja myszką w systemie plików

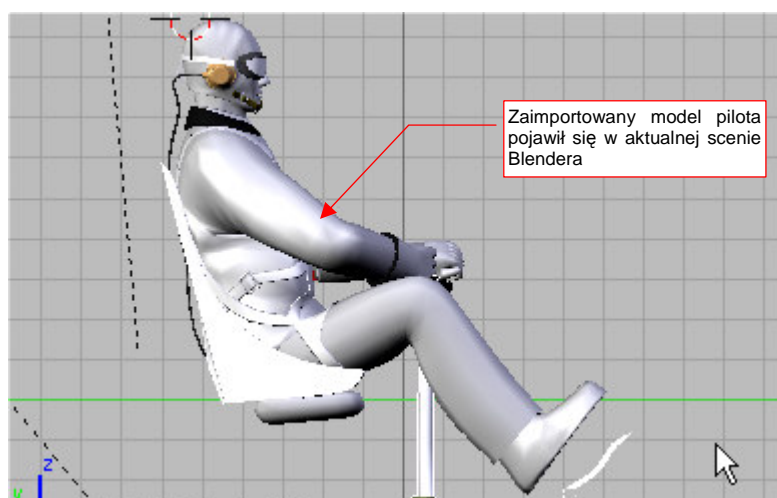
Jeżeli wolisz korzystać z klawiatury, możesz od razu wpisać w pole u góry okna docelowy folder (*../textures/7.06*), a w pole poniżej — nazwę pliku (*pilot.blend*). Potem wystarczy nacisnąć przycisk **Load Library**. Alternatywnie możesz także posłużyć się myszką, przechodząc po strukturze katalogów na dysku.

Gdy „wejdiesz do środka” pliku *pilot.blend* (klikając **LPM**), upewnij się, że zaznaczyłeś przełącznik **Append** (Rysunek 13.5.3). Włączenie tego trybu oznacza, że wstawisz do aktualnej sceny kopie importowanych elementów¹. Potem przejdź do folderu **Object** (zazwyczaj importujemy obiekty — Blender sam zadba o to, by przenieśli się za nimi wszystkie siatki, materiały i tekstury):



Rysunek 13.5.3 Wybór elementów do importu

W środku folderu **Object** zobaczysz wszystkie obiekty, znajdujące się w *pilot.blend*. Aby ułatwić wybór, elementom składającym się na model pilota nadałem przedrostek „000.”. Pozaznaczaj je, klikając w każdy **PPM**. Potem wystarczy nacisnąć przycisk **Load Library** i pilot znalazł się w aktualnej scenie Blendera (Rysunek 13.5.4):



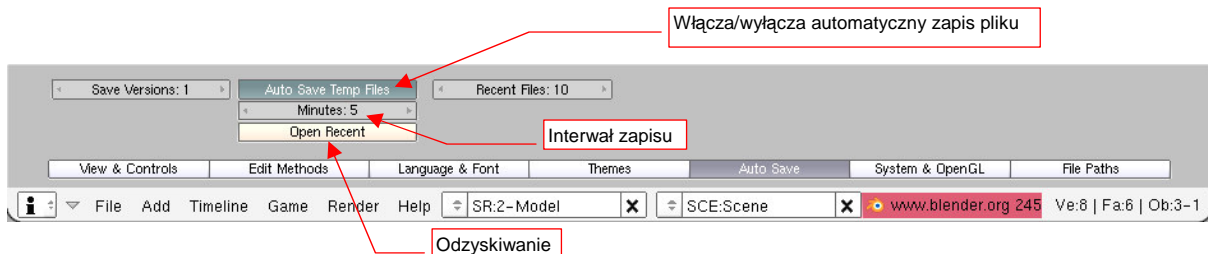
Rysunek 13.5.4 Model pilota, wczytany do aktualnej sceny

¹ Gdybyś wybrał alternatywną opcję **Link** — do aktualnej sceny wstawione zostałyby tylko „odsyłacze” (referencje) do obiektów umieszczonych we wskazanym pliku Blendera (w tym przykładzie byłby to *pilot.blend*). Możesz w ten sposób wykorzystywać pewne pliki *.blend jako biblioteki. Przechowywałbyś w nich wzorce elementów wykorzystywanych w wielu różnych projektach. Gdybyś zmienił definicję takiego „odsyłacza” — uległby zmianie wszędzie tam, gdzie jest użyty. Podłączając w ten sposób możesz nie tylko całe obiekty, ale — na przykład — same siatki. Wymaga to jednak przemyślanej struktury projektu. Poza tym — nie można modyfikować takich „podłączonych” elementów tam, gdzie są użyte. W praktyce często kończy się to i tak stworzeniem ich lokalnej kopii.

13.6 Kopia bezpieczeństwa i odtwarzanie

W tej sekcji opiszę sposoby „awaryjnego” odzyskiwaniu pliku. Uczciwie mogę powiedzieć, że przez długi czas (pierwszych 7-8 miesięcy pracy) Blender zachowywał mi się bardzo stabilnie. Na palcach jednej ręki mógłbym policzyć sytuacje, gdy straciłem choć trochę pracy z powodu niespodziewanego zakończenia przez błąd w programie. Potem, gdy model przekroczył ok. 500 tys. ścian, bywało, że czasami Blender mi się nagle „wysypał”¹. W każdym razie – istnieje pewien sposób, pozwalający odzyskać choć część pracy od ostatniego zapisu.

W oknie **User Preferences** przejdź do sekcji **Auto Save** (Rysunek 13.6.1):



Rysunek 13.6.1 Panel automatycznego zapisu

Ważne są tu trzy kontrolki:

- Przełącznik **Auto Save Temp Files**: moja rada: pozostaw go zawsze włączonym, jak na ilustracji;
- Pole numeryczne **Minutes**: tu wpisz, co ile minut ma być zachowywany stan pliku;
- Przycisk **Open Recent**: Załaduje ostatni stan pliku (uwaga: dopóki nie opuścisz Blendera!).

Gdy Blender nagle zgłosił „fatalny” błąd i został zamknięty przez system operacyjny, nie popadaj w rozpacz. Jeżeli miałeś włączony **Auto Save**, możesz jeszcze odzyskać swoją pracę. Uruchom ponownie Blender. Najprawdopodobniej gdy naciśniesz przycisk **Open Recent** ujrzysz komunikat w rodzaju „*Loading C:\tmp2020.blend failed: unable to open*”. To normalne – nazwą pliku tymczasowego jest numer procesu, przypisany Blenderowi przez system operacyjny. Teraz Blender jest uruchomiony ponownie, więc ma inny numer. Tym niemniej nasze dane można jeszcze odzyskać ręcznie.

W folderze, który Blender podaje w komunikacie – w tym przykładzie to „*C:\tmp*”, ale na Twoim komputerze może być inny² – znajdziesz co najmniej kilka plików Blendera. Wszystkie mają nazwy numeryczne i rozszerzenie **.blend*, np. „1140.blend”, lub podobne. To właśnie efekt automatycznego zachowywania. Uporządkuj je malejąco według daty. Jeżeli widzisz jakiś plik z ostatnich kilkunastu minut – spróbuj go otworzyć w Blenderze. Może się okazać, że to jest Twój model, zachowany jakiś czas temu! Zapisz go pod inną nazwą, jako kolejną wersję swego pliku, otwórz za pomocą Blendera, i pracuj dalej. Robiłem tak niejednym razem. Pliki, odzyskane w ten sposób, są zupełnie normalnymi plikami – bez żadnych luk w danych, czy skaz w zapisie.

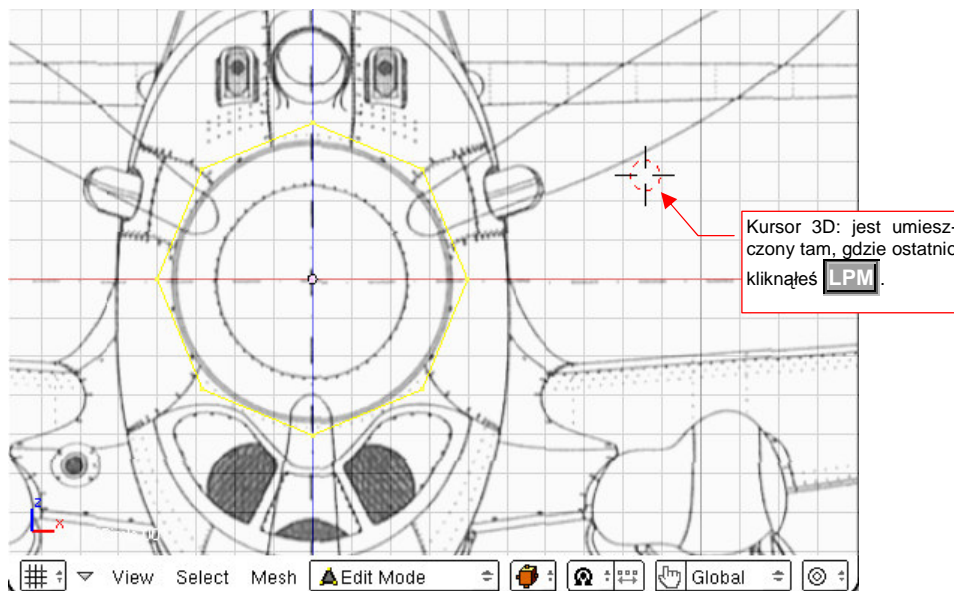
¹ Zaobserwowałem, że błąd w Blenderze występował zazwyczaj po wycofaniu poleceniem **Undo** kilku zmian w edytorze UV (*UV/Image Editor Window*). To było w wersji 2.45. Potem pojawiła się wersja 2.46, w której ten fragment uległ gruntownym przeróbkom. Miejmy nadzieję, że wyeliminowały one także tą przypadłość.

² Patrz na str. 759, Rysunek 13.2.4: folder, w którym znajdują się pliki tymczasowe, jest tam wpisany w polu *Temp*. Pojedynczy ukośnik na początku ścieżki jest konwencją przejętą z Unixa. W praktyce Windows *"/tmp/* oznacza *"C:\tmp*"

13.7 Kursor 3D — ustalanie położenia

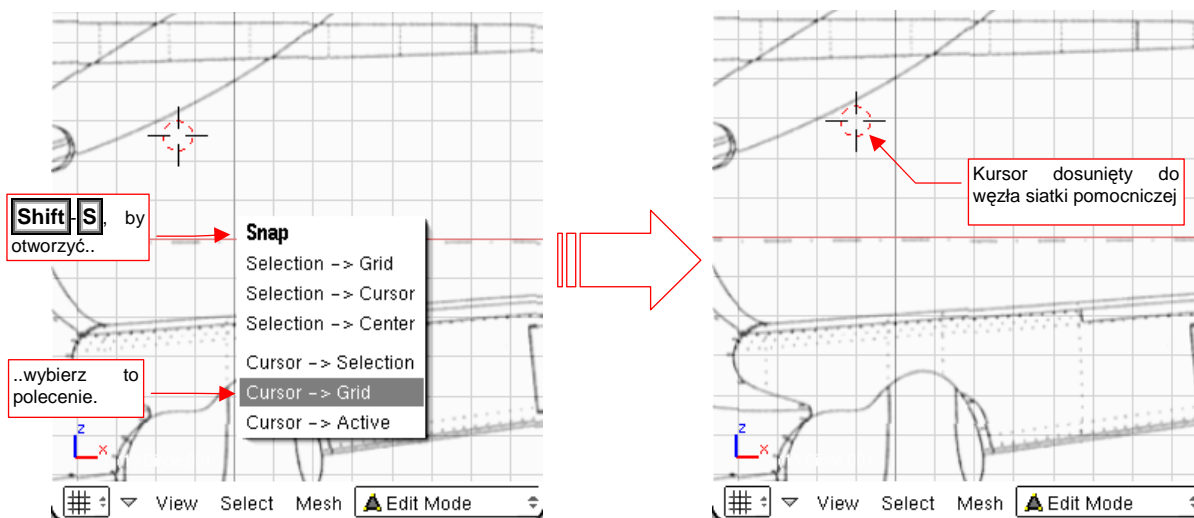
Kursor 3D (*3D Cursor*) jest w Blenderze specjalnym punktem w przestrzeni. Wyznacza "aktualną pozycję", która jest używana podczas dodawania do sceny nowych obiektów. Może być także wykorzystany jako punkt odniesienia podczas obrotu, zmiany skali, lub odbicia lustrzanego (*Mirror*).

Położenie kursora 3D jest oznaczane niewielkim krzyżem, otoczonym czerwono-białą obwódką (Rysunek 13.7.1):



Rysunek 13.7.1 Kursor 3D na scenie

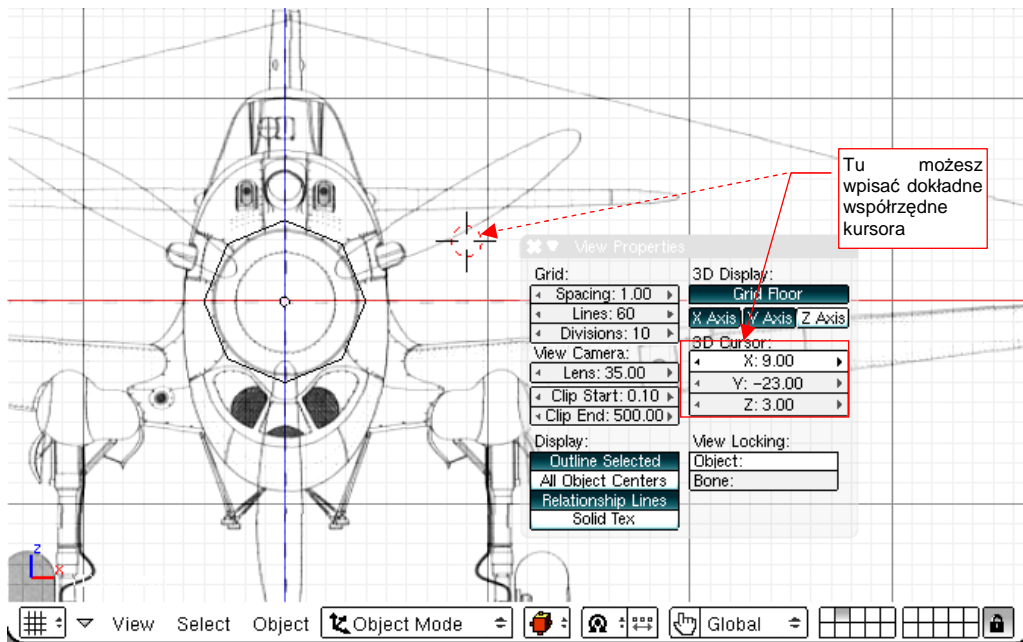
Kliknięcie **LPM** można traktować jako pierwszą, przybliżoną metodę wskazania położenia kursora. Z oczywistych przyczyn nie jest specjalnie dokładnie. Potem możesz zawsze dosunąć kursor do najbliższego węzła "kratki pomocniczej" (*grid*), która jest naniesiona na płaszczyznę widoku. Naciśnij w tym celu **Shift-S** (*Object*→*Snap*), i wybierz z menu *Snap* polecenie *Cursor*→*Grid* (Rysunek 13.7.2):



Rysunek 13.7.2 Dosunięcie kursora do węzła siatki pomocniczej (*grid*)

- W podobny sposób możesz także wybrać polecenie *Cursor*→*Selection*. W trybie obiektu (*Object Mode*) kursor zostanie umieszczony w środku (tzn. punkcie odniesienia) aktualnie zaznaczonego obiektu. W trybie edycji siatki (*Edit Mode*) kursor przesunie się do aktualnie zaznaczonego wierzchołka

Aby dokładnie wpisać położenie kursora — wywołaj polecenie **View→View Properties**. Spowoduje to pojawienie się na ekranie okna dialogowego **View Properties** (Rysunek 13.7.3):

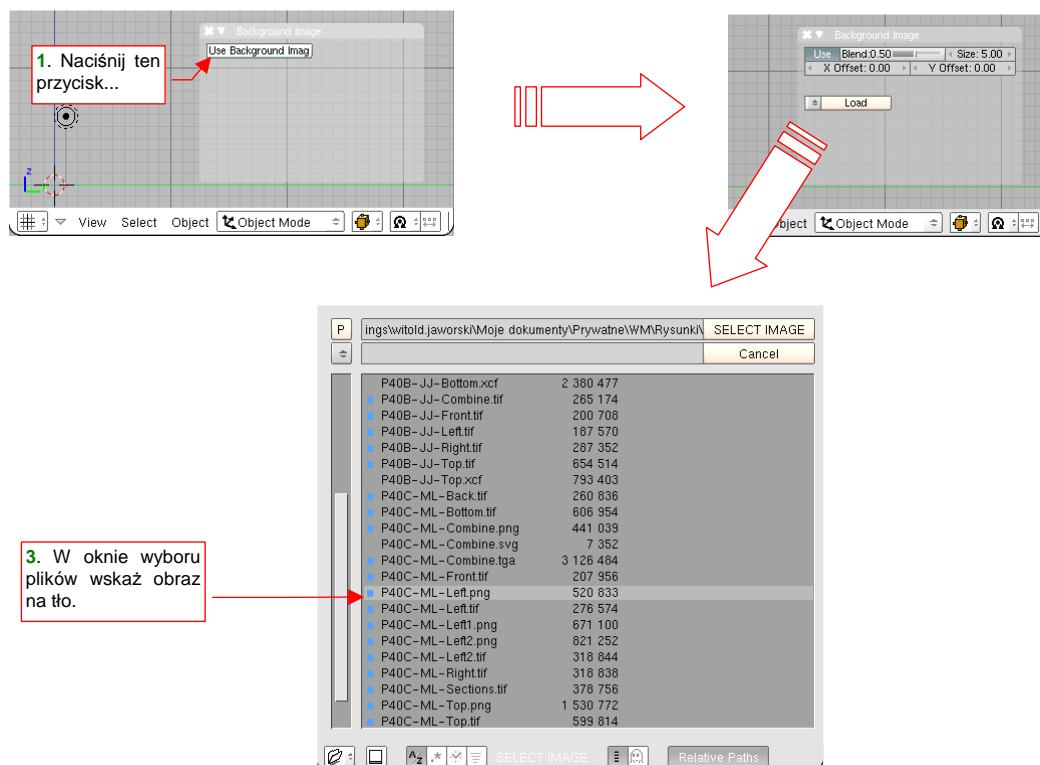


Rysunek 13.7.3 Ustalanie położenia kursora w oknie **View Properties**

W sekcji **3D Cursor** okna dialogowego **View Properties** możesz bezpośrednio wpisać liczbowe wartości współrzędnych **X**, **Y**, **Z** kursora.

13.8 Ustalenie tła widoku (planów samolotu)

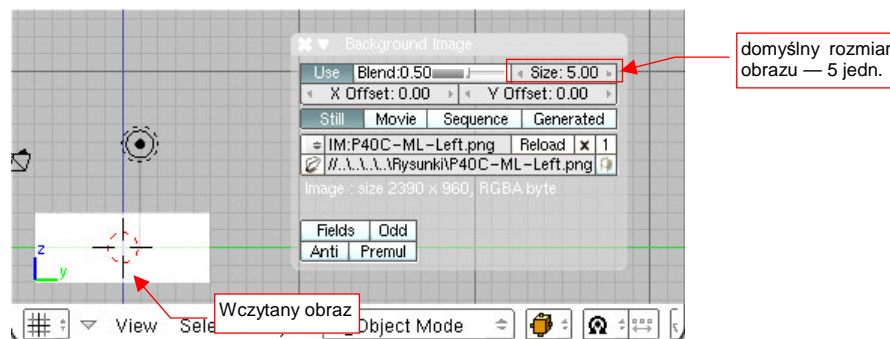
Aby podstawić plany samolotu jako tło, wywołaj polecenie **View→Background Image**. W oknie widoku pojawi się wówczas półprzezroczyste okno **Background Image** (Rysunek 13.8.1). Jak wszystkie okna, rysowane w Blenderze w ten sposób, jest to okno niemodalne (tzn. możesz dalej, nie zamykając go, edytować obiekty lub zmieniać projekcję).



Rysunek 13.8.1 Otwieranie obrazu — tła okna 3D View

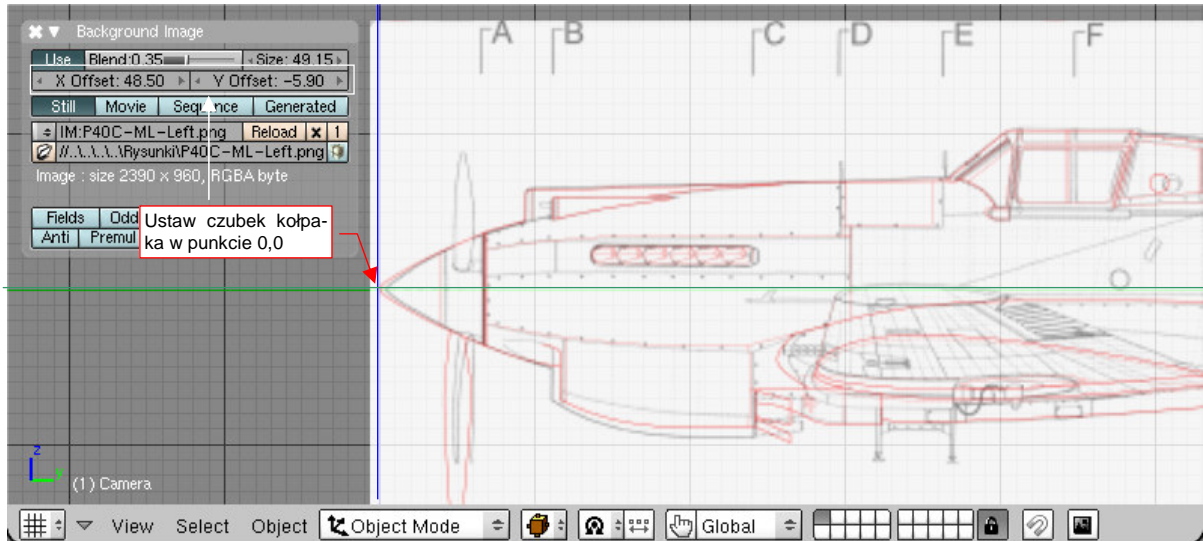
W oknie **Background Image** kliknij w przycisk **Use Background Image**. Następnie naciśnij przycisk **Load**, i w oknie wyboru plików wskaż plik, który ma być użyty jako tło widoku (Rysunek 13.8.1).

Obraz po załadowaniu wygląda dość niepozornie — jest bardzo mały (Rysunek 13.8.2). Blender umieścił jego środek w środku układu współrzędnych.



Rysunek 13.8.2 Plan samolotu zaraz po wczytaniu

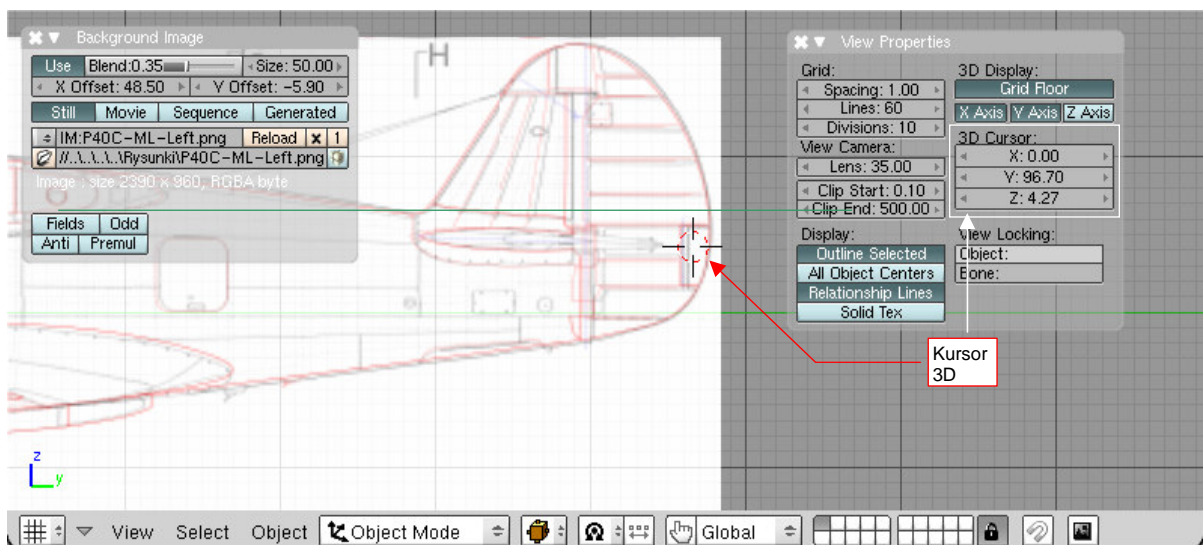
Niewielki rozmiar wczytanego rysunku wynika z domyślnego rozmiaru (**Size**), jaki jest wpisany w oknie **Background Image**: 5.0 (Rysunek 13.8.2). Wartość, wpisywana w to pole, określa rozmiar połowy dłuższego boku obrazu. Zakładając, że 1 jednostka Blendera to 10 cm rzeczywistego samolotu, powinniśmy zwiększyć **Size** do **50.0**. (Długość kadłuba P-40C wynosiła 967cm, a nasz cały obraz będzie się w tej skali rozciągał na niewiele więcej — 100 jednostek, czyli 1000cm). Jednocześnie przesuń obraz (pola **X Offset**, **Y Offset**) tak, by oś kadłuba pokryła się z osią **Y**, a czubek kołpaka śmigła dotykał osi **Z** (Rysunek 13.8.3).



Rysunek 13.8.3 Obraz tła — wyrównanie do dokładnego ustalenia rozmiaru

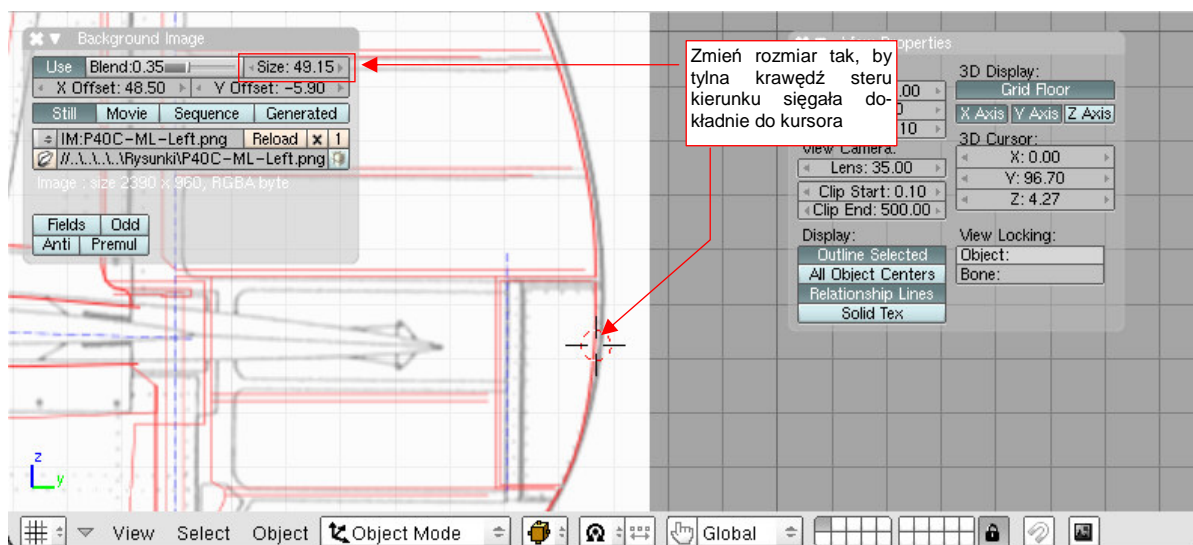
Możesz jednocześnie — na czas tych "przymiarek" — zmniejszyć przejrzystość tła. Służy do tego suwak **Blend** w oknie **Background Image**.

Aby wyznaczyć punkt, do którego ma sięgać koniec kadłuba samolotu, wywołaj polecenie **View→View Properties**. Spowoduje to otwarcie okna dialogowego **View Properties**. Podobnie jak **Background Image**, okno to nie jest modalne — możesz je pozostawić na ekranie i dalej zmieniać rysunek. W sekcji **3D Cursor** okna **View Properties**, ustaw współrzędne: **X = 0.0**, **Y = 96.7** (odpowiada 967 cm rzeczywistej długości kadłuba), **Z = 0.0**. Spowoduje to pojawienie się kursora 3D w odpowiednim miejscu na osi Y. (Potem możesz zmienić współrzędną **Z** na nieco większą, by kursor znalazł się dokładnie tam, gdzie leży koniec kadłuba — trochę ponad osią Y).



Rysunek 13.8.4 Przymiarki do ustalenia końca kadłuba

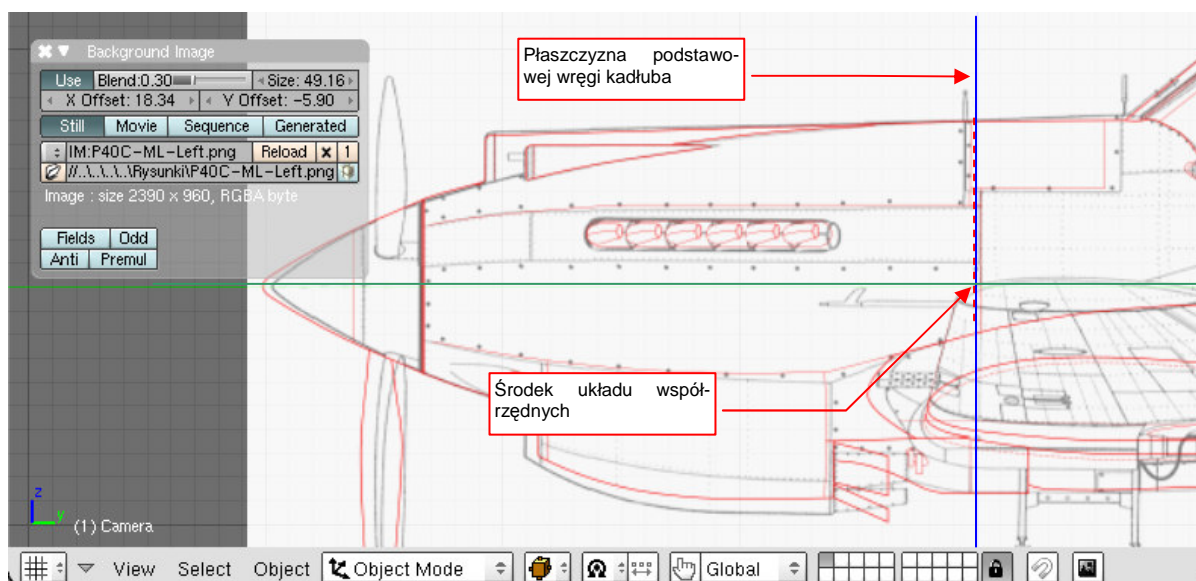
Teraz zmień wartość w **Background Image:Size** na trochę mniejszą — aby krawędź steru kierunku sięgnęła dokładnie punktu, wyznaczonego przez kursor 3D (Rysunek 13.8.5).



Rysunek 13.8.5 Dopasowywanie rysunku tła do zadanych wymiarów

Pamiętaj, że zmiana rozmiaru zachodzi względem środka obrazu — w obydwu kierunkach! Po każdej zmianie skali tła sprawdź obydwa krańce obrazu. Czasami będziesz musiał go odrobinę przesunąć (zmianą wartości **X Offset**), aby skorygować powstałą różnicę. Po kilku takich przybliżeniach powinieneś osiągnąć założone położenie rzutu z lewej. (Czubek kołpaka śmigła ma znajdować się w środku układu współrzędnych, krawędź steru dotykać kursor 3D).

Po wyskalowaniu obrazu pozostaje przesunąć go w docelowe położenie. W praktyce końcówka nosa samolotu nie jest najlepszą powierzchnią odniesienia. Proponuję ustawiać obrazy tak, by przez środek układu współrzędnych przechodziła jakaś inna, "ważniejsza" dla konstrukcji powierzchnia. Użyjmy do tego celu podstawowej wręgi kadłuba, do której były zamocowane: silnik i przedni dźwigar płata¹. Przesuń obraz (zmieniając wartość **X Offset**) tak, by środek układu współrzędnych znalazł się w tym miejscu (Rysunek 13.8.6):

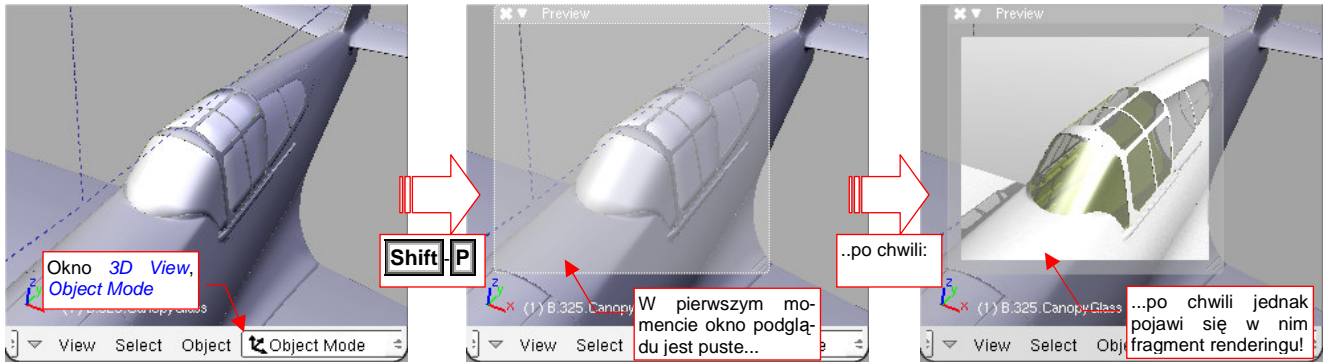


Rysunek 13.8.6 Ostateczne położenie tła — podstawa kołpaka śmigła będzie powierzchnią bazową

¹ Jeżeli nie planowałbym wykonania innych wersji P-40, wybrałbym jako bazę płaszczyznę podstawy kołpaka śmigła. Płaszczyzna ta uległa jednak przesunięciu od P-40D, i taki wybór utrudniłby użycie elementów z P-40B do stworzenia modelu tej nowej wersji.

13.9 Szybki podgląd fragmentu renderingu (*Render Preview*)

Gdy pracujemy nad szczegółem jakiejś tekstury czy materiału, właściwie co chwila coś zmieniamy, a następnie sprawdzamy jak wygląda efekt na finalnym renderingu. Korzystanie z **F12** (*Render Current Frame* — por. str. 88) byłoby w tym przypadku uciążliwe: trzeba by było co chwila ustawiać kamerę i jej cel w nowe miejsca! Aby tego uniknąć, twórcy Blendera dodali specjalne polecenie: **View → Render Preview...** (dostępne pod skróttem **Shift-P**, w *Object Mode*). Jego działanie ilustruje Rysunek 13.9.1:

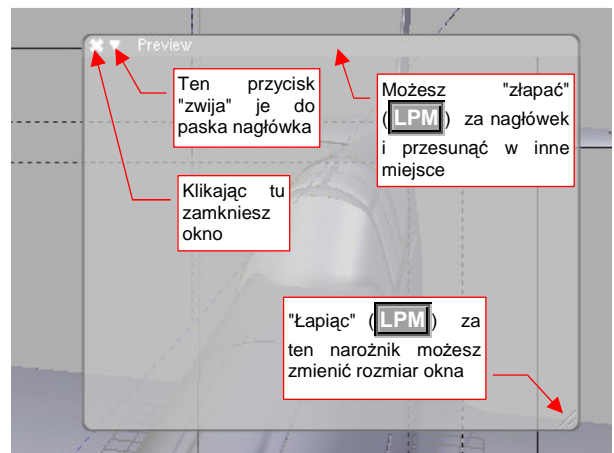


Rysunek 13.9.1 Wywołanie podglądu renderingu

Po naciśnięciu **Shift-P** pojawia się półprzezroczyste okno. Jeżeli przez chwilę nie będziesz niczego robił — zacznie się wypełniać renderowanym obrazem. Jeżeli nie czekając na zakończenie zmienisz cokolwiek w rysunku (Blender nadal śledzi Twoją myszkę!) — rendering zostanie przerwany. Potem, przy okazji kolejnego "bezzruchu", obraz zacznie być wyznaczany na nowo.

Okno podglądu, choć zajmuje miejsce na ekranie, nie blokuje dostępu ani do modelu, ani do przycisków.

Rysunek 13.9.2 pokazuje zastosowanie poszczególnych elementów ramki okna podglądu.



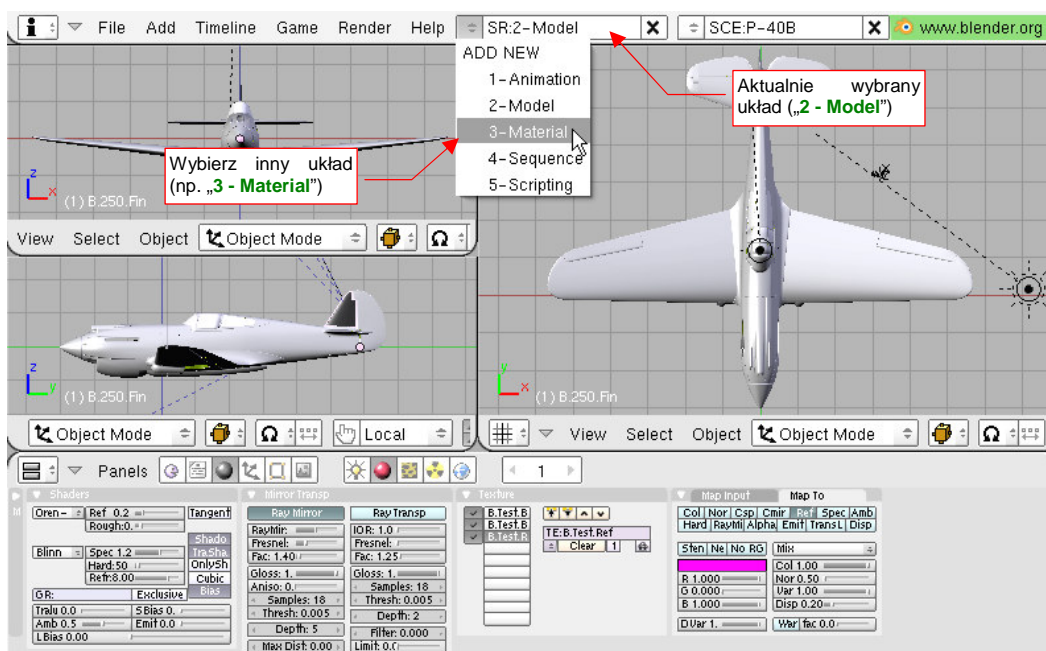
Rysunek 13.9.2 Funkcja ramki okna podglądu

- Podgląd w oknie *Preview* należy traktować wyłącznie jako orientacyjną pomoc. Pomija się w nim pewne czasochłonne operacje, które są wykonywane podczas generowanego ostatecznego obrazu. Na przykład — wygładzanie konturów (*antialiasing*). Także inne, bardziej złożone efekty mogą "wyjść" na ostatecznym renderingu nieco inaczej, niż na podglądzie. Warto więc co jakiś czas jednak wygenerować ostateczny obraz, aby upewnić się, czy wszystko wychodzi na nim tak, jak chcemy.

- Preview* nie reaguje na zmianę widoczności warstwy. Potrafi pokazywać na podglądzie elementy, które w oknie *3D View* już zostały ukryte. Odświeżenie można zazwyczaj wymusić niewielkim obrotem widoku w jakąkolwiek stronę. Jeżeli masz wątpliwości, czy widzisz prawdziwy rezultat tego, co przed chwilą zmieniłeś — zamknij okno *Preview* i otwórz je ponownie.

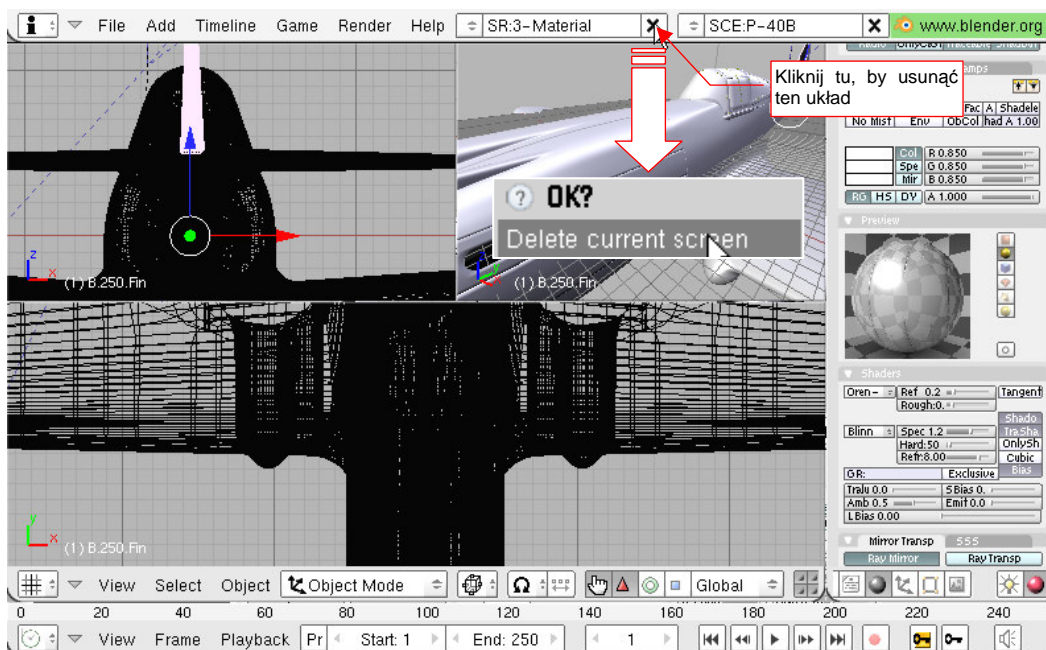
13.10 Układ ekranu (**Screen**) — zarządzanie

Blender domyślnie proponuje kilka różnych układów ekranu, przygotowanych do różnych faz pracy nad projektem. Możesz się pomiędzy nimi przełączać za pomocą listy **SR** (**Sc**Reen — Rysunek 13.10.1):



Rysunek 13.10.1 Przełączenie na inny układ ekranu (aktualny układ — „2 - Model”)

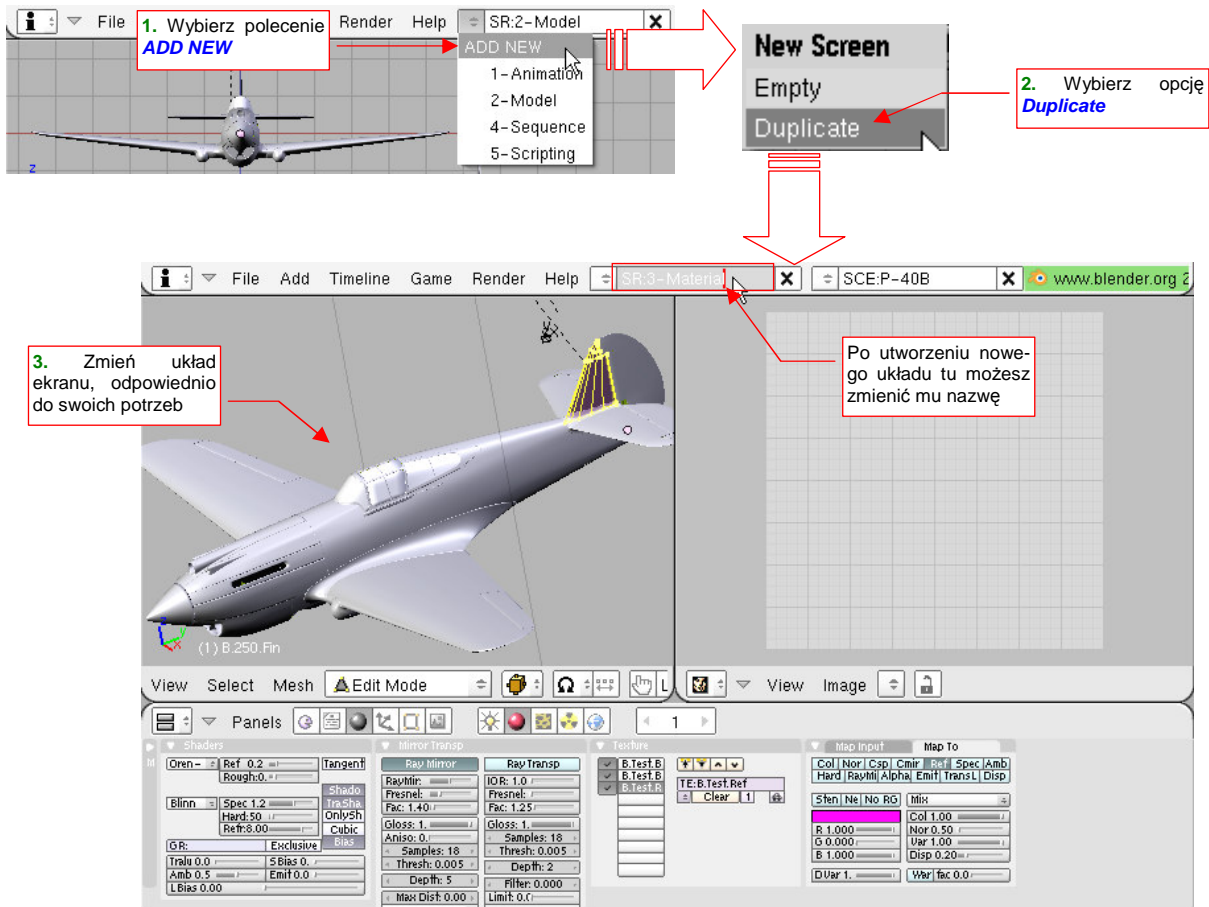
Po wybraniu z listy **SR** np. układu do pracy nad materiałami (**3 - Material**), zobaczysz zupełnie inne okna (Rysunek 13.10.2):



Rysunek 13.10.2 Przełączenie na inny układ ekranu (nowy układ — „3 - Material”)

Jeżeli nie jesteś zadowolony z domyślnych ustawień Blendera — możesz tu wszystko pozmienić. Możesz także usunąć ten układ — klikając w przycisk **X**, umieszczony po prawej stronie listy **SR**. Blender upewni się tylko, czy wiesz co robisz, wyświetlając pytanie „Delete current screen?” (Rysunek 13.10.2). Gdy potwierdzisz, aktualny układ ekranu zostanie usunięty, a Ty znajdziesz się w poprzednim układzie (w tym przykładzie będzie to „2 - Model”).

Czasami szybszą drogą do osiągnięcia układu, o który nam chodzi jest stworzenie nowego, na podstawie jednego z istniejących. Aby stworzyć nowy układ ekranu, wybierz z listy **SR** polecenie **ADD NEW** (Rysunek 13.10.3):



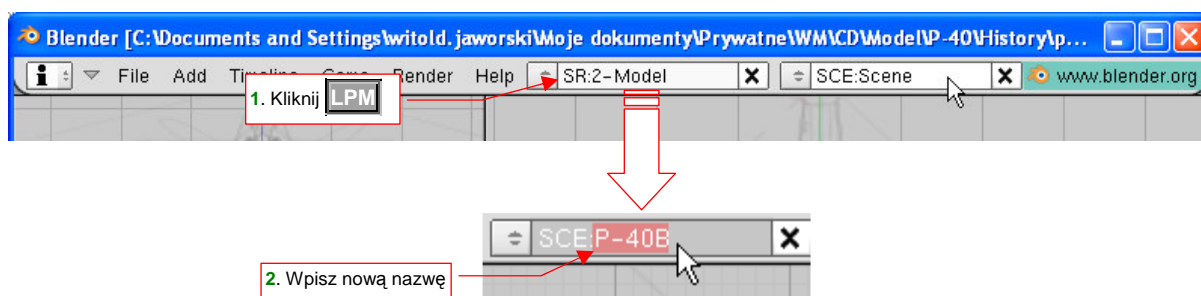
Rysunek 13.10.3 Utworzenie nowego układu ekranu

W oknie **New Screen**, które się pojawi, wybierz opcję **Duplicate**. W odpowiedzi Blender stworzy nowy układ ekranu, i nada mu taką samą nazwę, jaką miał poprzedni, z przyrostkiem „001” (np. „2 - Model.001”). Dostosuj ten ekran stosowanie do swoich potrzeb. Rysunek 13.10.3 pokazuje układ, w którym najwygodniej mi się pracuje z materiałami: pojedyncze okno **3D View** z lewej, okno **UV/Image Editor** z prawej. Możesz temu układowi nadać nawet nazwę „3 - Material” — taką, jaką miał układ ekranu, który przed chwilą usunęliśmy.

13.11 Sceny — zarządzanie

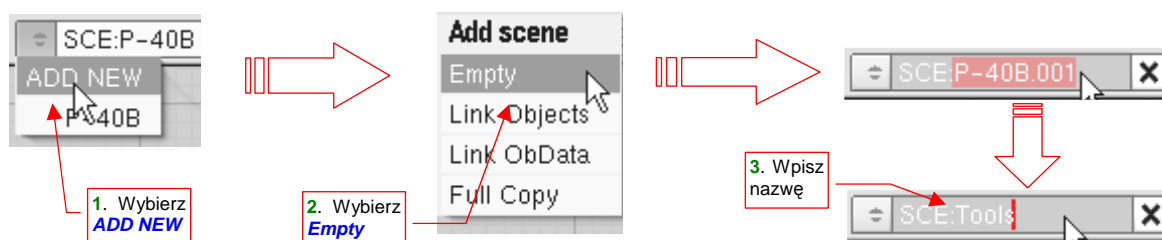
Plik Blendera może zawierać wiele scen. Każda z nich to odrębna "przestrzeń", w której można umieszczać obiekty (więcej na ten temat — str. 569). Domyślnie w nowym rysunku istnieje tylko jedna, o nazwie **Scene**. Do zarządzania scenami służy zespół kontrolki umieszczony w nagłówku okna *User Preferences* (Rysunek 13.11.1). Składa się z pola, pokazującego nazwę aktualnej sceny, przycisku rozwijającego listę scen, oraz przycisku usuwającego aktualną scenę.

Aby zmienić nazwę aktualnej sceny, kliknij **LPM** w jej nazwę, i wpisz nową (Rysunek 13.11.1):



Rysunek 13.11.1 Zmiana nazwy aktualnej sceny

Aby stworzyć nową scenę, wybierz z listy scen polecenie **ADD NEW** (Rysunek 13.11.2):



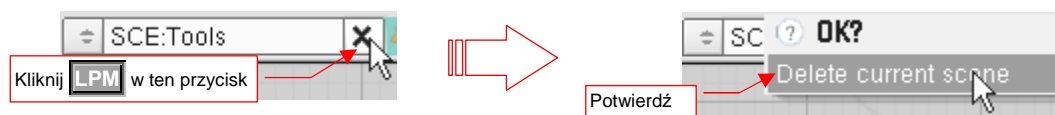
Rysunek 13.11.2 Dodanie nowej sceny

Spowoduje to pojawienie się menu *Add scene*, z którego możesz wybrać jedną z opcji:

- **Empty**: tworzy nową, pustą scenę;
- **Link Objects**: wypełnia nową scenę referencjami do obiektów z aktualnej sceny. (Ten sam obiekty będzie "użyty" przez obydwie sceny. Więcej na ten temat — str. 569);
- **Link ObData**: tworzy w nowej scenie kopie obiektów z aktualnej sceny, które współdzielą z obiektami aktualnej siatki, materiały, itp. (To tak, jak gdybyś wszystkie powielił poleceniem **Alt-D** — *Duplicate Linked* — str. 796);
- **Full Copy**: cała zawartość nowej sceny będzie niezależną kopią aktualnej;

W praktyce stosuję zazwyczaj opcję **Empty** (pojedyncze obiekty można później dodać do innej sceny poleceniem *Object→Make Links→To Scene*). Nowa scena nosi taką nazwę, jak scena oryginalna, z dodanym na końcu ".001". Zmień ją na właściwą.

Aby usunąć aktualną scenę, naciśnij przycisk z prawej strony nazwy (Rysunek 13.11.3):



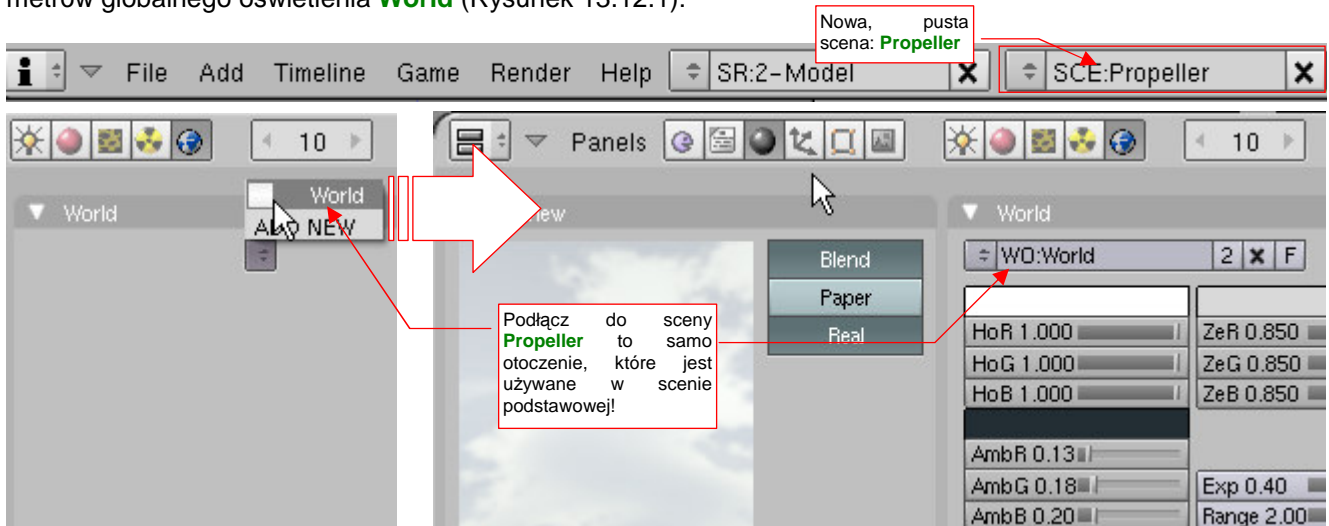
Rysunek 13.11.3 Usunięcie aktualnej sceny

Potwierdź **Delete current scene**, i cała scena zniknie z pliku Blendera.

13.12 Przykład wykorzystania drugiej sceny

W tej sekcji pokażę, jak zastosować drugą scenę do uzyskania szybkiego (pod względem czasu renderowania) efektu rozmycia ruchu śmigła¹. Chodzi tu o znaczną redukcję czasu, jaki zabiera renderowanie pojedynczej sceny z włączoną opcją *MBLUR (Motion Blur)*. To szczegółowy opis rozwiązania, które wprowadziłem w sekcji o kompozycji sceny w locie (por. sekcja 8.1, str. 514). Dlatego w tym przykładzie użyjemy tego samego pliku, który jest tam wykorzystywany: *p40history/P40-8.01a.blend*.

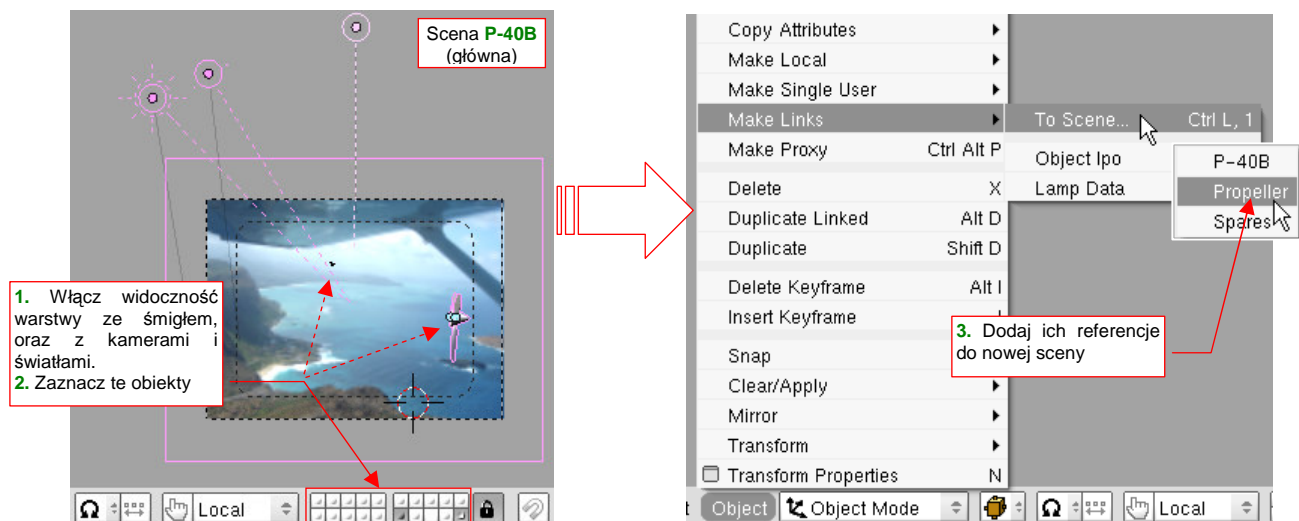
Zacznijmy od dodania do pliku nowej, pustej sceny (por. str. 778). Nadaj jej nazwę **Propeller**, bo będziemy na niej renderować samo śmigło. Od razu podłącz do niej wykorzystywany w scenie głównej (**P-40B**) zestaw parametrów globalnego oświetlenia **World** (Rysunek 13.12.1):



Rysunek 13.12.1 Podłączenie do sceny **Propeller** otoczenia **World**

Zawartość obydwu scen: **P-40B** (głównej) i **Propeller** będzie składana w jeden obraz. Wykorzystanie tych samych ustawień globalnego oświetlenia **World** znacznie to ułatwi.

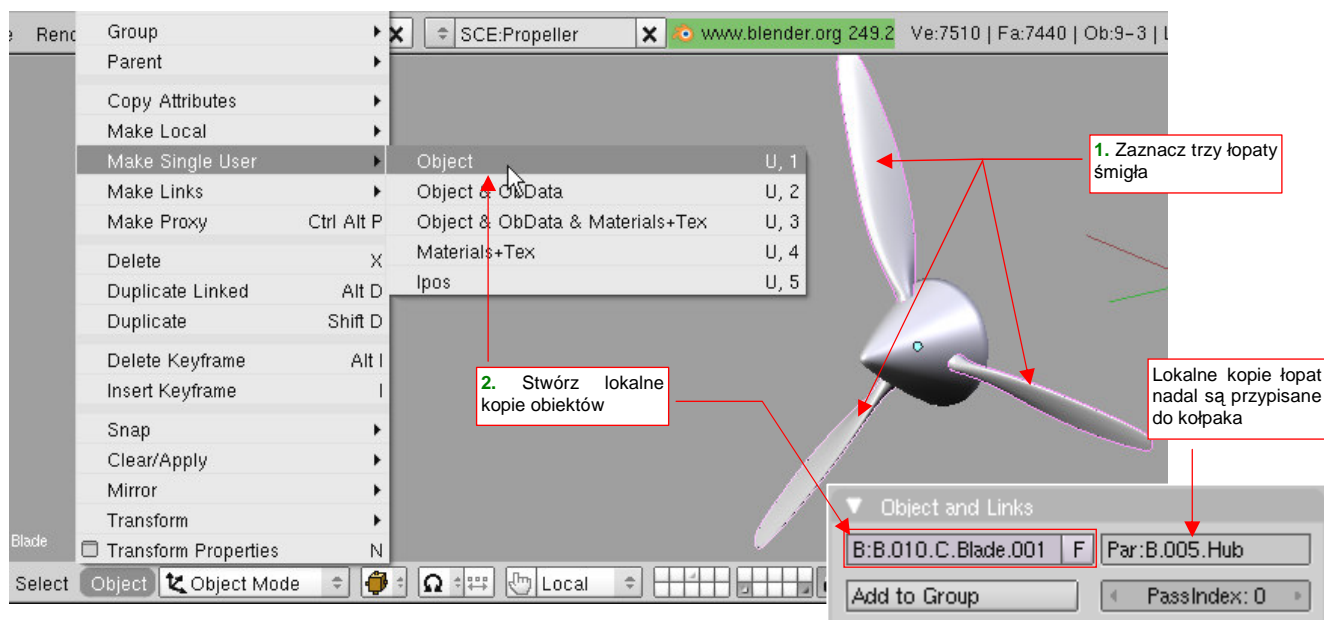
Zaznacz w scenie głównej zespół śmigła, oraz kamery, oświetlenia, i ich „cel” (obiekt **Target**). Wstaw odnośniki (**Object → Make Links → To Scene**) tych obiektów do sceny **Propeller** (Rysunek 13.12.2):



Rysunek 13.12.2 Podłączenie do sceny **Propeller** kamery, światła, i śmigła.

¹ Zagadnienie, które opisuje ta sekcja, jest „przekrojowe”. Łączy się z nim zastosowanie edytora obiektów, animacji (krzywe IPO), edytora węzłów (*Node Editor*), ... Nie mogłem zdecydowanie zakwalifikować tego fragmentu do żadnego konkretnego rozdziału ze szczegółowym opisem poleceń Blendera. Umieściłem więc go tutaj, w rozdziale „ogólnym”. W razie czego pamiętaj, że to jedno z ostatnich objaśnień, dodanych do tej książki. Łączy wiele zagadnień, które wyjaśniałem wcześniej.

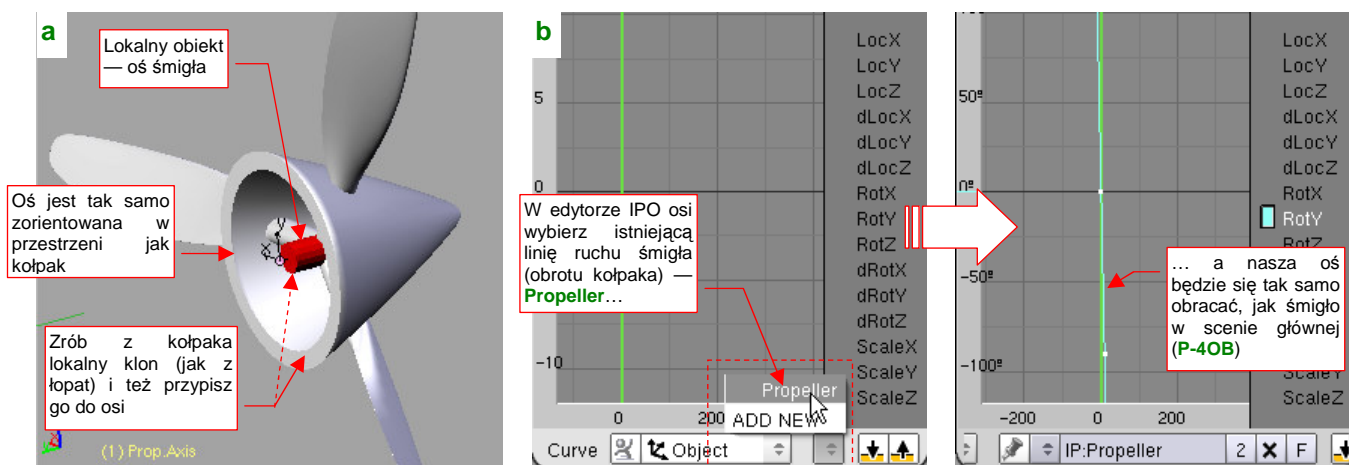
Przełącz się teraz na scenę **Propeller**. Zmniejsz do zera wpływ (*Influence*) położenia uchwytu sterowania na kąt obrotu kołpaka. Teoretycznie można byłoby teraz przejść do ustawiania parametrów renderowania tej sceny. Sprawdź jednak najpierw, czy przygotowana jeszcze na scenie głównej animacja działa poprawnie. Naciśnij **Alt-A**, aby „puścić śmigło w ruch”. I co? Okazuje się, że łopaty nie obracają się jak powinny, za swoim „rodzicem” (*parent*) — kołpakiem! Wygląda mi to na błąd Blendera. No cóż, aby go obejść, musimy zamienić wszystkie łopaty śmigła na lokalne obiekty tej sceny. Zaznacz je, a następnie wywołaj polecenie **Object → Make Single User** (Rysunek 13.12.3):



Rysunek 13.12.3 Przekształcenie łopat śmigła w obiekty lokalne

Zwróć uwagę, że wybierając opcję **Make Single User → Object** tworzymy lokalny „klon” łopaty. Ten obiekt nadal używa tej samej siatki, co wszystkie pozostałe łopaty na obydwu scenach. Dzięki temu nadal łatwo jest w razie potrzeby skorygować jej kształt, czy też umieszczony na powierzchni napis.

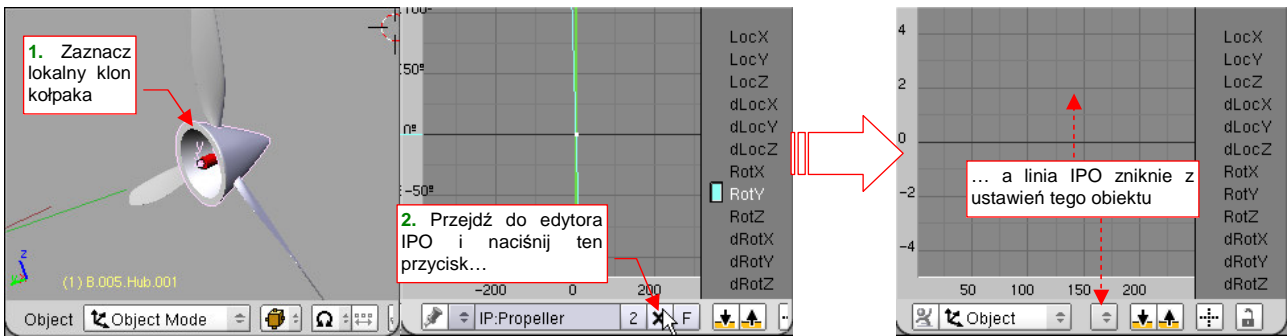
Nawet lokalne klony łopat, choć są przyłączone do kołpaka relacją *Parent*, nie obracają się wraz z nim w kolejnych klatkach animacji. Rozwiązaniem okazało się dopiero wstawienie do sceny **Propeller** nowego obiektu: osi śmigła (**Prop Axis** — Rysunek 13.12.4a). Przypisałem do niej także lokalny klon kołpaka:



Rysunek 13.12.4 Nowa oś śmigła (**Prop.Axis**) i przypisanie jej krzywej animacji (linii IPO)

Ustawiłem ten obiekt tak, by był zorientowany w przestrzeni identycznie jak kołpak. Pozwala to przypisać mu krzywą IPO kołpaka (o nazwie **Propeller** — Rysunek 13.12.4b). Ta linia była już raz użyta w scenie **P-40B**, do obrotu śmigła. Podobnie jak w przypadku klonów siatki, takie współdzielenie bloku danych pomiędzy scenami ułatwia zachowanie ich spójności.

Z kolei tę samą linię IPO należy „odpisać” (usunąć) z lokalnego klonu śmigła, by nie obracał się dwa razy szybciej od reszty zespołu (Rysunek 13.12.5):



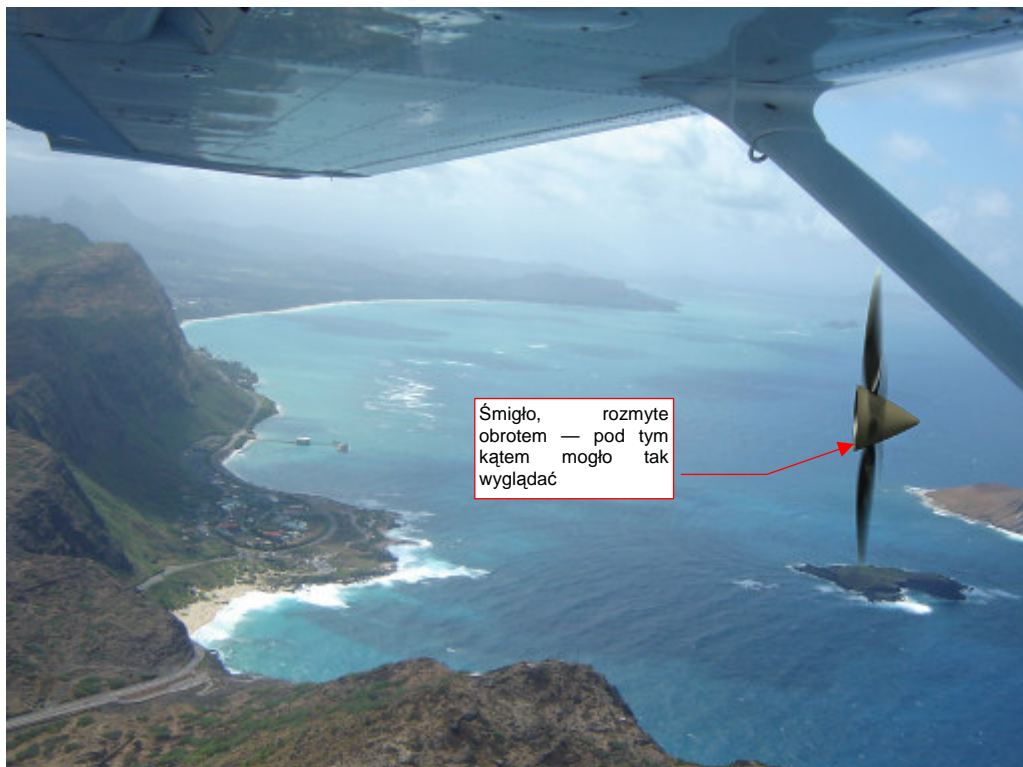
Rysunek 13.12.5 Usunięcie krzywej animacji z kołpaka śmigła

Teraz można już przestawić parametry renderowania sceny **Propeller** (Rysunek 13.12.6):



Rysunek 13.12.6 Ustawienia renderowania pomocniczej sceny **Propeller**

Włącz maksymalne rozmycie ruchu **MBLUR**, wyłącz **Do Composite**. Gdy to zrobisz, możesz wykonać testowy render (**F12**). Przekonamy się, czy śmigło jest dostatecznie rozmyte (Rysunek 13.12.7):

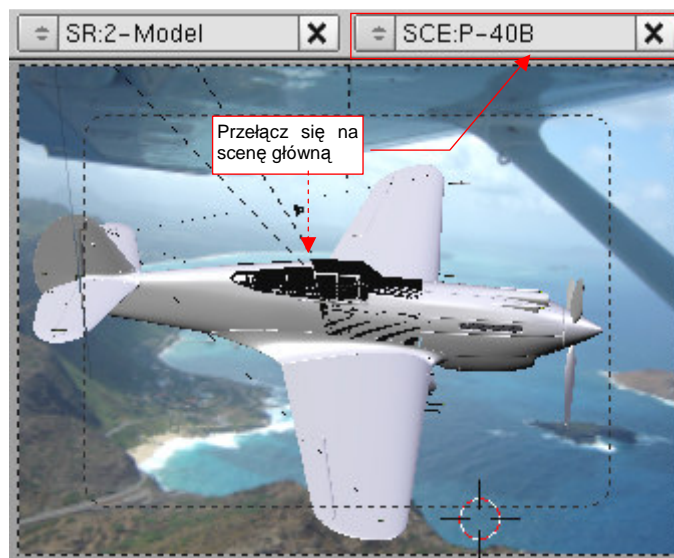


Rysunek 13.12.7 Testowy render sceny **Propeller**

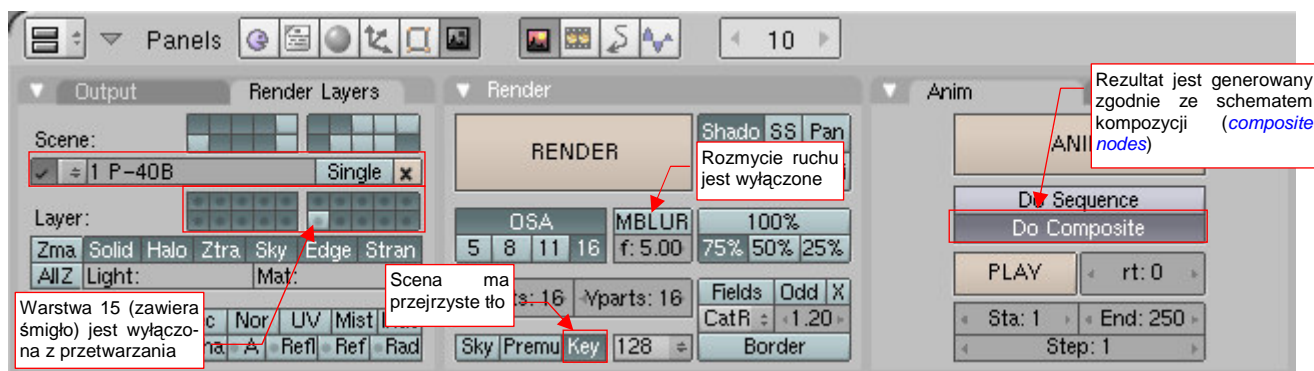
Pomocnicza scena ze śmigłem jest już gotowa. Teraz przełącz się na scenę główną (**P-40B** — Rysunek 13.12.8).

- Uważaj, aby na obydwu scenach była zawsze aktywna ta sama kamera.

Rysunek 13.12.9 przedstawia ustawienia renderowania tej sceny. Zwróć uwagę, że zawartość warstwy 16 jest wyłączona z przetwarzania. To właśnie na niej są umieszczone kołpak i łopaty śmigła — czyli to, co się renderuje na scenie **Propeller** (por. Rysunek 13.12.7). **P-40B** ma przezroczyste tło (opcja **Render:Key**) i wyłączony **MBLUR** (*Motion Blur*). Dla porządku nazwałem jego domyślną warstwę renderowania tak samo, jak scena — **P-40B** (Rysunek 13.12.9):

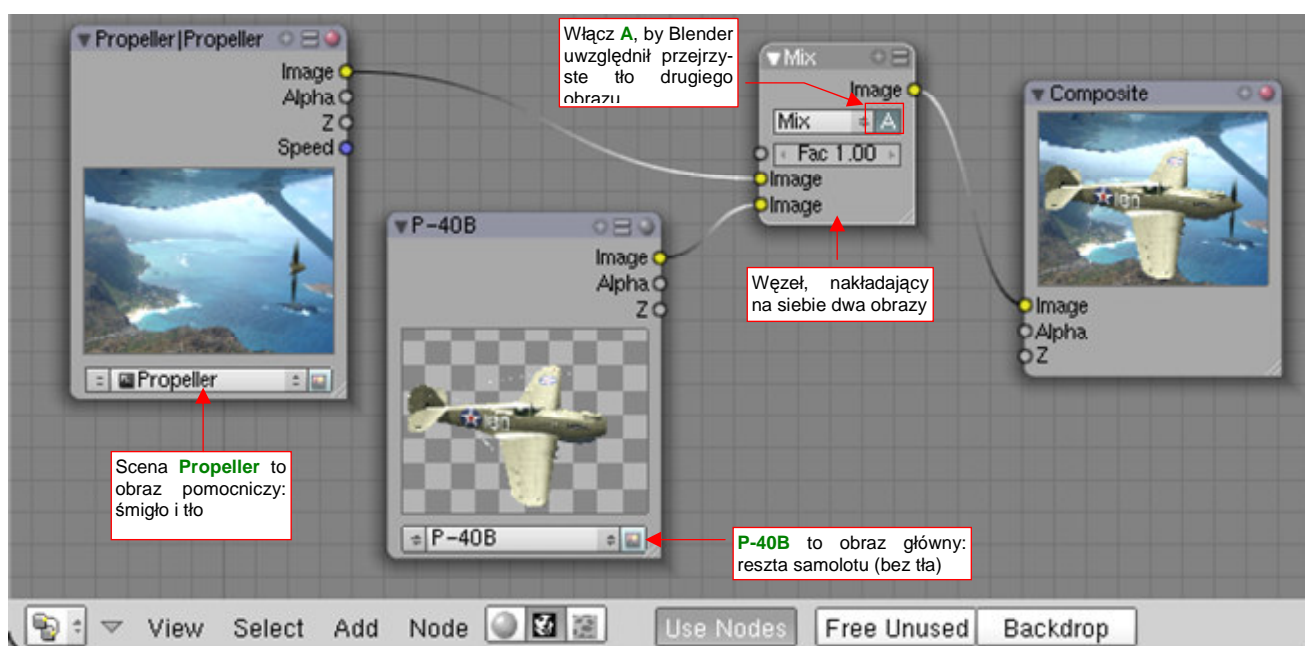


Rysunek 13.12.8 Przełączmy się teraz na scenę podstawową (**P-40B**)



Rysunek 13.12.9 Ustawienia renderowania podstawowej sceny (**P-40B**)

Zwróć uwagę, że w zakładce **Anim** jest włączony przycisk **Do Composite**. To oznacza, że wynik zostanie stworzony zgodnie ze schematem kompozycji, przygotowanym w **Node Editor** (Rysunek 13.12.10):



Rysunek 13.12.10 Schemat połączenia obrazów z dwóch scen w ostateczny render

Obrazy z obydwu scen składam za pomocą węzła **Mix** w ostateczny rezultat: samolot z rozmytym śmigłem.

Zobaczmy, jak prezentuje się rezultat na próbnym renderze (Rysunek 13.12.11):



Rysunek 13.12.11 Połączenie obrazów z dwóch scen w ostateczny render

Samolot wygląda tak samo, jak wynik zwykłego włączenia opcji *MBLUR* w scenie głównej. Różnica to czas generowania tego obrazu. Maszyna na rysunku powyżej tworzył się niecałe dwie minuty (dokładniej: 1 min 46s). Na render tego samego modelu, ale generowanego bez użycia węzłów, zużyłem nieco ponad 14 min. Oznacza to, że dzięki podziałowi na dwie sceny renderujemy 16 razy (tak jest ustawiony tutaj *MBLUR*) tylko to, co jest potrzebne — śmigło. Unikamy bezsensownego, wielokrotnego przetwarzania tego, co się nie porusza. W efekcie uzyskaliśmy dziesięciokrotne skrócenie czasu generowania ostatecznego obrazu!

Rozdział 14. Blender — edytor obiektów (Object Mode)

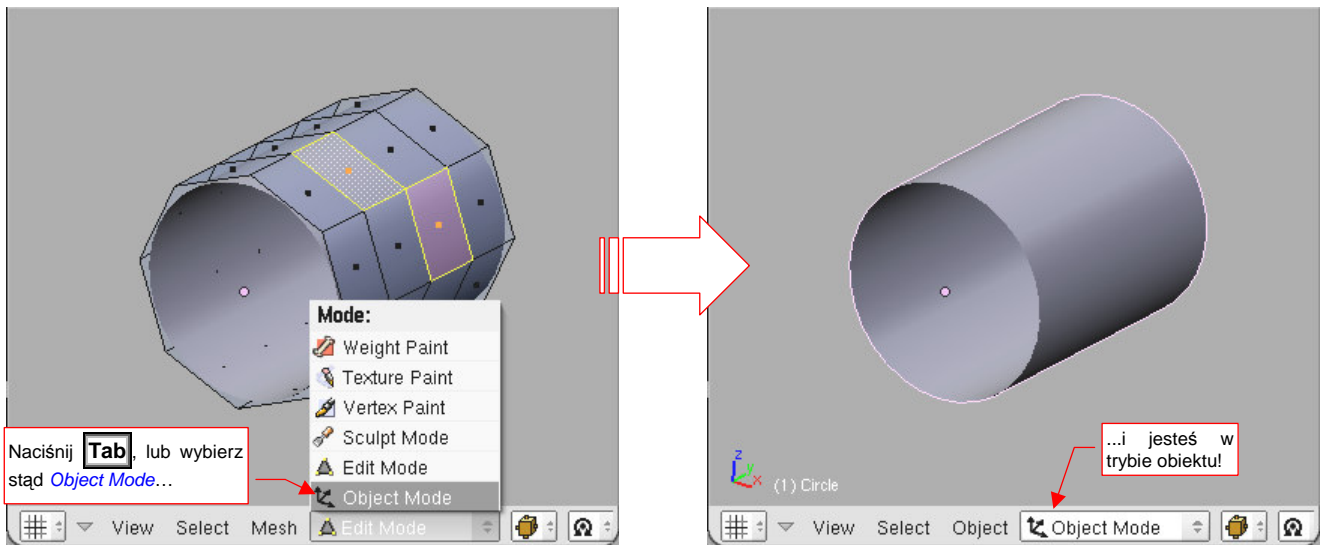
Z edytorem obiektów miałeś już okazję się zetknąć (Rozdział 3, "Blender – pierwsze kroki"). W tym rozdziale omówimy podstawowe pojęcia, które się z nim wiążą.

Pominę tu zagadnienia, które przedstawiłem wcześniej (Rozdział 3). Chodzi o:

- metody selekcji. Opisałem je już w sekcji 3.3, (str. 70)
- podstawowe transformacje (przesunięcie, obrót obiektu). Opisałem je już w sekcji 3.4 (str. 74). Opiszę jednak zmianę skali, bo tam tylko o niej wspomniałem.

14.1 Włączenie trybu obiektów (*Object Mode*)

Tryb edycji siatki włączasz naciskając przycisk **Tab**. Alternatywnie możesz także zrobić to za pomocą menu z nagłówka okna widoku (Rysunek 14.1.1):



Rysunek 14.1.1 Okno widoku: przełączenie w trybie obiektu (*Object Mode*)

Powtórne naciśnięcie **Tab** przełączy Cię z powrotem do poprzedniego trybu (np. edycji — *Edit Mode*). Możesz także wybrać inny tryb z nagłówka widoku, listy rozwijalnej *Mode*.

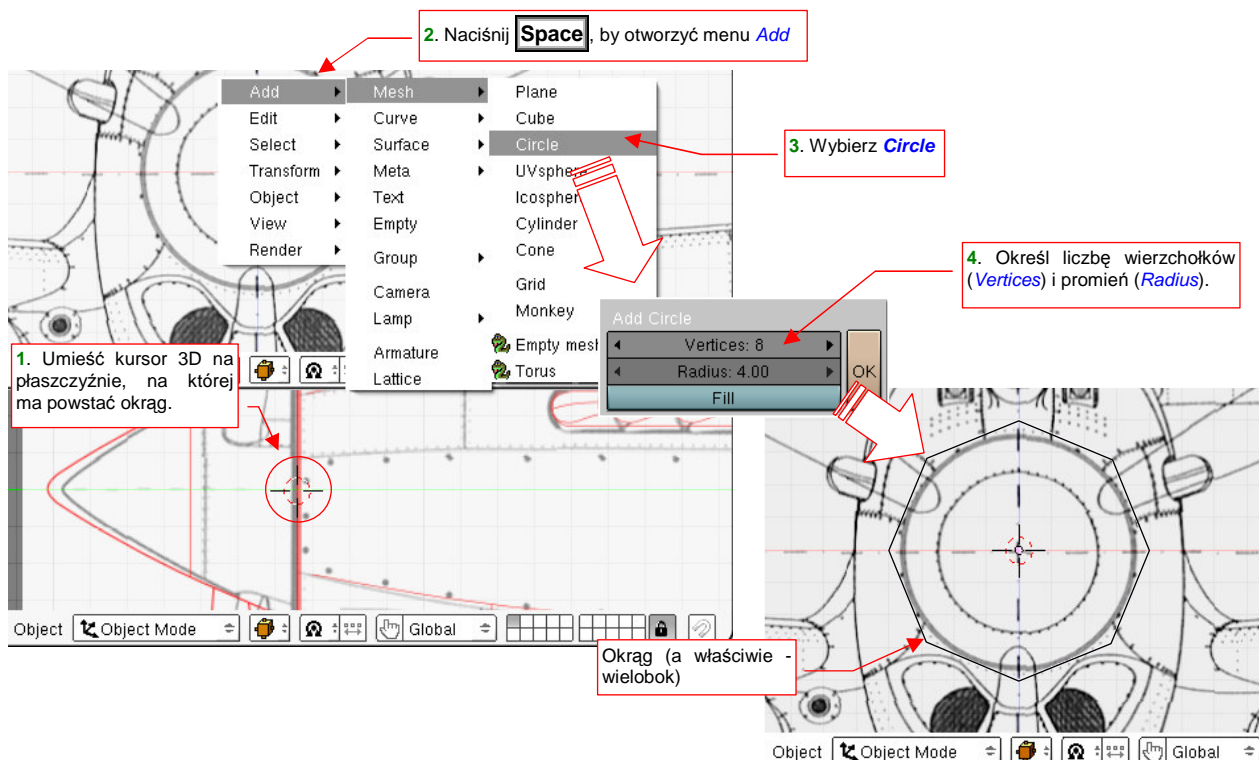
Nagłówek okna *View 3D* ma w trybie obiektu wiele przydatnych kontrolki (Rysunek 15.2.2) :



Rysunek 14.1.2 Okno widoku: kontrolki dostępne w trybie edycji

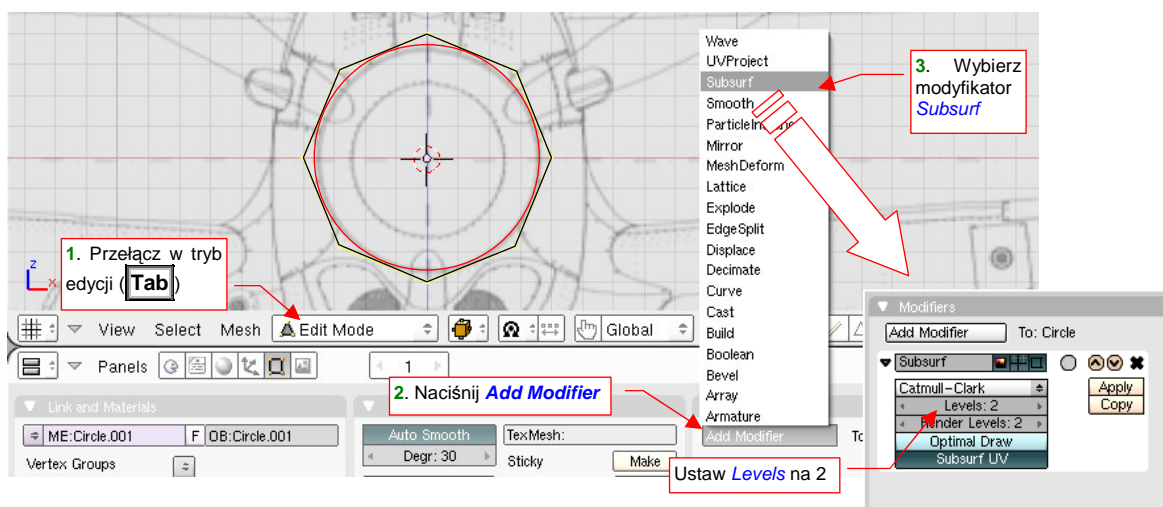
14.2 Rysowanie okręgu (*Circle*)

Najpierw umieść kursor 3D w miejscu, gdzie ma być środek okręgu. (Każdy nowa siatka jest umieszczana tam, gdzie aktualnie jest kursor). Następnie naciśnij **Space** (lub rozwiń menu *Add* — jest zaraz obok menu *File*). Wywołaj polecenie *Add*→*Mesh*→*Circle*. W oknie dialogowym, które się pojawi, określ liczbę wierzchołków i promień okręgu. Nie staraj się wybierać zbyt wielu wierzchołków (pole *Vertices*). Za chwilę zobaczysz, że 8 zupełnie wystarczy. Po naciśnięciu OK, na rysunku ujrzysz rezultat (Rysunek 14.2.1):



Rysunek 14.2.1 Stworzenie okręgu (wieloboku)

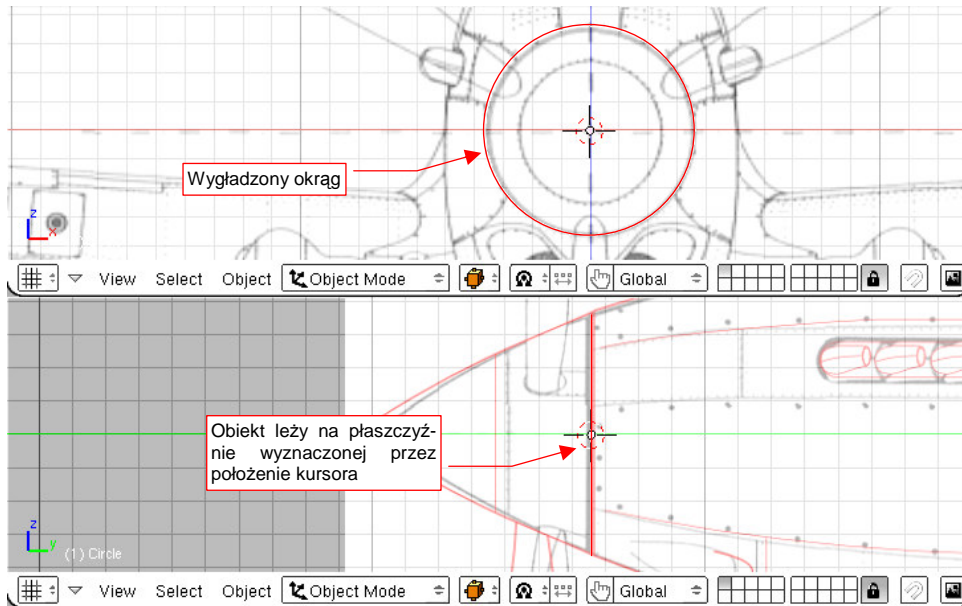
Jest to właściwie wielobok, a nie okrąg. Aby bardziej odpowiadał nazwie polecenia, które go stworzyło, należy jeszcze go wygładzić. Naciśnij **Tab**, aby przełączyć się w tryb edycji. Następnie w panelu *Modifiers* wybierz z listy *Add Modifier* modyfikator *Subsurf* (Rysunek 14.2.2):



Rysunek 14.2.2 "Wygładzenie" okręgu

Dodałeś w ten sposób do siatki modyfikator, który w sposób dynamiczny zmienia jej właściwości. Modyfikator *Subsurf* posiada parametry, które możesz zmienić w panelu *Modifiers* (Rysunek 14.2.2). Ustaw wartości *Levels*

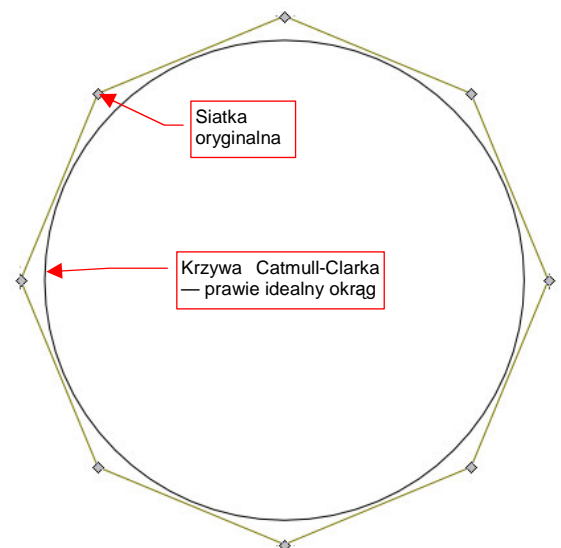
i **Render Levels** na 2. To wystarczy do uzyskania całkiem porządnego przybliżenia okręgu. Naciśnij teraz powtórnie **Tab**, aby przełączyć się z powrotem w tryb obiektu, i zobaczyć rezultat wygładzenia (Rysunek 14.2.3):



Rysunek 14.2.3 Wygładzony okrąg, i jego położenie w innym rzucie

Do wygładzania okręgu i elips najlepiej jest używać schematu obliczeniowego o nazwie **Catmull-Clark**. Nie zmieniaj tego ustawienia w modyfikatorze **Subsurf** (Rysunek 14.2.2). Schemat Catmulla-Clarka całkiem dokładnie odwzorowuje takie krzywizny już dla oryginalnej siatki złożonej z 6-8 wierzchołków, i dla niewielkiego poziomu podziału (**Levels** = 2).

W przypadku, gdy stworzone koło ma być podstawą do utworzenia jakiegoś niewielkiego, mało istotnego elementu (np. rurek, kabli) — możesz zmniejszyć poziom podziału do 1. Zmniejszy to ogólną liczbę ścian modelu. Gdy chodzi o duży obiekt, jak kołpak śmigła — możesz zwiększyć poziom podziału do renderowania do 3 (aby na pewno wyeliminować nawet małe "nierówności" na obwodzie okręgu).

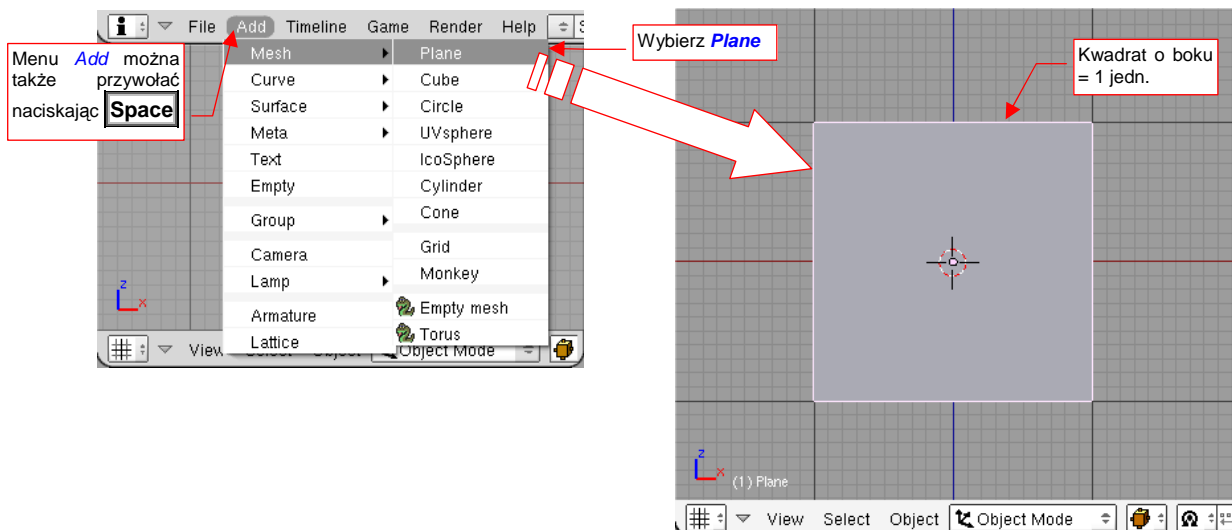


Rysunek 14.2.4 Linia podziałowa, uzyskana z 8 wierzchołków linii bazowej i poziomu podziału (**Level**) = 3

- Jeżeli chcesz dowiedzieć się więcej o krzywych i powierzchniach podziałowych — p. "Powierzchnie podziałowe", str. 571.

14.3 Rysowanie kwadratu (*Plane*)

Umieść kursor 3D w miejscu, gdzie ma być środek kwadratu. (Każdy nowa siatka jest umieszczana tam, gdzie aktualnie jest kursor). Następnie naciśnij **Space** (lub rozwiń menu *Add* — jest zaraz obok menu *File*). Wywołaj polecenie *Add*→*Mesh*→*Plane* (Rysunek 14.3.1):

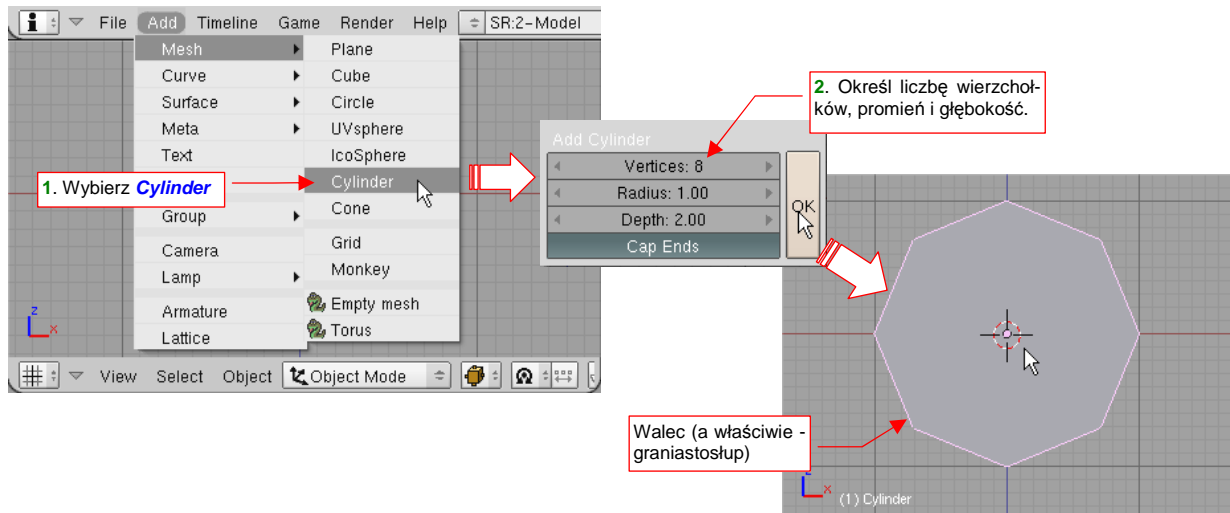


Rysunek 14.3.1 Stworzenie kwadratu

Utworzeniu kwadratu nie towarzyszy żadne okno dialogowe. Nowy obiekt ma zawsze bok = 1 jednostce. Możesz zaraz po utworzeniu nacisnąć **S** (*Scale*) i powiększyć go do odpowiedniego rozmiaru.

14.4 Rysowanie walca (*Cylinder*)

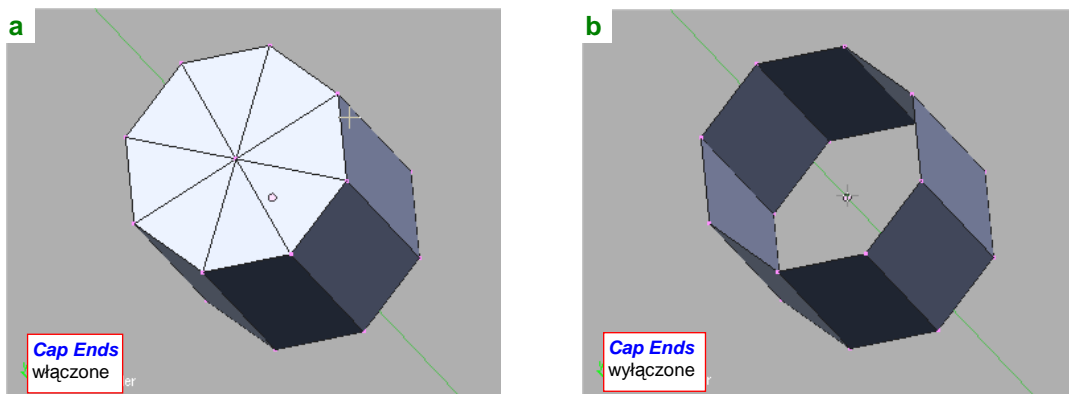
Najpierw umieść kursor 3D w miejscu, gdzie ma być środek walca. Następnie naciśnij **Space** (lub rozwiń menu **Add** — jest zaraz obok menu **File**). Wywołaj polecenie **Add→Mesh→Cylinder**. W oknie dialogowym, które się pojawi, określ liczbę wierzchołków (**Vertices**) i promień (**Radius**), oraz głębokość walca (**Depth**). Po naciśnięciu **OK** na rysunku zobaczysz rezultat (Rysunek 14.4.1):



Rysunek 14.4.1 Stworzenie walca (graniastosłupa)

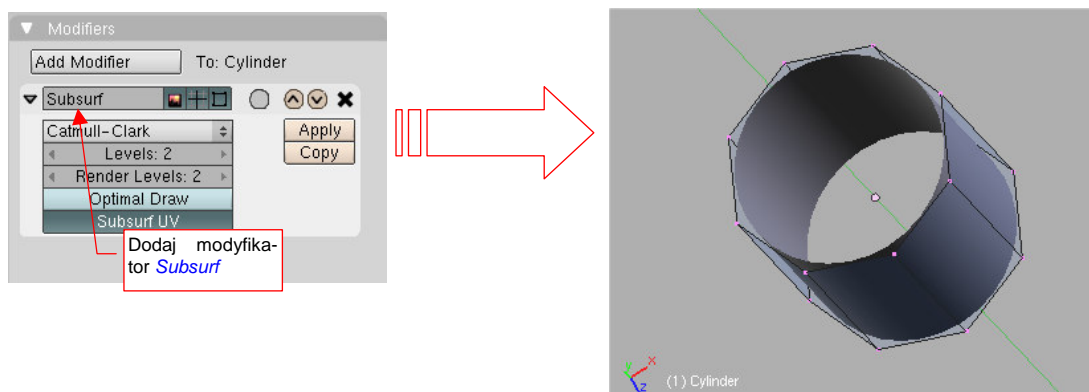
Jest to właściwie graniastosłup, a nie walec. Przełącznik **Cap Ends** z okna **Add Cylinder** pozwala uzyskać:

- włączony: walec zamknięty (z obydwu stron) — Rysunek 14.8.2a;
- wyłączony: otwartą "rurę" — Rysunek 14.8.2a



Rysunek 14.4.2 Działanie przełącznika **Cap Ends**

Podobnie jak okrąg, walec można jeszcze wygładzić. Wystarczy w trybie edycji, w panelu **Modifiers** wybrać z listy **Add Modifier** modyfikator **Subsurf** (Rysunek 14.4.3):



Rysunek 14.4.3 "Wygładzenie" walca

Ustaw parametry **Subsurf:Levels** i **Subsurf:Render Levels** na **2**. To wartość wystarczająca do uzyskania całkiem porządnego przybliżenia okręgu.

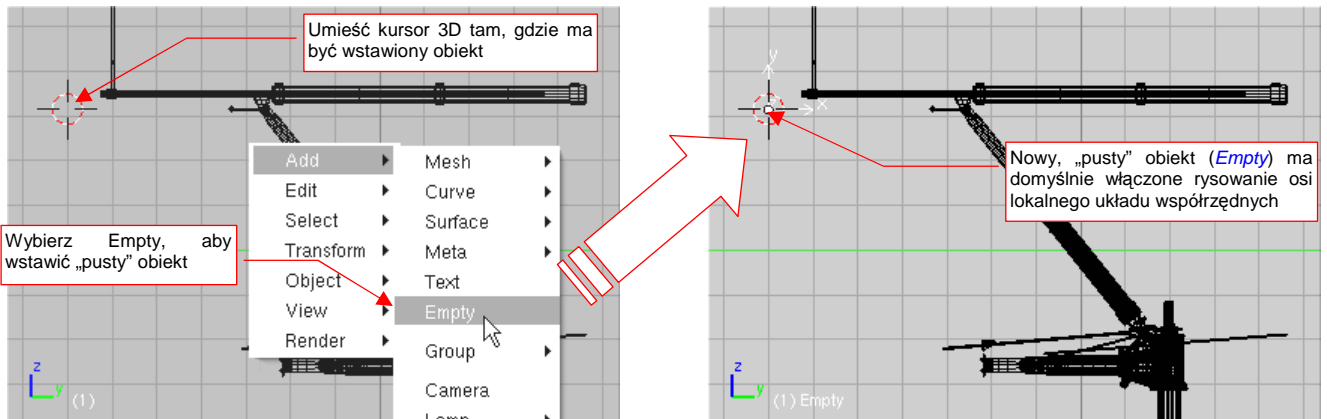
Do wygładzania okręgu i elips najlepiej jest używać schematu obliczeniowego o nazwie **Catmull-Clark**. Nie zmieniaj tego ustawienia w modyfikatorze **Subsurf** (Rysunek 14.2.2). Schemat Catmulla-Clarka całkiem dokładnie odwzorowuje takie krzywizny już dla oryginalnej siatki złożonej z 6-8 wierzchołków, i dla niewielkiego poziomu podziału (**Levels** = 2).

- Jeżeli chcesz dowiedzieć się więcej o krzywych i powierzchniach podziałowych — p. "Powierzchnie podziałowe", str. 571.

14.5 Wstawienie pustego obiektu (*Empty*)

Czasami, podczas pracy z ogranicznikami (*constrains*) przydają się pomocnicze elementy, służące wyłącznie jako „punkty odniesienia” w przestrzeni. Takie obiekty określane są w Blenderze jako „puste” (*Empty*), gdyż nie zawierają żadnej siatki. To samo „opakowanie”: środek (położenie), obrót, a także skala. (Skala nie ma wpływu na sam obiekt *Empty*, ale może mieć na inne, zwykłe elementy „siatkowe”, dla których jest „rodzicem”).

Aby dodać do rysunku nowy obiekt *Empty*, umieść kursor 3D w miejscu, gdzie ma być jego środek. Następnie naciśnij **Space** (lub rozwiń menu *Add* — jest zaraz obok menu *File*). Wywołaj polecenie *Add*→*Empty* (Rysunek 14.5.1):



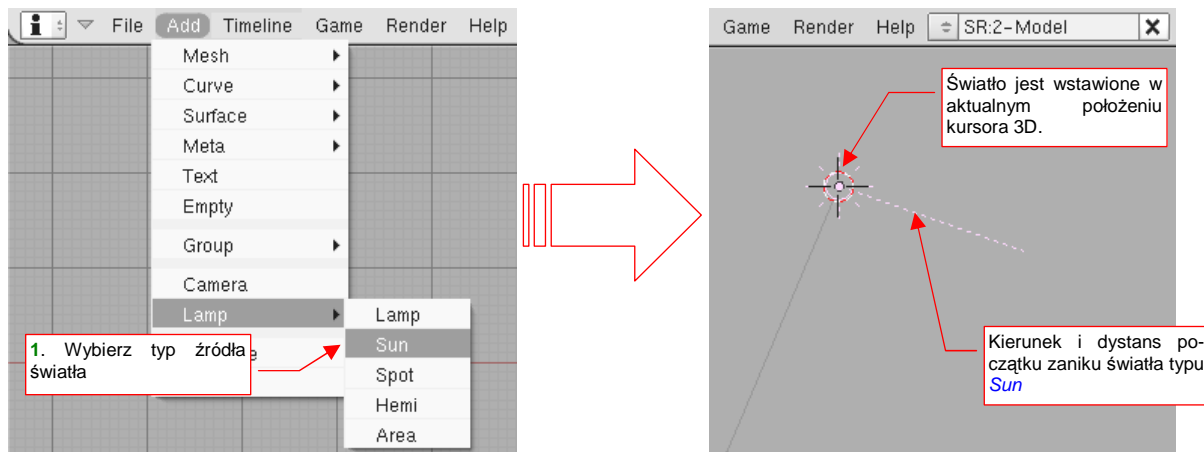
Rysunek 14.5.1 Stworzenie walca (graniastostupa)

Spowoduje to stworzenie nowego obiektu. Aby w ogóle był widoczny, każdy obiekt typu *Empty* ma włączoną opcję rysowania osi lokalnego układu współrzędnych

14.6 Wstawienie źródła światła (*Lamp*)

Umieść kursor 3D w miejscu, gdzie ma być umieszczone źródło światła. (Każdy nowy obiekt jest umieszczany tam, gdzie aktualnie jest kursor). Następnie naciśnij **Space** (lub rozwiń menu *Add* — jest zaraz obok menu *File*).

Wywołaj polecenie *Add*→*Lamp*→*Sun* (Rysunek 14.6.1):

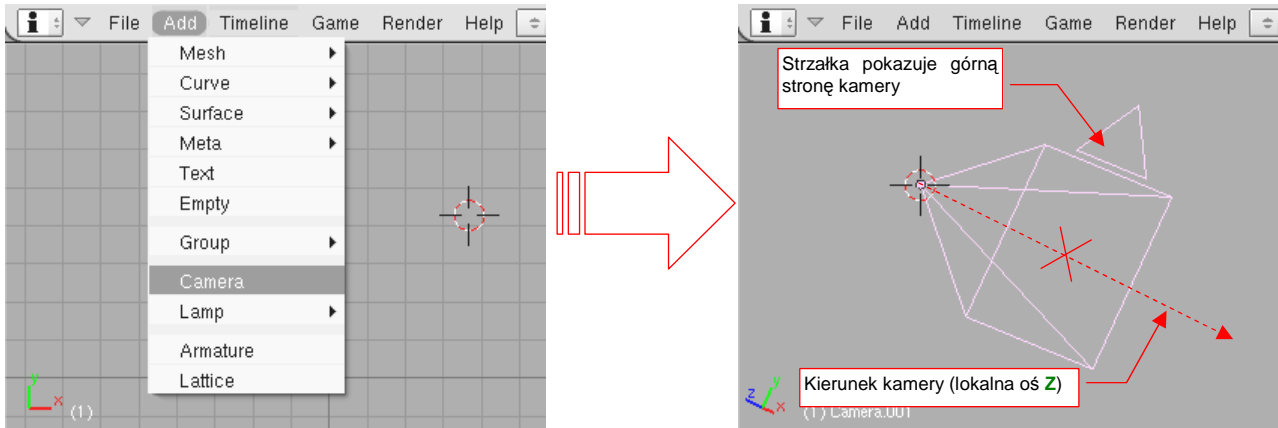


Rysunek 14.6.1 Wstawienie źródła światła (lampy)

Z submenu wybierz jeden z dostępnych typów źródeł światła. (Jeżeli nawet wybierzesz nieprawidłowy — nic złego się nie stanie. Typ źródła światła można zmienić także po jego stworzeniu).

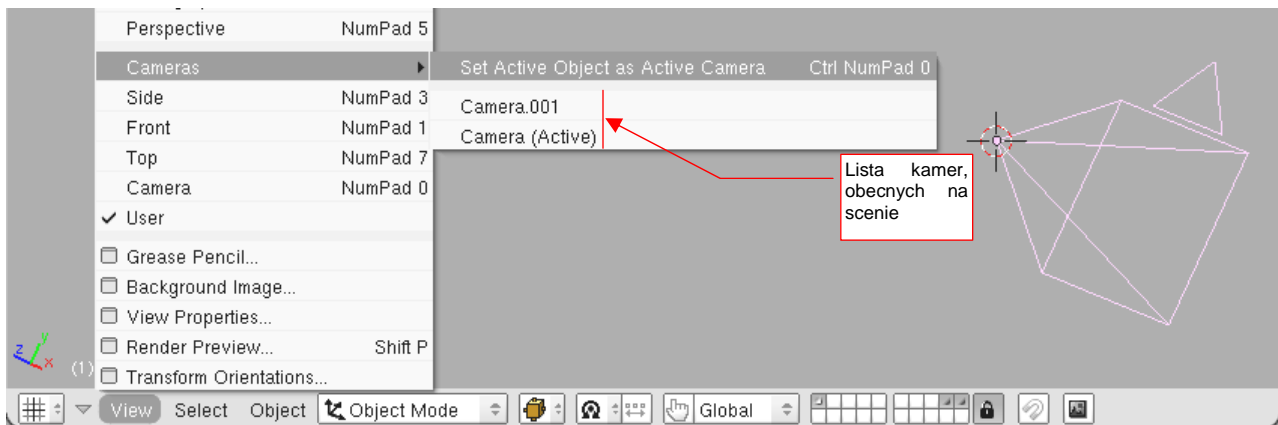
14.7 Wstawienie kamery (*Camera*)

Umieść kursor 3D w miejscu, gdzie ma być umieszczona nowa kamera. (Każdy nowy obiekt jest umieszczany tam, gdzie aktualnie jest kursor). Następnie naciśnij **Space** (lub rozwiń menu *Add* — jest zaraz obok menu *File*). Wywołaj polecenie *Add*→*Camera* (Rysunek 14.7.1):



Rysunek 14.7.1 Wstawienie nowej kamery

Jeżeli dodasz do sceny drugą kamerę, będziesz musiał się pomiędzy nimi przełączać. Służy do tego polecenie *View*→*Cameras* (Rysunek 14.7.2):

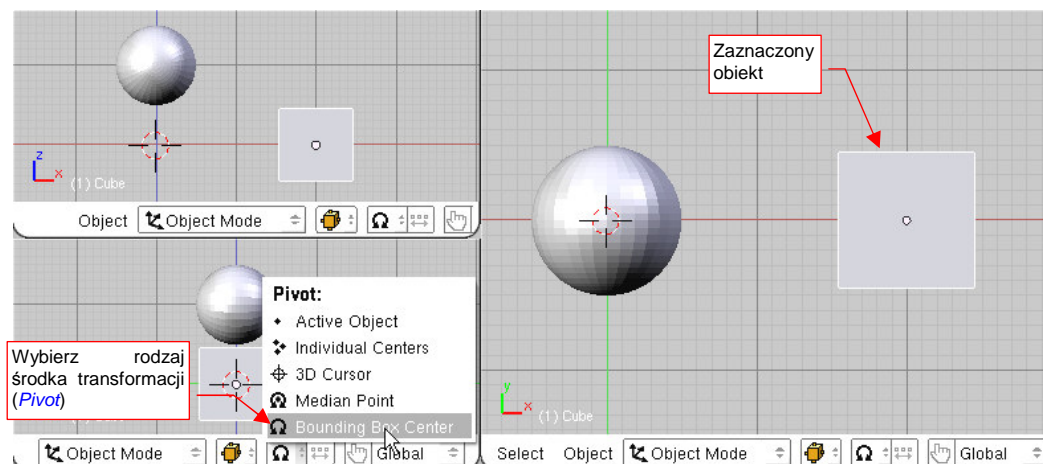


Rysunek 14.7.2 Menu przełączania kamer

W menu rozwijalnym widzisz nazwy wszystkich kamer, istniejących w aktualnej scenie. Wybierając jedną z nich, czynisz ją aktualną (*Active*). Jednocześnie bieżący widok zostaje przełączony na projekcję z tej kamery.

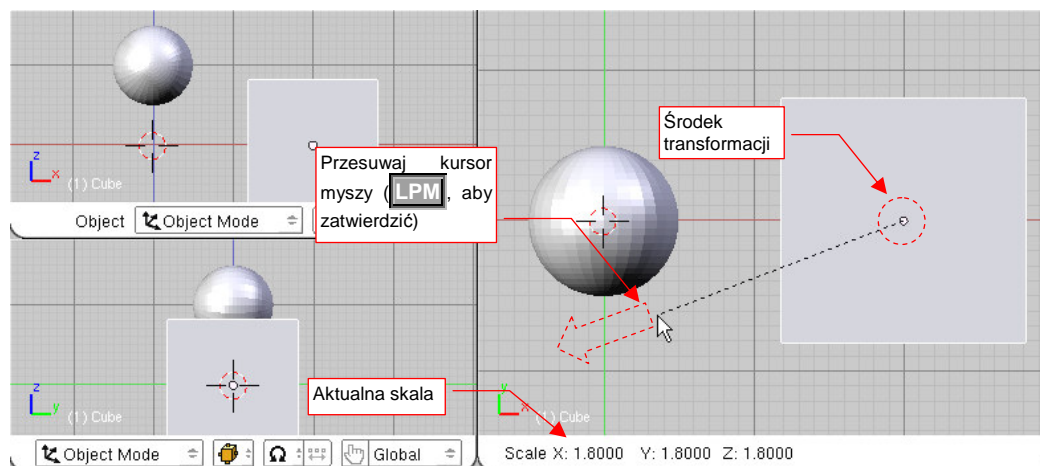
14.8 Skalowanie (*Scale*)

Zaznacz obiekt, które chcesz powiększyć/pomniejszyć, oraz ustal środek transformacji (lista *Pivot*). Praktycznie stosują jeden z dwóch rodzajów środka: *Bounding Box Center* albo *Cursor 3D*. *Bounding Box Center* to środek najmniejszego "pudełka", które zawiera środki wszystkich zaznaczonych obiektów. Zaczniemy od przypadku, gdy środkiem skalowania jest właśnie *Bounding Box Center* (możesz go ustawić z klawiatyry — **[.]**). Na początek zaznaczymy pojedynczy obiekt (Rysunek 14.8.1):



Rysunek 14.8.1 Zaznaczenie obiektu i wybór środka transformacji

Naciśnij **[S]** (*Object* → *Transform* → *Scale*). Przełączysz się w ten sposób w tryb skalowania. Na rysunku pojawi się kreskowana linia od kursora myszy do środka transformacji. Każde przesunięcie myszy będzie powodować zmianę skali zaznaczonego obiektu — proporcjonalnie do zmiany długości kreskowanej linii. Dokonywane zmiany są pokazywane na bieżąco we wszystkich oknach Blendera (Rysunek 14.8.2):



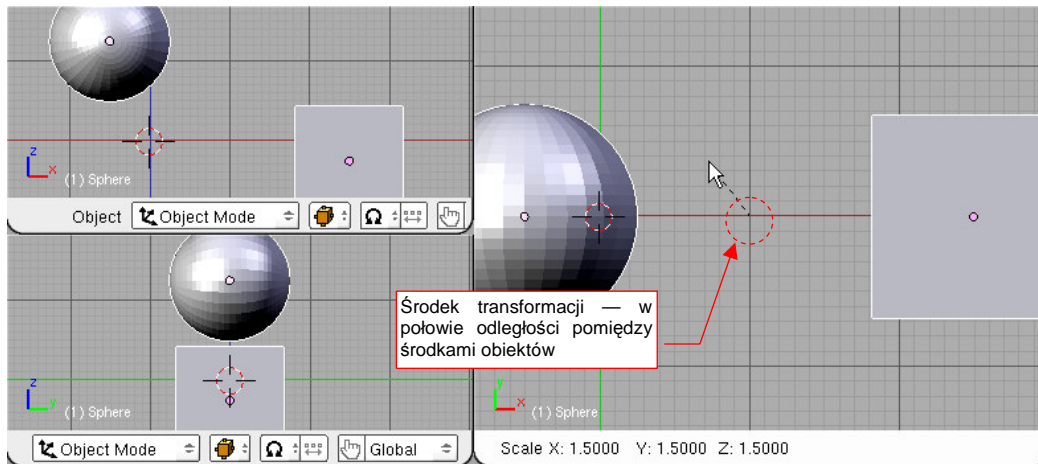
Rysunek 14.8.2 Powiększanie pojedynczego obiektu

W nagłówku aktywnego okna jest wyświetlana aktualna zmiana skali wzdłuż każdej osi. Aby zatwierdzić nowe wymiary obiektu (zakończyć transformację), należy kliknąć **[LPM]**. Aby zrezygnować z operacji, możesz w każdej chwili nacisnąć **[Esc]**.

Podczas zmiany skali możesz dodatkowo trzymać wciśnięty klawisz **[Ctrl]**. Spowoduje to skokową zmianę proporcji (np. co 0.1). Jeżeli będziesz trzymał wciśnięty klawisz **[Shift]** — zmiana skali na ekranie "wyhamuje", pozwalając na dokładniejsze ustalenie. Wreszcie kombinacja obydwu — **[Shift]-[Ctrl]** — umożliwi skokową zmianę skali o niewielkie odległości (np. co 0.01).

Zwróć uwagę, że gdy zaznaczysz pojedynczy obiekt, całe to *Bounding Box Center* wypada po prostu w środku obiektu (Rysunek 14.8.2). (To dlatego, że dla jednego punktu *Bounding Box* ma zerowy rozmiar).

A jak działa *Bounding Box Center* dla grupy obiektów? Spróbujmy na naszym rysunku powiększyć sześcian i kulę (Rysunek 14.8.3):



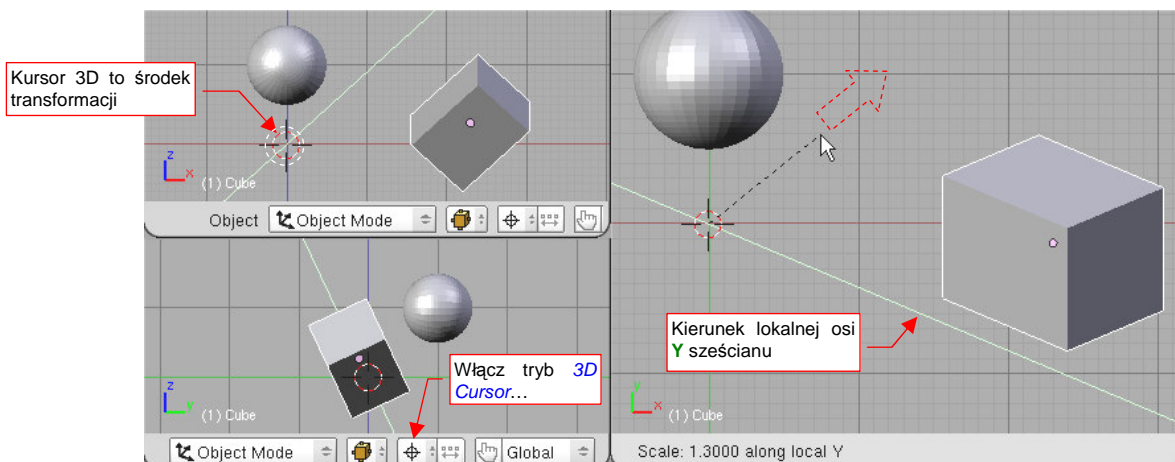
Rysunek 14.8.3 Zmiana skali grupy obiektów, dla *Pivot* = *Bounding Box Center*

Obydwa obiekty jednocześnie oddalają się od siebie i zmieniają rozmiar. Dzieje się tak dlatego, że środkiem transformacji jest punkt w połowie odległości pomiędzy środkami kuli i sześcianu.

- *Bounding Box Center* dla obiektów leży w środku ich punktów odniesienia, a nie siatek (tzn. kształtu)

Jeżeli w trakcie przesuwania naciśniesz **X**, **Y**, lub **Z** — ograniczysz zmianę skali do jednej z tych osi. W pozostałych kierunkach żaden rozmiar nie ulegnie zmianie. Po pierwszym naciśnięciu klawisza wybrana jest oś globalna. Jeżeli jednak jeszcze raz naciśniesz ten sam klawisz — przełączysz się na lokalną oś obiektu. (Odpowiada to poleceniom *Object*→*Transform*→*Scale on Axis*). To rozróżnienie pomiędzy osią globalną i lokalną ma swój sens wtedy, gdy obiekt jest w jakiś sposób obrócony.

Rysunek 14.8.4 pokazuje przykład skalowania wzdłuż jednej osi — **Y**. Wybrałem tu także inny sposób określenia środka — tym razem jest to *3D Cursor*. Obróciłem trochę sześcian, by oś lokalna różniła się od globalnej. Przełączyłem typ punktu odniesienia na *3D Cursor* (na klawiaturze: **I**).



Rysunek 14.8.4 Zmiana skali w jednym kierunku, i względem kursora 3D

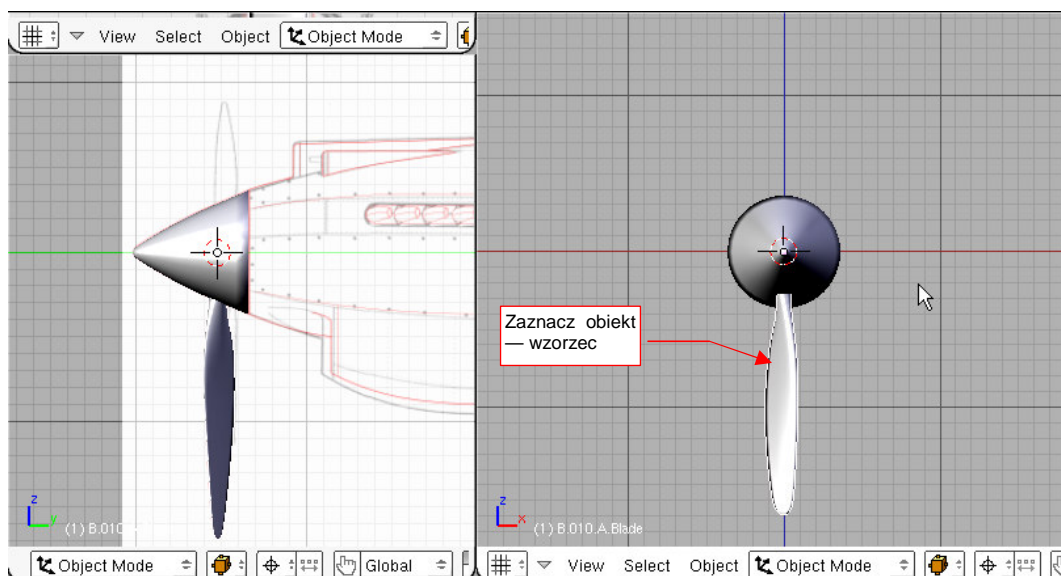
Nacisnąłem **S**, **Y**, **Y**. Trzymając wciśnięty klawisz **Ctrl** (aby uzyskać dokładną zmianę skali) zwiększyłem grubość sześcianu tylko wzdłuż jego lokalnej osi **Y** o 1.3.

14.9 Powielanie obiektu (*Duplicate*)

W Blenderze istnieją dwa polecenia do tworzenia duplikatów obiektu:

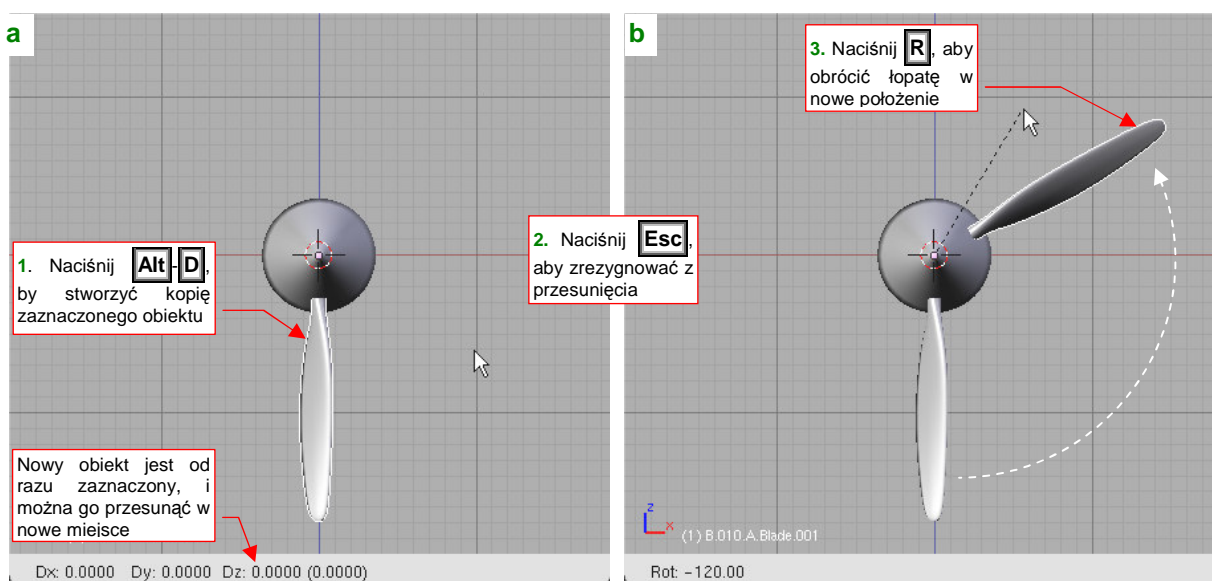
1. Duplikat jest stworzony jako kopia "powiązana" (**Object**→**Duplicate Linked**, **Alt-D**). Kopia może mieć oddzielną skalę, położenie, obrót, ale korzysta nadal z tej samej siatki, co wzorzec. (W efekcie ma zawsze ten sam kształt, co oryginał);
2. Duplikat jest stworzony jako zwykła kopię (**Object**→**Duplicate**, **Shift-D**). Kopia jest obiektem, który w żaden sposób nie zależy od oryginału.

Wywołanie i obsługa obydwu poleceń jest identyczna, więc opiszę je na przykładzie tworzenia kopii "powiązanej". Rysunek 14.9.1 pokazuje sytuację początkową. Zaznaczyliśmy obiekt — wzorzec: łopatę śmigła. Utworzymy jej dwie powiązane kopie, tworząc trójłopatowe śmigło.



Rysunek 14.9.1 Łopata śmigła, do której dodamy dwie dalsze — jako powiązane kopie

Naciśnij teraz **Alt-D** (**Object**→**Duplicate Linked**). To stworzyło nowy obiekt. Blender nadaje mu unikalną nazwę poprzez dołączenie do nazwy oryginału końcówki w rodzaju ".001". Od razu także proponuje dokonać jego przesunięcia (Rysunek 14.9.2a):

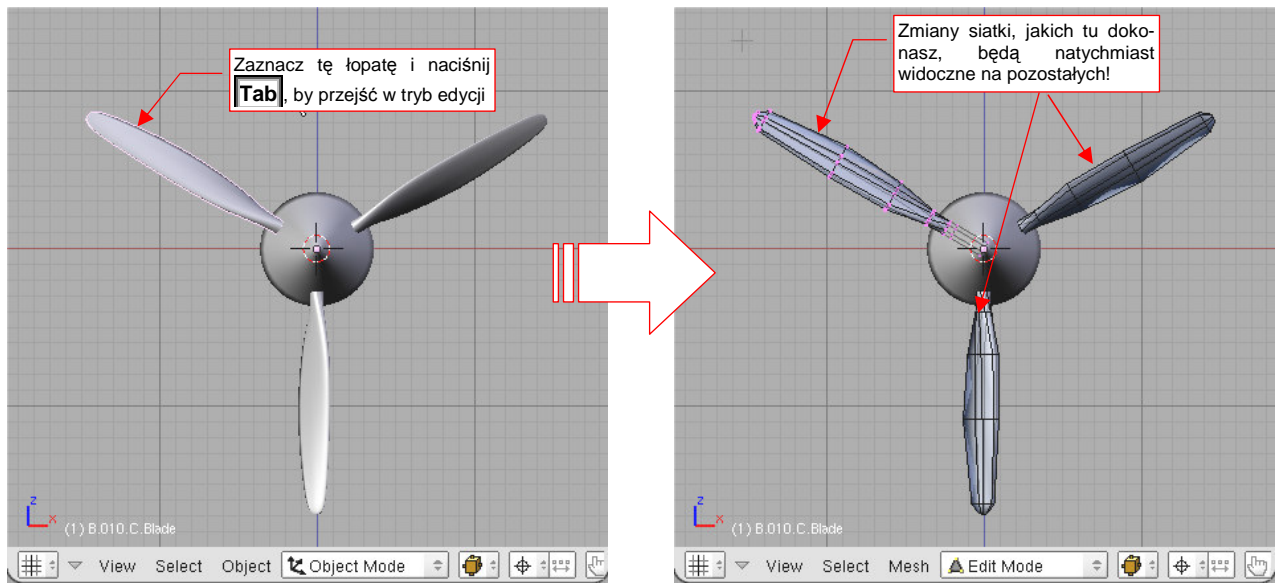


Rysunek 14.9.2 Powielenie (poprzez obrót)

Jeżeli nie chcesz przesunąć nowego obiektu, a np. obrócić, wystarczy nacisnąć **Esc**. To przerwie przesuwanie kopii, ale pozostawi ją jako zaznaczoną. Teraz wystarczy wybrać polecenie dowolnej innej transformacji, którą chcesz zastosować — na przykład **R**, by nowy obiekt obrócić (Rysunek 14.9.2b).

- Zwykłą kopię obiektu tworzysz identycznie, tyle tylko że zaczynasz poleceniem **Shift-D** (**Object**→**Duplicate**)

Jaka jest w takim razie praktyczna różnica pomiędzy kopią zwykłą, a kopią powiązaną? Pokażę to na przykładzie śmigła (w którym dodałem już także trzecią łopatę). Wszystkie trzy łopaty są kopiami połączonymi. Gdy zaznaczysz jedną z nich, i wejdiesz w tryb edycji siatki — siatka stanie się widoczna na wszystkich trzech (Rysunek 14.9.3):



Rysunek 14.9.3 Obiekty powiązane — siatka jest zmieniana we wszystkich równocześnie

Gdy teraz zmienisz kształt siatki — zmiana będzie natychmiast widoczna także na pozostałych obiektach powiązanych. (Więcej na ten temat — patrz "Struktura modelu w Blenderze", str. 565.) Jeżeli przyjrzy się dokładnie, zobaczysz że tylko wybrana łopata ma zaznaczone wierzchołki. Siatki pozostałych są widoczne, ale składają się z samych krawędzi. Nie są także wygładzone. (Choć to akurat jest jakiś mankament wyświetlania w tym trybie — po powrocie w tryb obiektu wygładzenie znów stanie się widoczne).

14.10 Nadanie obiektowi nazwy

Każdy element rysunku Blendera musi mieć unikalną nazwę. Blender nadaje pewne nazwy domyślne, ale w bardziej złożonym modelu konieczne jest wprowadzenie nazewnictwa w sposób bardziej kontrolowany. Stąd, po utworzeniu każdego nowego obiektu powinieneś nazwać go na nowo.

Zacznijmy od zasad: jaki system nazewnictwa przyjąć dla elementów naszego modelu? Maksymalna długość nazwy w Blenderze to 20 znaków. Proponuję w związku z tym stosować tekst składający się z następujących segmentów:

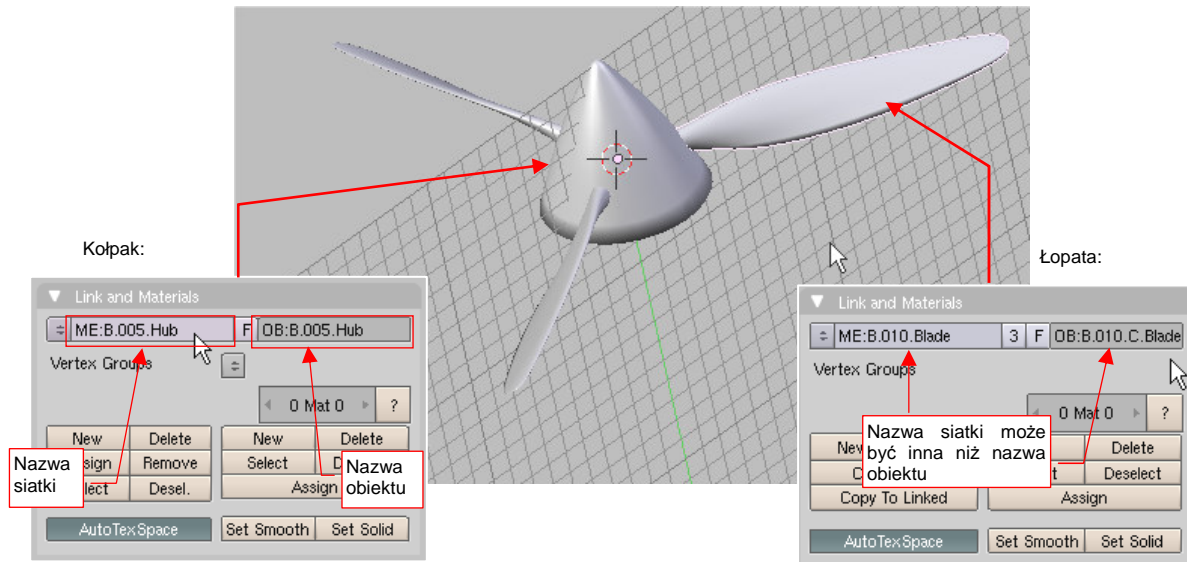
- segment pełniący rolę unikalnego identyfikatora: składa się z 2 lub 3 części, oddzielonych kropkami:
 - o **1 znak**: znak, który będzie pełnił rolę wyróżnika wersji samolotu. Wszystkie części w pojedynczym pliku Blendera będą używać takiej samej litery. Na przykład w modelu P-40B nazwy będą zaczynać się od "B". Okaże się to przydatne później¹. Wyjątkiem są elementy specjalne: uchwyty i inne artefakty, używane wyłącznie do animacji modelu. Te mają przedrostek „X”
 - o **3 znaki**: unikalny numer części. Wydaje mi się, że dobrze jest wydzielić główne podzespoły za pomocą pierwszego znaku. Pozostałe dwa — może to być cyfra, określająca numer części w obrębie zespołu. Oczywiście, także można starać się, aby elementy podzespołów miały zbliżone numery. W takim przypadku radziłbym zwiększać numerację nie co 1, a co 5 (np. "A05", "A10", "A15", "A20",...). W ten sposób pozostawiamy sobie miejsce w numeracji na ewentualne dodatkowe elementy. (Podczas pracy często się okazuje, że trzeba przekonstruować gotowy zespół. Takie luki w numeracji są wtedy jak znalazł!);
 - o **1 znak**: opcjonalny znacznik "powielenia". Może to być "L" lub "R" dla części lewej lub prawej, które są symetryczne. Może także być "nr kolejny", np. dla 3 łopat śmigła jest to "A", "B", "C";
- segment z dodatkową, "ludzka" nazwą (na ile się zmieści w 15-13 znakach, które pozostały). Rola tego fragmentu jest pomocnicza — ma ułatwić identyfikację przez użytkownika. Sądzę, że w związku z tym, że nigdy nie wiesz, do kogo trafi Twoja praca, najlepiej jest stosować nazwy angielskie.

Zgodnie z powyższymi regułami, kołpak i jego siatka (Rysunek 14.10.1) noszą nazwę **"B.005.Hub"**:

- segment identyfikatora:
 - o "B" — przedrostek, którego będę używał do wszystkich części P-40B;
 - o "005" — numer tej części: "0" będę używał dla podzespołu śmigła, a "05" — to numer kolejny, nadany z zachowaniem zasady "odstępu w numerach co 5";
- segment nazwy: angielska (mam nadzieję, że się nie mylę) nazwa tej części — "Hub". (Szczęśliwie jest bardzo krótka);

¹ Gdy będziesz tworzył, w innym pliku Blendera, model P-40D, nadasz jego elementom przedrostek "D". Podczas pracy nad wariantem "D" może okazać się przydatne np. dołączenie do rysunku kadłuba z P-40B. (Aby porównywać go z kadłubem P-40D). Dzięki temu przedrostkowi unikniesz konfliktu nazw tych części!

Aby zmienić nazwę, zaznacz obiekt, i w zestawie edycji (*Editing* — **F9**), w panelu *Link and Materials*, wpisz nazwę w obydwie pola: nazwy siatki i nazwy obiektu (Rysunek 14.10.1). (Przedrostki "ME:" i "OB:" nadaje sam Blender — nie masz na nie wpływu). Zazwyczaj obydwie nazwy są identyczne (por. nazwę kołpaka śmigła, Rysunek 14.10.1). Dlaczego trzeba nadawać oddzielne nazwy obiektowi i siatce? Skąd w ogóle taki podział, i czym w Blenderze jest w takim razie "obiekt"? Jeżeli nurtują Cię takie pytania, zajrzyj na str. 565, do sekcji "Struktura modelu w Blenderze".

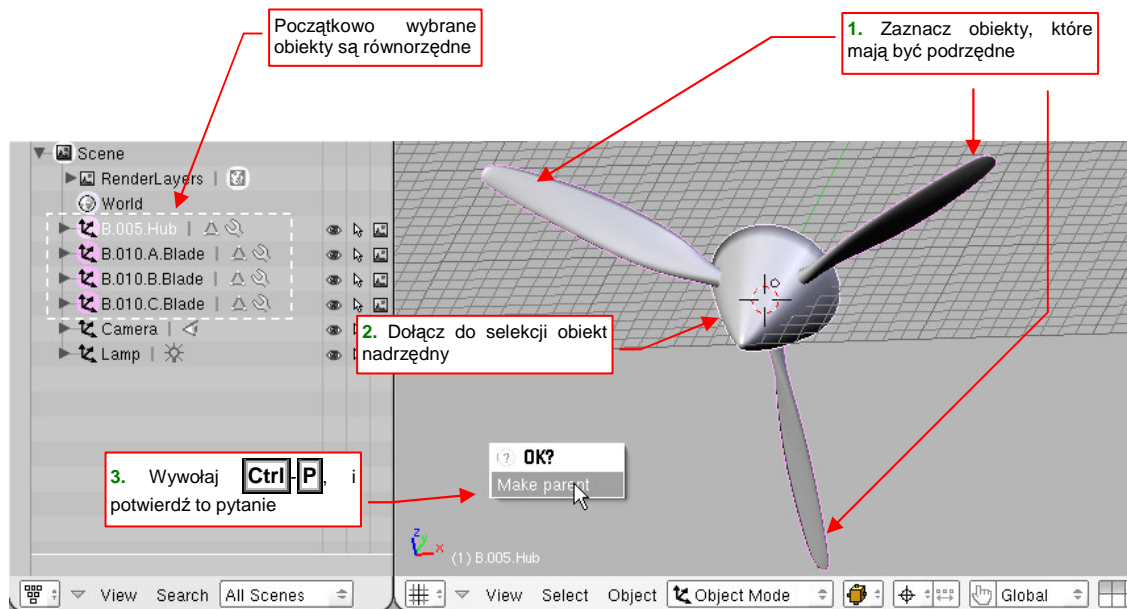


Rysunek 14.10.1 Przykłady nazw obiektów i siatek

Rysunek 14.10.1 pokazuje także, oprócz kołpaka, szczegóły łopaty śmigła. Nazwa jej siatki (**B.010.Blade**) i nazwa obiektu (**B.010.C.Blade**) różnią się od siebie. Stało się tak dlatego, że wszystkie trzy łopaty śmigła (**B.010.A.Blade**, **B.010.B.Blade**, **B.010.C.Blade**) używają jednej i tej samej siatki, o nazwie **B.010.Blade**. Są one tzw. kopiami powiązаныmi, które, w wyniku użycia tej samej siatki, będą miały zawsze ten sam kształt. (Więcej szczegółów na temat kopii powiązanych — str. 786)

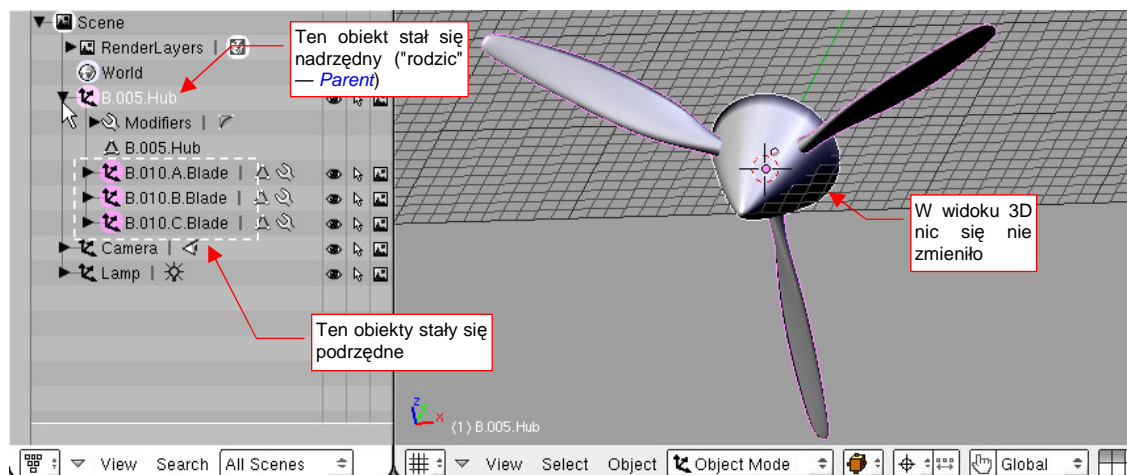
14.11 Przypisanie do hierarchii (*Parent*)

Najpierw zaznacz obiekty, które chcesz uczynić podrzędnymi. Gdy wszystkie już zaznaczysz, włącz do selekcji obiekt, który ma być ich obiektem nadrzędnym. Koniecznie upewnij się, że jego nazwa pojawiła się w lewym, dolnym narożniku ekranu, (tzn. że stał się obiektem aktywnym). Następnie naciśnij **Ctrl-P** (*Object→Parent→Make Parent*). Rysunek 14.11.1 pokazuje ten moment w przypisywaniu trzech łopat śmigła (**B.010.*.Blade**) do kołpaka (**B.005.Hub**):



Rysunek 14.11.1 Tworzenie powiązań hierarchicznych — przygotowania

Gdy potwierdzisz pytanie *Make Parent*, które pojawi się na ekranie (Rysunek 14.11.1), nic właściwie widocznego na scenie się nie zmienia. Dopiero w podglądzie struktury sceny zobaczysz różnice (Rysunek 14.11.2):



Rysunek 14.11.2 Utworzone powiązanie hierarchiczne

Bezpośrednim efektem przypisania łopat do kołpaka jest ułatwienie w manipulacji obiektami na scenie. Wcześniej, aby obrócić czy przesunąć ten zespół, musiałbyś dokonać odpowiedniej transformacji oddzielnie na każdym obiekcie. Przypisując łopaty do kołpaka, "skleiłeś" je ze sobą. Od tej chwili, gdy tylko obrócisz lub przesuńiesz kołpak, łopaty będą także się obracać i przesuwać — jak w prawdziwym śmigle.

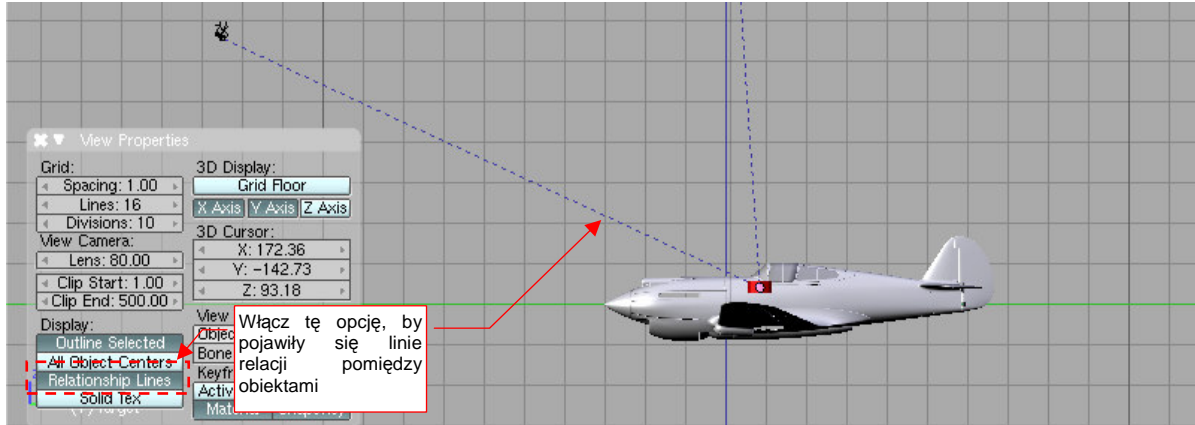
Nic nie stoi na przeszkodzie, aby później uczynić kołpak śmigła obiektem podrzędnym kadłuba. W ten sposób stworzysz bardzo istotną hierarchię części modelu.

- Pamiętaj, że powinieneś doprowadzić do sytuacji, gdy cały samolot ma jedną część główną, do której należą, pośrednio lub bezpośrednio, wszystkie podzespoły. Tylko takim modelem będziesz mógł wygodnie manipulować, przygotowując statyczną scenę czy film.

- Aby wyłączyć obiekt z hierarchii — zaznacz go i naciśnij **Alt-P** (*Object → Parent → Clear Parent*).

14.12 Wizualizacja powiązań pomiędzy obiektami (*Relationship Lines*)

W Blenderze można włączyć pokazywanie linii relacji pomiędzy obiektami. Chodzi wizualizację powiązań hierarchicznych (*parent*) i ograniczeń (*constraints*). Robi się to we właściwościach widoku *3D View* (*View*→*View Properties*). Należy tam włączyć opcję *Relationship Lines* (Rysunek 14.12.1):

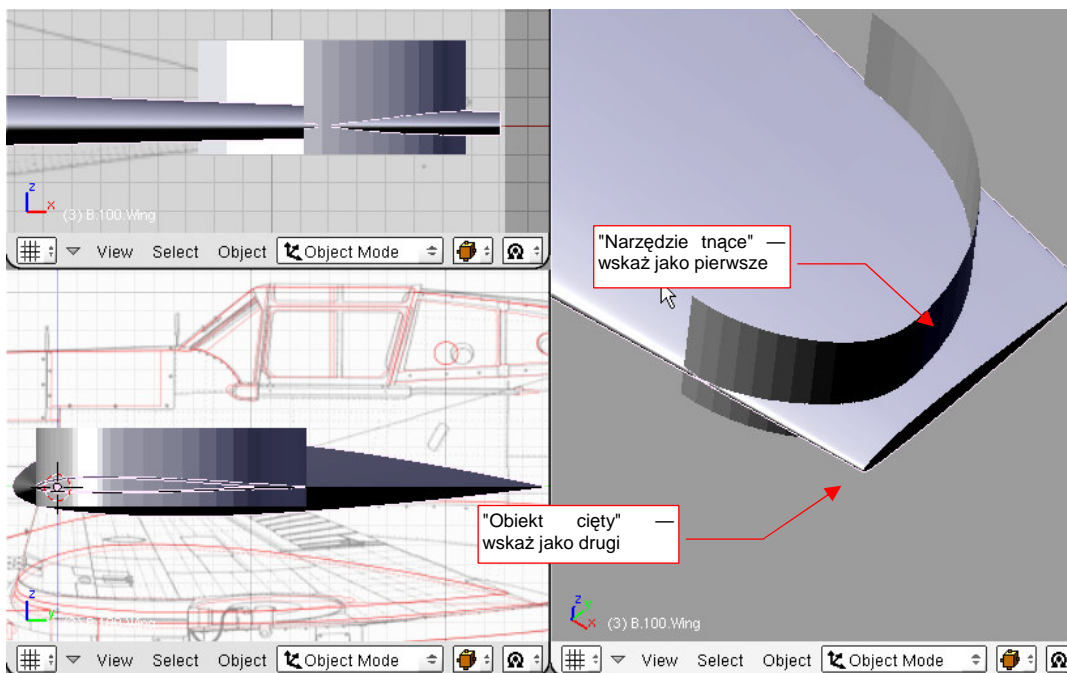


Rysunek 14.12.1 Włączenie wizualizacji powiązań obiektów

Relacje hierarchiczne (*parent*) są oznaczone liniami czarnymi. Linie niebieskie to powiązania wynikające z ograniczeń (*constraints*) pomiędzy obiektem i jego celem.

14.13 Wyznaczenie krawędzi przecięcia dwóch powłok

Zaznacz najpierw dwa obiekty, które się przenikają (Rysunek 14.13.1) :



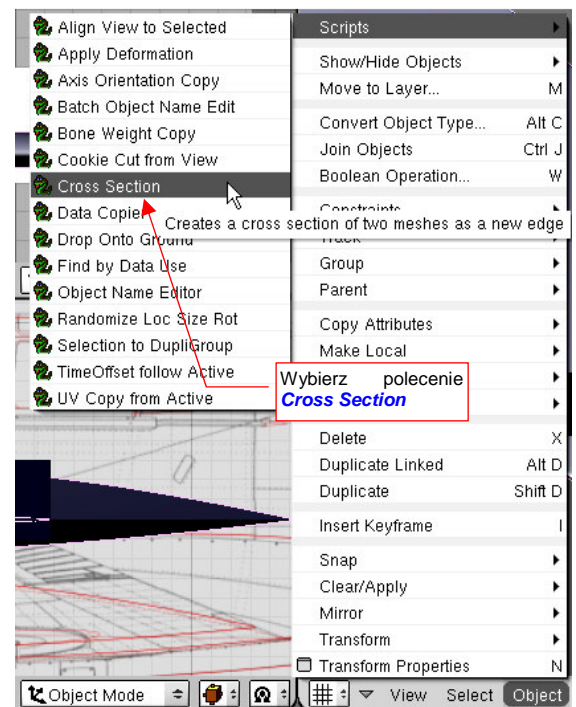
Rysunek 14.13.1 Dwie powierzchnie, dla których ma być wyznaczona krawędź przecięcia

Do wyznaczenia krawędzi przecięcia dwóch dowolnych powierzchni służy skrypt *Pythona*, który opracowałem¹.

Jego wywołanie znajdziesz w menu **Object**→**Scripts** (Rysunek 14.13.2) — jest to skrypt **Cross Section**. (To polecenie pojawi się w menu, gdy w konfiguracji Blendera wskażesz odpowiedni folder dla skryptów *Pythona* — por. str. 759.

Gdyby jednak coś Ci z tym przypisaniem nie wyszło — załaduj udostępniony wraz z tą książką (w pliku *source.zip*) plik `Source\Python\CrossSection.py` do okna edytora tekstu, i naciśnij **Alt-P**.)

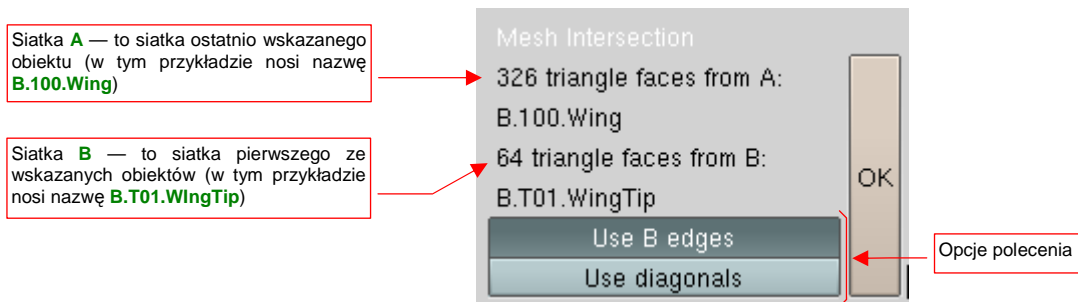
Wyznaczona przez skrypt krawędź przecięcia zostanie dodana do siatki ostatniego ze wskazanych obiektów.



Rysunek 14.13.2 Wywołanie skryptu **Cross Section**

¹ Owszem, Blender ma zespół funkcji, pozwalających wykonać na siatkach operacje "Boole'a": różnicę, sumę, część wspólną. Mankamentem jednak wszelkich takich rozwiązań jest siatka, uzyskana jako rezultat. Zawiera wiele wierzchołków, których nie potrzeba do dalszej pracy, a które tylko niepotrzebnie zwiększają liczbę ścian i komplikują model. Osobiście wolę mieć pełną kontrolę nad siatką, którą formuję. Stąd przygotowałem skrypt, który wyznacza tylko krawędź przecięcia. Co później z nią zrobisz — to zależy wyłącznie od Ciebie. Zazwyczaj, przy dobrze zaplanowanych siatkach, wkomponowanie nowej krawędzi w dotychczasową powierzchnię nie wymaga dużo pracy. Usuniesz parę ścian, stworzysz — korzystając z wierzchołków wyznaczonej krawędzi — kilkanaście nowych. W ten sposób uzyskujesz zaplanowany kształt za pomocą ładnej — bo prostej i regularnej — siatki. (Jest takie stare inżynierskie przysłowie: "co oko lubi, to i konstrukcja lubi")

Nim jednak zobaczysz rezultat — pojawi się okno dialogowe tego polecenia (Rysunek 14.13.3):

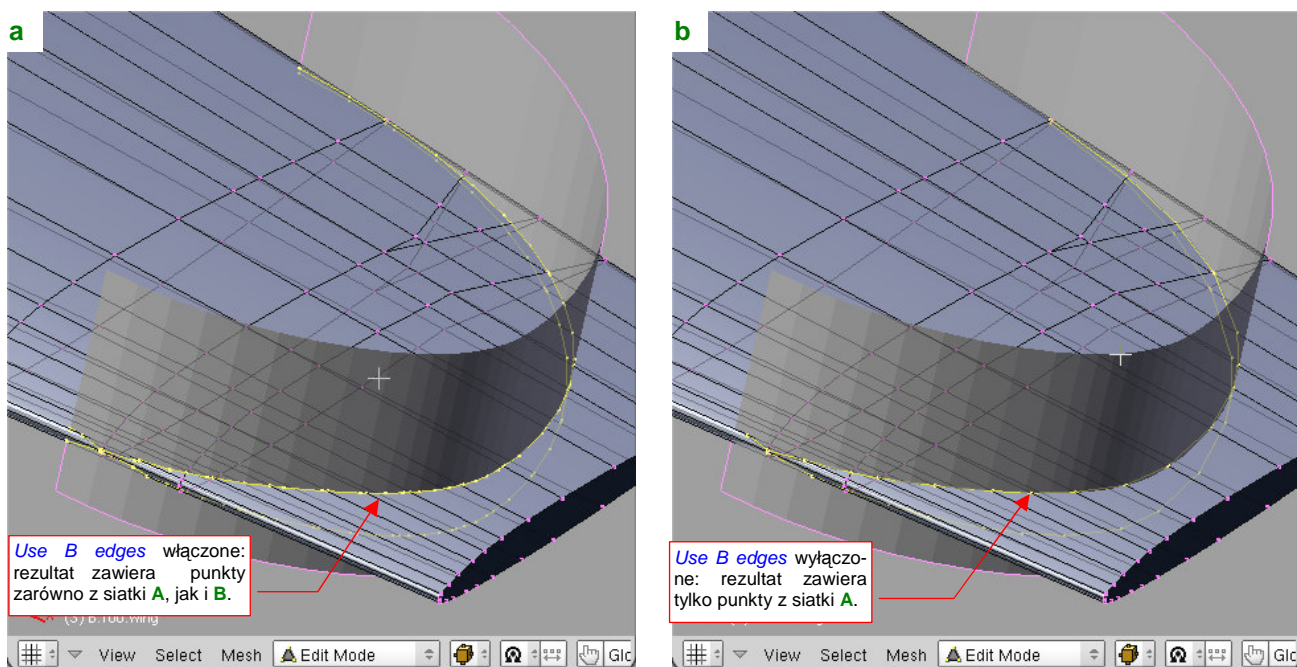


Rysunek 14.13.3 Okno dialogowe polecenia *Cross Section*

Okno *Mesh Intersection* podaje liczbę trójkątnych ścian użytych do obliczeń, z każdego z obiektów. Siatką **A** nazywana jest siatka ostatniego wskazanego obiektu (czyli obiektu aktywnego). Siatką **B** nazywana jest siatka pierwszego ze wskazanych obiektów. Zazwyczaj wygodnie jest traktować siatkę **B** jako "narzędzie", a siatkę **A** — jako "przecinany obiekt".

- Jeżeli na siatce zaznaczono wcześniej (w *Edit mode*) jakieś wierzchołki — przecinane będą wyłącznie zaznaczone ściany. Jeżeli nic nie jest zaznaczone — użyte są wszystkie ściany siatki

Naciśnij *Mesh Intersection:OK*, aby wyznaczyć krawędź przenikania. Skrypt przełącza Blendera w tryb edycji aktywnego obiektu (obektu **A**). Krawędź jest nową linią wierzchołków. Linia ta jest od razu zaznaczona. Jeżeli nie jesteś zadowolony z rezultatu, wystarczy jeden klawisz (**X**, lub **Delete**), by wymazać wynik operacji. Rysunek 14.13.4a) przedstawia rezultat, uzyskany w wyniku domyślnych ustawień parametrów operacji (por. Rysunek 14.13.3):

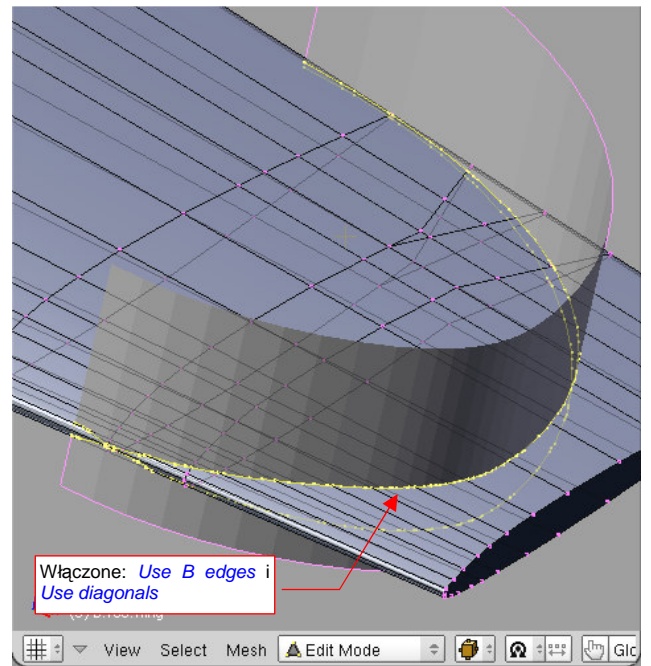


Rysunek 14.13.4 Rezultaty przecięcia — w zależności od ustawienia *Mesh Intersection:Use B edges*

Skrypt *Cross Section* wyszukuje punkty przecięcia krawędzi siatki **A** ze ścianami siatki **B**. Jeżeli włączona jest opcja *Mesh Intersection:Use B edges*, to wyszukiwane są także punkty przecięcia krawędzi **B** ze ścianami **A**. (W takim przypadku krawędź przecięcia zawiera więcej punktów).

Podczas obliczeń każda czworokątna jest dzielona przekątną (*diagona*), aby zamienić ją na dwie ściany trójkątne. (Stąd w informacjach o siatce w oknie *Mesh Intersection* podawana jest liczba ścian trójkątnych).

Ostatnia opcja okna dialogowego (Rysunek 14.13.3) to *Mesh Intersection:Use diagonals*. Ta rzadko używana możliwość powoduje włączenie do rezultatu także punktów przecięcia tymczasowych przekątnych czworokątnych ścian z drugą siatką. Rezultat takiej operacji to bardzo "gęsta" linia wierzchołków (Rysunek 14.13.5):

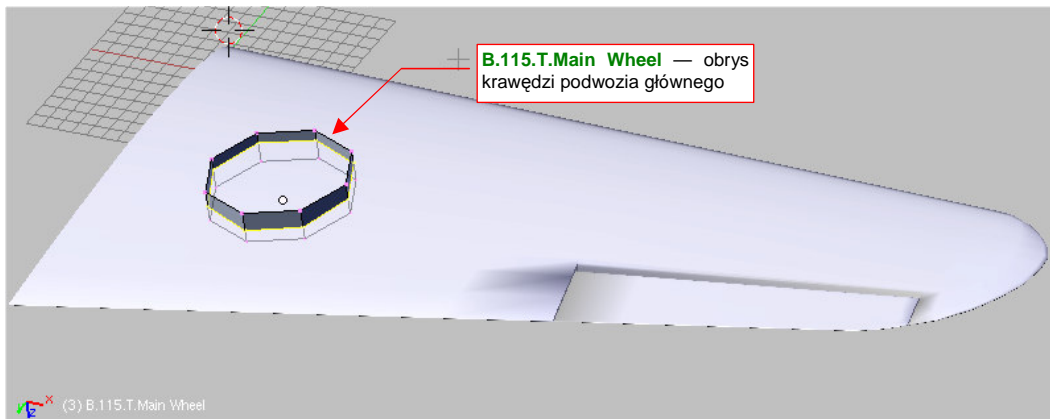


Rysunek 14.13.5 Rezultat włączenia *Mesh Intersection:Use diagonals*

14.14 Scalanie obiektów (*Join Objects*)

Scalenie dwóch obiektów oznacza scalenie ich siatek. Jednocześnie wszystkie wskazane obiekty, poza obiektem aktywnym, przestają istnieć.

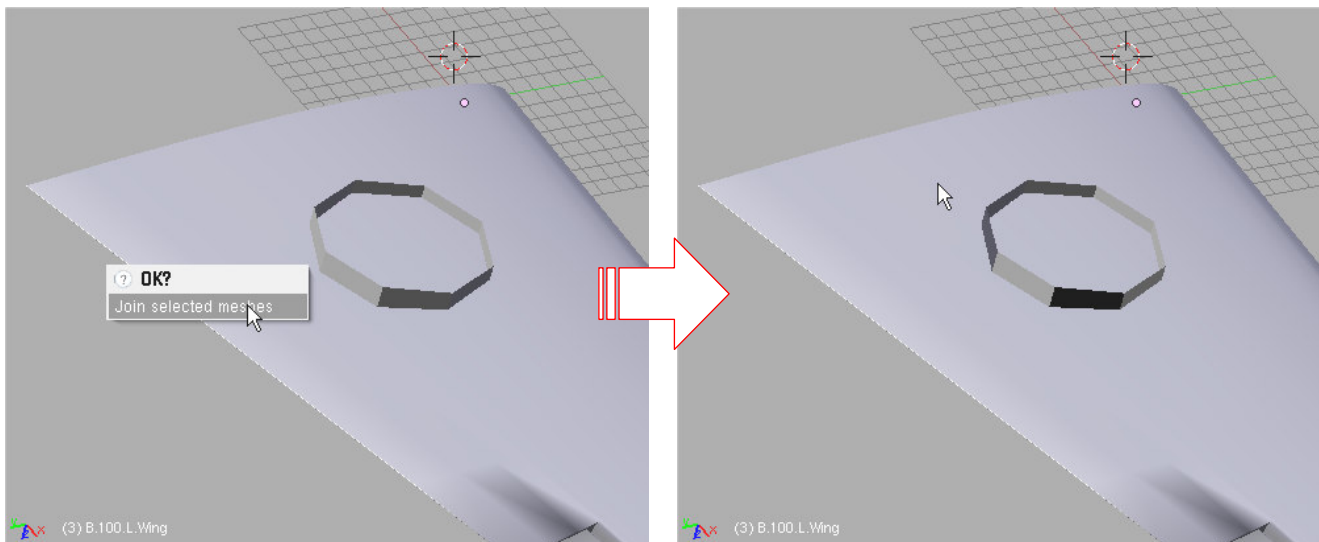
Operację tę wykonuję często by przenieść uzyskaną krawędź przecięcia (p. str. 802) do drugiej siatki. Rysunek 14.14.1 przedstawia jedną z takich sytuacji. Pomocniczy obiekt (**B.115.T.Main Wheel**) to obrys (nie wygładzony) otworu na koło główne. Jego siatka zawiera także krawędź przecięcia z powierzchnią płata:



Rysunek 14.14.1 Pomocniczy obiekt — zawiera krawędź przecięcia ze skrzydłem

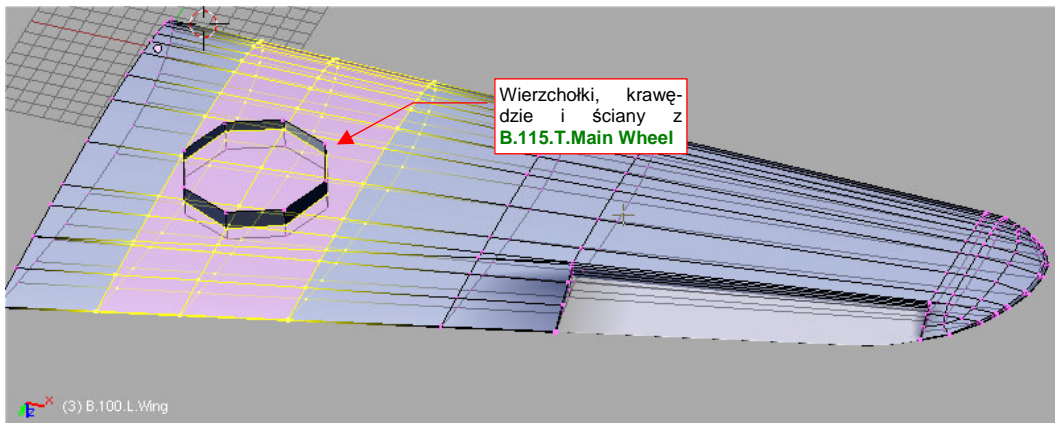
Tylko ta krawędź jest potrzebna do dalszej pracy. Powinna się jednak znaleźć w siatce skrzydła, aby można z niej było stworzyć otwór na podwozie główne. W tym celu należy scalić obydwie obiekty.

Zaznacz najpierw obiekt, który zniknie, a na koniec — obiekt, który ma pozostać po tej operacji. (Podczas scalania obiekt aktywny "wchłania" pozostałe). Następnie naciśnij **Ctrl-J** (*Object→Join Objects*), i potwierdź pytanie "*Join selected meshes*" (Rysunek 14.14.2):



Rysunek 14.14.2 Połączenie siatek dwóch obiektów

I co, wszystko wygląda jak przedtem? Niby tak, ale przejdź teraz w tryb edycji aktywnego obiektu (naciśnij **Tab** — Rysunek 14.14.3):



Rysunek 14.14.3 Rezultat — połączone siatki

Pomocniczy obiekt (**B.115.T.Main Wheel**) przestał istnieć — elementy jego siatki zostały dodane do siatki skrzydła. Możesz teraz usunąć niepotrzebne wierzchołki, i masz w płacie krawędź do wykonania otworu.

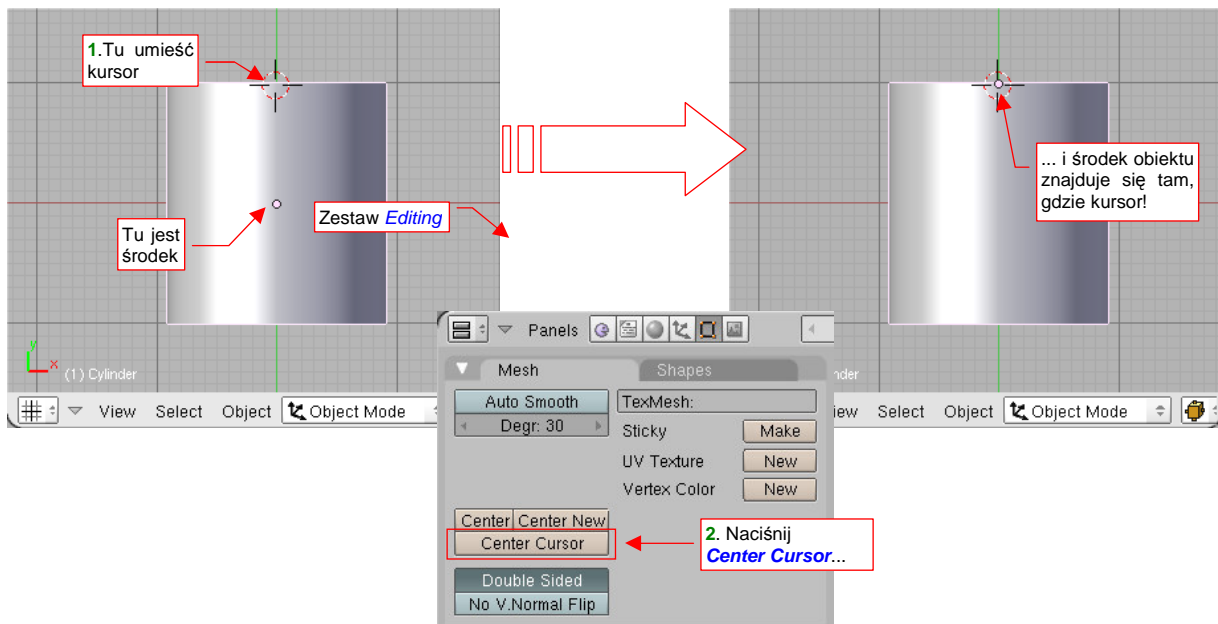
- Operacją odwrotną do scalenia siatek obiektów jest wydzielenia siatki ([Separate](#), str. 887).

14.15 Zmiana położenia środka obiektu (*Center*)

Punkt odniesienia obiektu, używany dla określenia położenia, obrotu, i skali, nazywany jest w Blenderze **środkiem** obiektu. Nazwa wywołuje czasem niewłaściwe skojarzenia, bo środek obiektu często wcale nie leży w jego środku geometrycznym.

Stosunkowo często pojawia się konieczność zmiany położenia środka obiektu względem jego siatki. Najszybciej możesz to zmienić w następujący sposób (Rysunek 14.15.1):

- umieść kursor 3D w miejscu, w którym chcesz umieścić nowy środek obiektu;
- przełącz się na zestaw paneli *Editing* (**F9**)
- na panelu *Mesh* naciśnij przycisk *Center Cursor*;



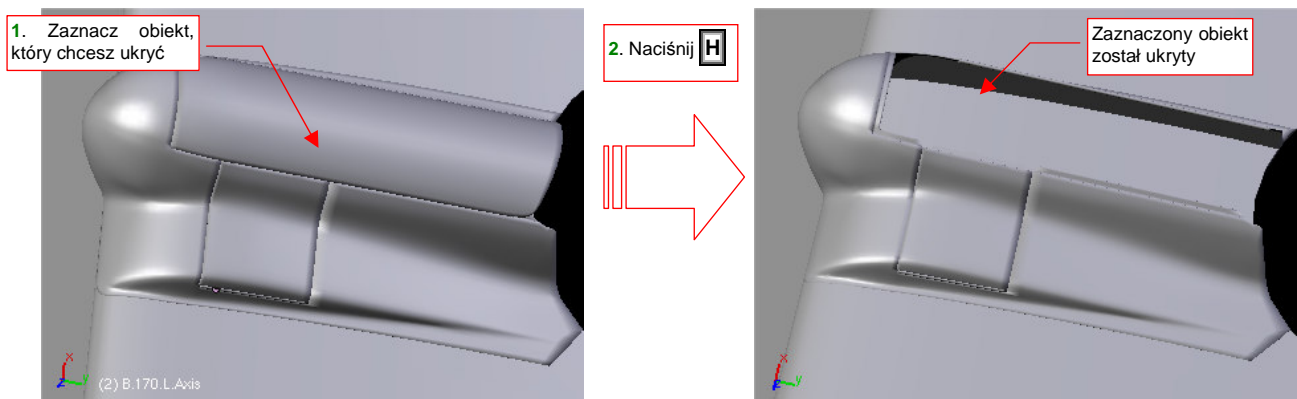
Rysunek 14.15.1 Zmiana położenia środka obiektu

Zmiana położenia środka obiektu jest "jego wewnętrzną sprawą". Nie powinna wpłynąć na jego rozmiar ani położenie w przestrzeni.

14.16 Chwilowe ukrycie obiektu (*Hide Selected*)

Czasami, podczas pracy, przydatna jest możliwość ukrycia na jakiś czas niektórych obiektów. Jest to szczególnie ważne wtedy, gdy jedna część zasłania drugą.

Aby ukryć zaznaczony obiekt (lub obiekty), po prostu naciśnij **H** (*Object*→*Show/Hide Objects*→*Hide Selected*). To sprawi, że zaznaczenie zniknie (Rysunek 14.16.1):



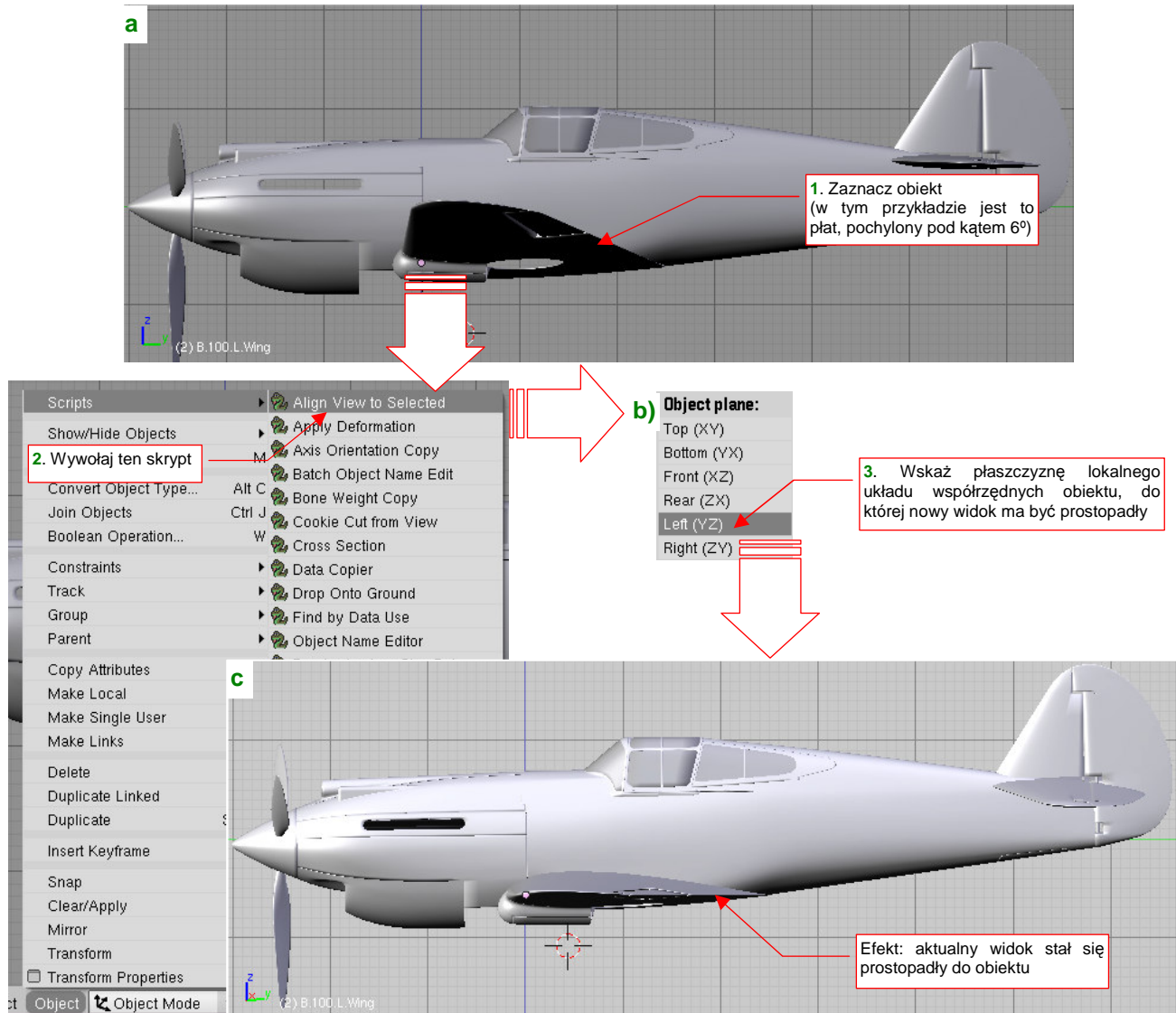
Rysunek 14.16.1 Ukrycie wybranego obiektu

Aby ukryte obiekty stały się z powrotem widoczne — naciśnij **Alt-H** (*Object*→*Show/Hide Objects*→*Show Hidden*).

14.17 Wyrównanie widoku do orientacji obiektu (*Align View*)

Podczas pracy nad modelem potrzeba czasami ustawić płaszczyznę widoku (czyli płaszczyznę, po której możemy przesuwać obiekty) prostopadłe do wybranego obiektu. W edytorze siatki takie polecenie: *View→Align View to Selected* (str. 893). W edytorze obiektów istnieje analogiczne polecenie, ale jest ograniczone: umożliwia tylko dopasowanie się do płaszczyzny **XY**. Stworzyłem więc specjalny skrypt, który umożliwi dopasowanie widoku do dowolnego z sześciu boków "pudełka", w którym jest opisany obiekt.

Aby z tego skryptu skorzystać, zaznacz obiekt, do którego chcesz się "wyrównać" (Rysunek 14.17.1a). Następnie wywołaj skrypt *Align View to Selected* (z menu *Object→Scripts*):

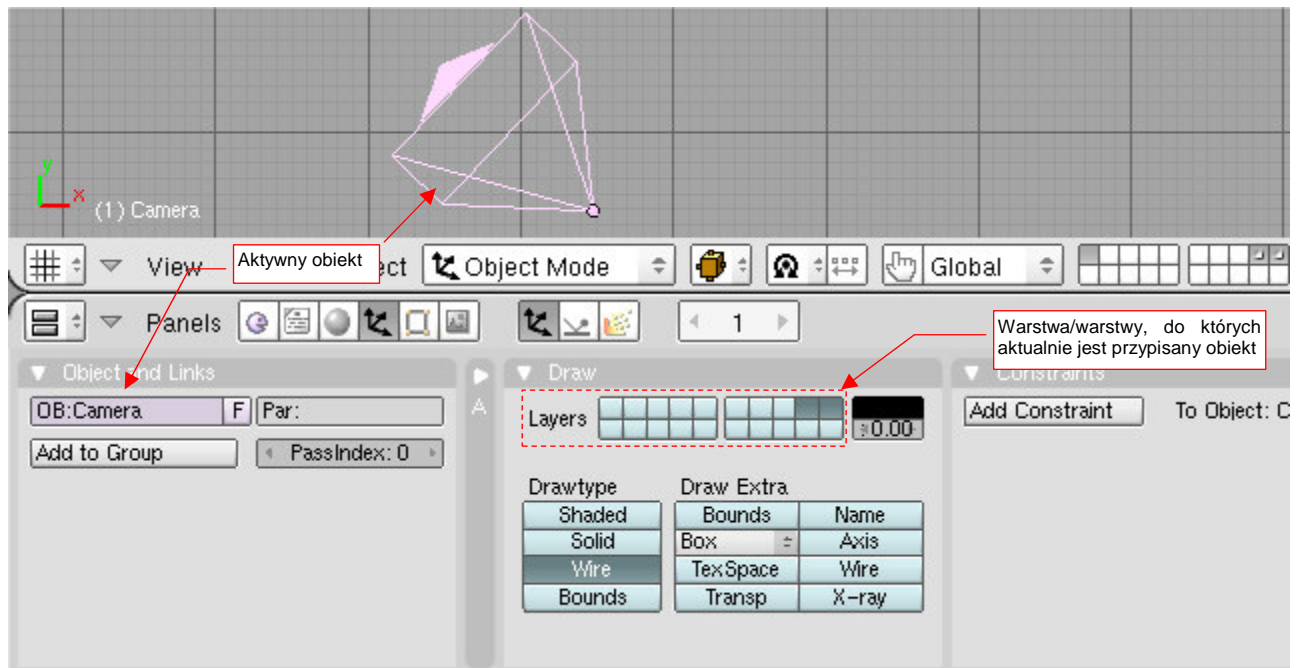


Rysunek 14.17.1 Wyrównanie aktualnego widoku do wybranego obiektu

Spowoduje to pojawienie się na ekranie okna, z którego wybierz jedną z sześciu "kardynalnych" płaszczyzn obiektu (Rysunek 14.17.1b). Efekt zobaczysz natychmiast (Rysunek 14.17.1c). W razie czego, jeżeli za pierwszym razem wybrałeś złą płaszczyznę kardynalną - wywołaj ten skrypt jeszcze raz, i wybierz właściwą.

14.18 Przepisanie obiektu do warstwy

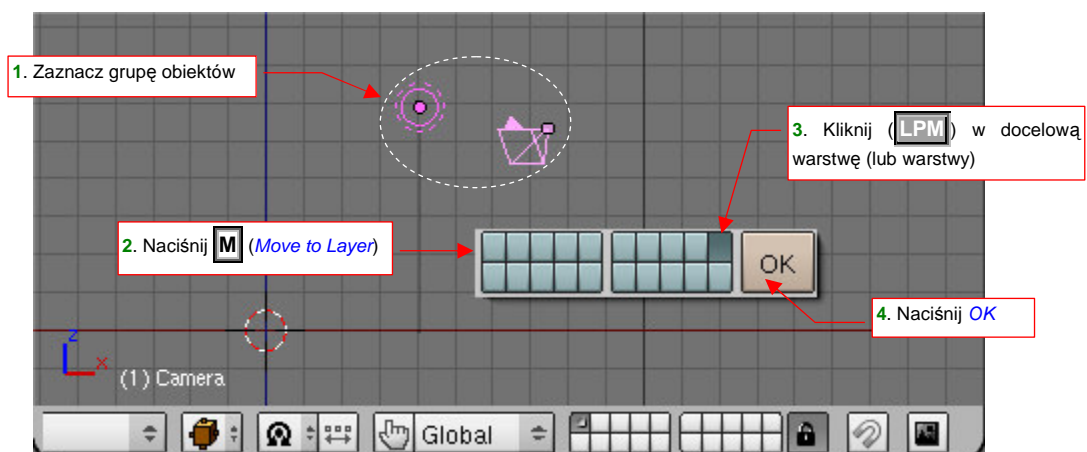
Aby sprawdzić, na jakiej warstwie jest pojedynczy obiekt: Zaznacz go, i przejdź do zestawu przycisków **Object** (**F7**). Tam, w panelu **Draw** widzisz, warstwę, do której należy wybrany element (Rysunek 14.18.1):



Rysunek 14.18.1 Warstwy, do których jest przypisany aktywny obiekt.

Klikając w kontrolki **Layers** (Rysunek 14.18.1) zmienisz przypisania obiektu do warstwy. Zwróć uwagę, że w Blenderze obiekt może być równocześnie na wielu warstwach. Jeżeli włączysz widoczność którejkolwiek z nich — obiekt także stanie się widoczny.

Aby zmienić warstwę dla wielu obiektów jednocześnie: zaznacz je i naciśnij **M** (**Object** → **Move to Layer**). Rysunek 14.18.2 pokazuje, jak to masz zrobić:

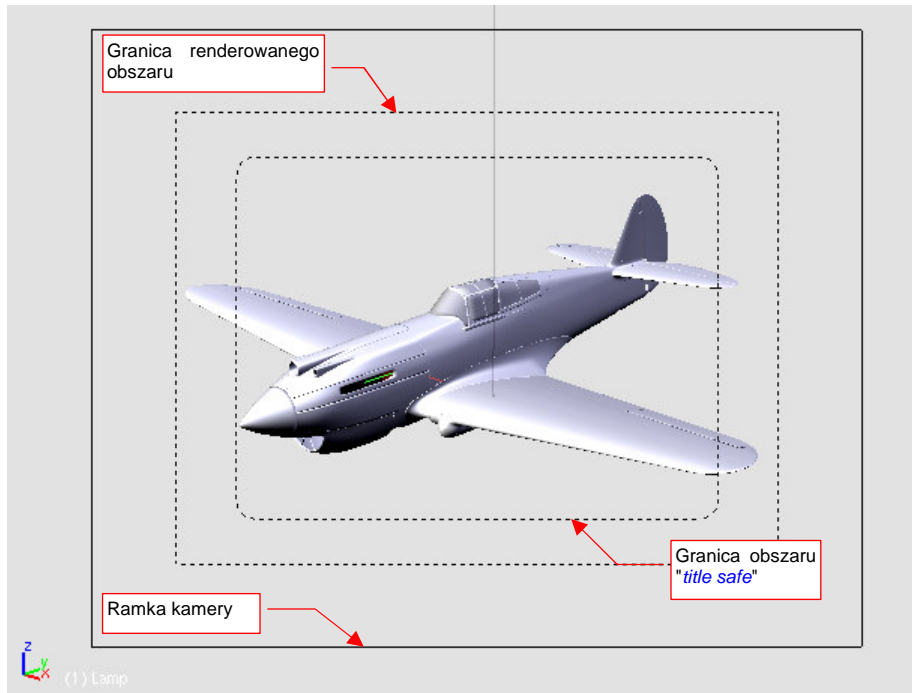


Rysunek 14.18.2 Zmiana warstwy dla grupy obiektów


Na kontrolce, pokazanej w oknie dialogowym (Rysunek 14.18.2) także możesz wskazać naraz wiele warstw. Pamiętaj, że jeżeli wybrane obiekty przypiszesz do warstwy, która jest aktualnie niewidoczna, znikną z ekranu!


14.19 Zmiana właściwości kamery (*Camera*)

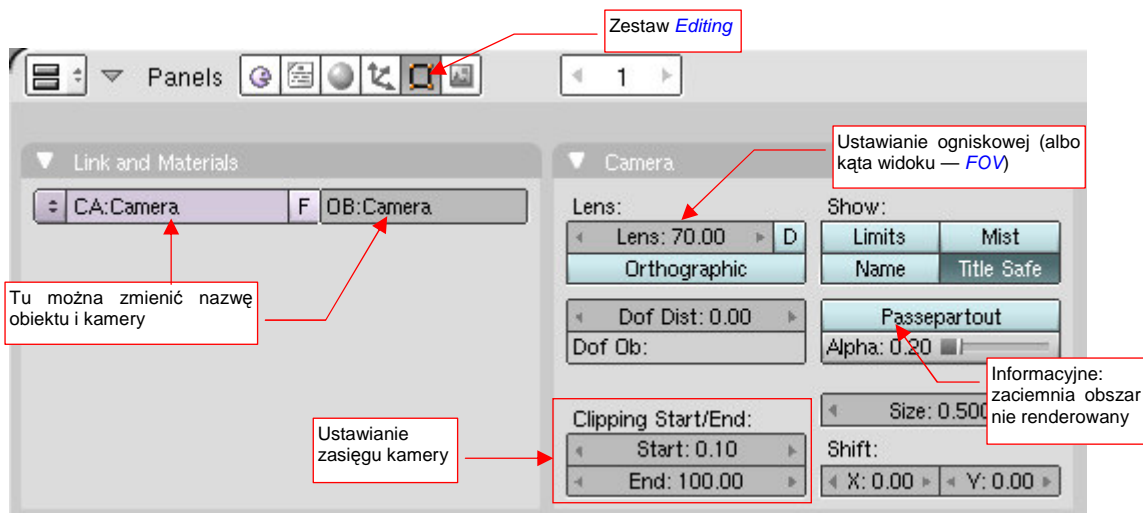
Widok z kamery (, albo **View→Cameras→Camera (Active)**) jest w oknie *3D View* pokazywany w następujący sposób (Rysunek 14.19.1):



Rysunek 14.19.1 Widok z kamery (w oknie *3D View*)

Największa ramka, narysowana linią ciągłą, to sama kamera. (Możesz np. w nią kliknąć , aby została zaznaczona). Wewnątrz niej znajduje się drugi prostokąt, narysowany linią przerywaną. To obszar, który pojawi się na renderowanym obrazie. Najmniejszy obszar — prostokąt o zaokrąglonych narożnikach — to informacja, gdzie się kończą brzegi renderowanego obrazu.

Aby dostać się do właściwości kamery, zaznacz ją, i przełącz się na zestaw przycisków *Editing*¹ () (Rysunek 14.19.2):

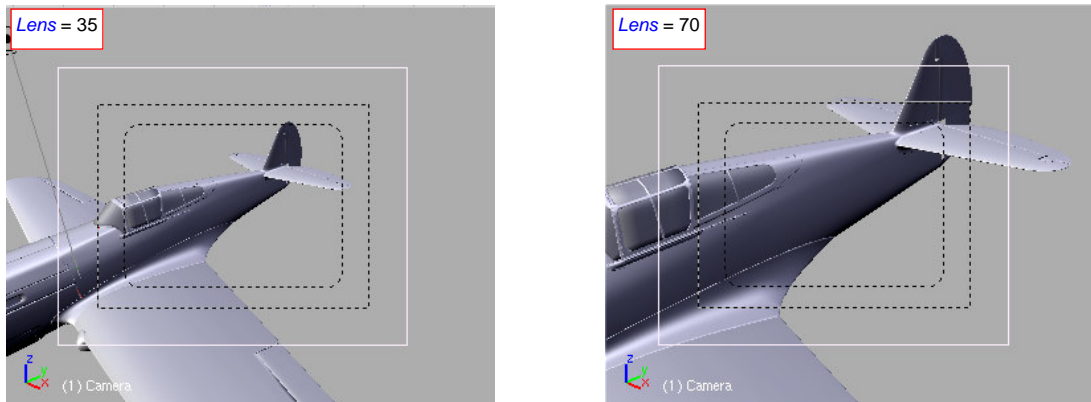


Rysunek 14.19.2 Panel właściwości kamery (zestaw *Editing*)

Rysunek 14.19.2 pokazuje istotne pola panelu *Camera*. (Pozostałych w tym modelu nie używałem).

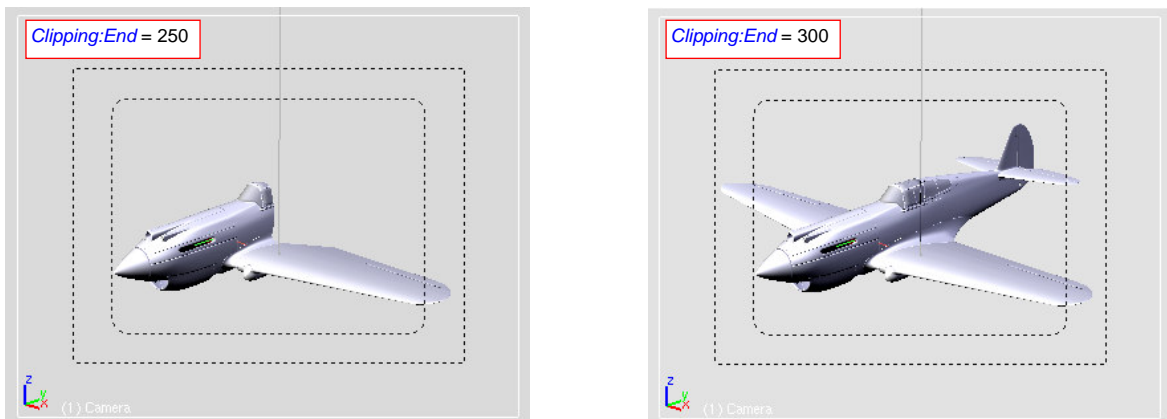
¹ Kamera to w Blenderze coś takiego jak siatka (*mesh*). To alternatywna zawartość obiektu (*datablock*, por str. 565), i dlatego jej właściwości są wyeksponowane tym samym zestawie *Editing*, służącym do zmiany zawartości obiektów.

Najbardziej oczywistym parametrem kamery jest ogniskowa (**Lens** — p. Rysunek 14.19.2). Domyślna ogniskowa kamery w Blenderze to 35mm. Odpowiada to kątowemu rozwarciu 50°. To nieco za duży kąt jak dla odległości oczu od ekranu. Często powoduje, że zniekształcenie perspektywiczne kamery wywołuje wrażenie "rybiego oka". Lepiej jest ją przestawić na większą wartość, np. 70mm (Rysunek 14.19.3):



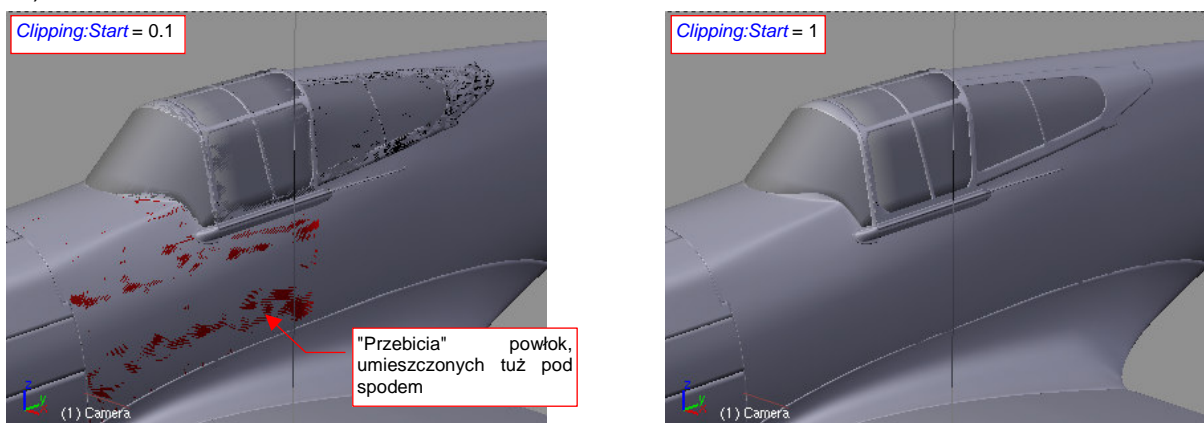
Rysunek 14.19.3 Kamera — efekt zmiany ogniskowej z 35mm do 70mm

Aby kamera nie "obciąła" zbyt odległej części modelu, wydłuż także jej zasięg. Chodzi o pola **Clipping Start/End**. Zmień pole **End** z domyślne 100 na 1000 jedn. Blendera, gdyż inaczej kamera może "obciąć" model (Rysunek 14.19.4):



Rysunek 14.19.4 "Obcięcie" modelu przez zbyt małą wartość **Clipping:End**

Warto także wydłużyć wartość **Clipping:Start** z domyślnego 0.1 do 1 jednostki, aby w widoku **3D View** blisko położone powłoki modelu nie generowały artefaktów "przebiec" jednej powierzchni przez drugą (Rysunek 14.19.5):

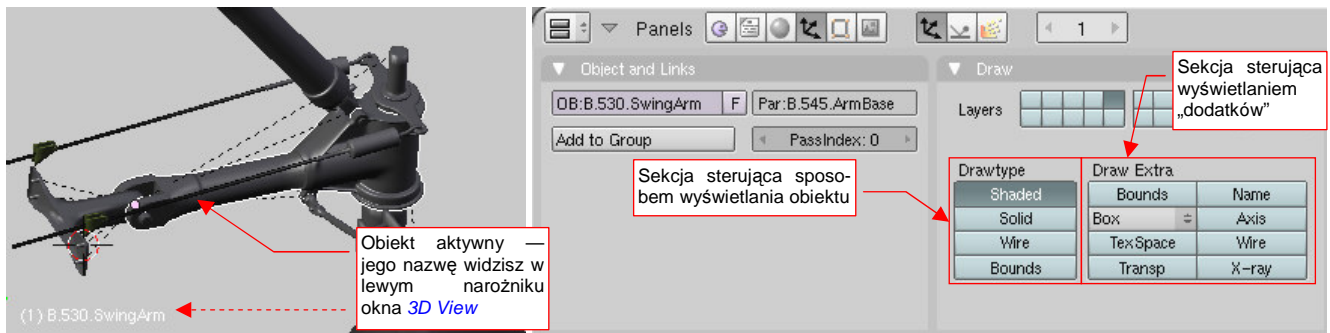


Rysunek 14.19.5 Artefakty na powierzchni modelu, powstałe przez zbyt małą wartość **Clipping:Start**

Co prawda, "przebiecia", które pokazuje Rysunek 14.19.5 nie pojawią się na ostatecznym renderze. Po co jednak mają nas drażnić podczas edycji obiektu!? Lepiej zmienić pole **Clipping:Start** i po kłopotcie!

14.20 Opcje wyświetlania obiektu

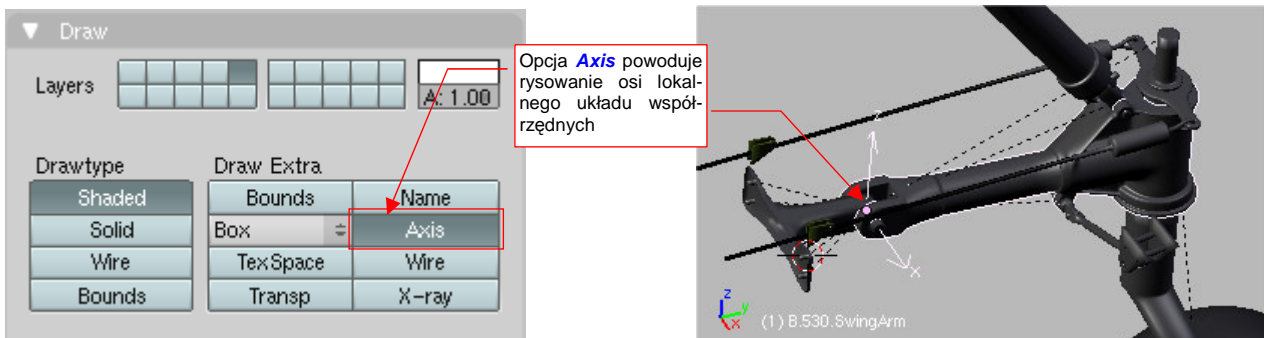
W panelu **Draw** zestawu **Object** (F7) znajdziesz opcje wyświetlania obiektu. Pamiętaj, że dotyczą zawsze obiektu aktywnego (Rysunek 14.20.1):



Rysunek 14.20.1 Panel **Draw** — opcje rysowania obiektu w oknie widoku.

W opcjach **Drawtype** zazwyczaj nic nie musisz zmieniać — pozostaw sposób wyświetlania na **Shaded**. No, może czasami, można przełączyć wyświetlanie powierzchni szyb kabiny na **Wire**. (Dzięki temu także podczas modelowania będziesz „przez nie widzieć” wewnątrz kabiny).

Z kolei najczęściej wyświetlanym przez mnie elementem dodatkowym są osie lokalnego układu współrzędnych. Ich widocznością steruje opcja **Axis**, z sekcji **Draw Extra** (Rysunek 14.20.2):



Rysunek 14.20.2 Włączenie rysowania osi lokalnego układu współrzędnych (**Axis**).

Czasami, podczas pracy z ograniczeniami, przydaje się włączyć także wyświetlanie nazw obiektów. Służy do tego opcja **Name**, z tej samej sekcji (Rysunek 14.20.3):

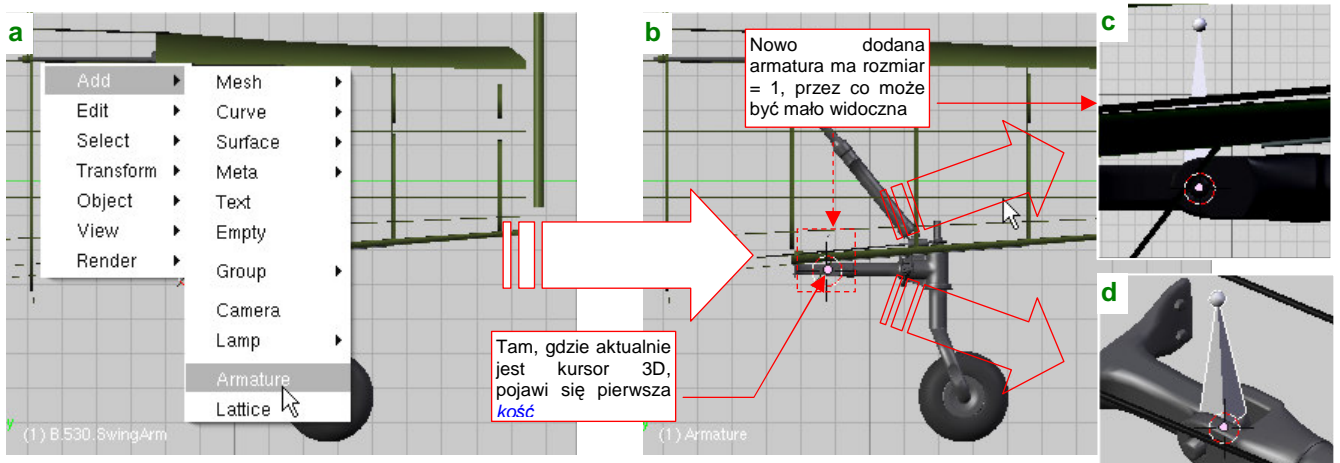


Rysunek 14.20.3 Włączenie rysowania nazwy obiektu (**Name**).

14.21 Dodanie armatury (*Armature*)

Aby „puścić w ruch” wykonany model, w grafice komputerowej używa się specjalnych szkieletów, nazywanych także armaturami (*Armature*). Skoro szkielet, to (jak skojarzyło się artystom), muszą być i kości. I tak się już utarło, że szkielety używane do animacji są złożone z „kości” (*Bones*). W tej sekcji pokażę ich zastosowanie na przykładzie animacji mechanizmu chowania kółka ogonowego (idea tego mechanizmu — zob. str. 421, Rysunek 7.1.1)

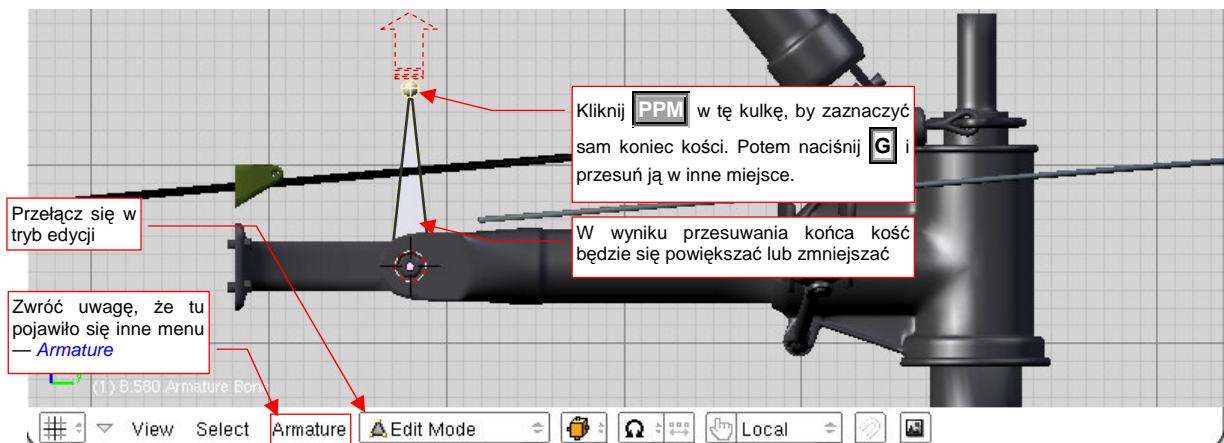
W Blenderze armaturę można dodać jak każdy inny obiekt: najpierw ustaw kursor 3D w miejscu, gdzie ma się znaleźć początek pierwszej kości. Potem wywołaj z menu **Add** polecenie *Armature* (Rysunek 14.21.1a):



Rysunek 14.21.1 Dodanie armatury

W pierwszym momencie po stworzeniu armatury, jeżeli pracujesz w mniejszym powiększeniu, możesz jej nie dostrzec (Rysunek 14.21.1b). To dlatego, że początkowo to jest jedna kość (skierowana do góry), o rozmiarze 1 jednostki (Rysunek 14.21.1c,d). Zdecydowanie odradzam zmianę skali armatury jako obiektu¹.

Zamiast tego, przełącz się w tryb edycji, i zacznij dopasowywać położenie i rozmiar kości do obiektu (Rysunek 14.21.2):

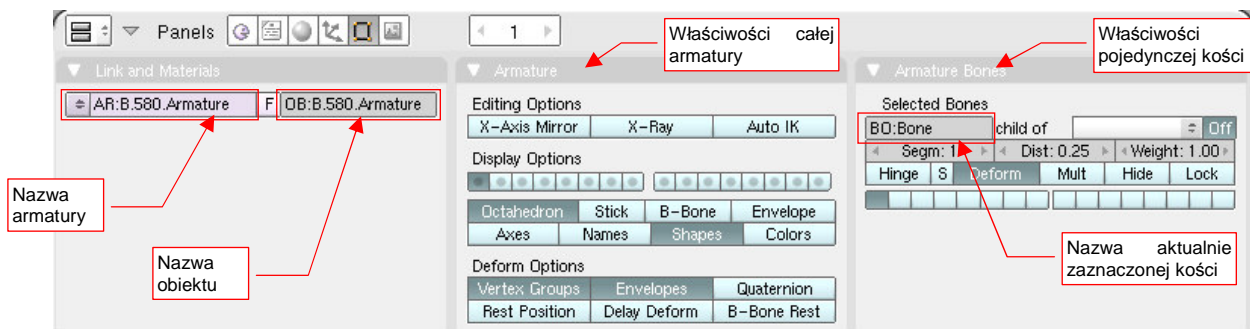


Rysunek 14.21.2 Powiększenie kości do odpowiednich rozmiarów

W trybie edycji armatury kość możesz zaznaczyć na dwa sposoby. Gdy klikniesz **PPM** w jej trzon — podświetli się cała, i możesz ją całą przesunąć w inne miejsce. Gdy klikniesz **PPM** w jej końcówkę — podświetli się tylko umieszczona na niej kula. W tym trybie przesuwasz sam koniec kości (początek pozostaje tam, gdzie jest).

¹ Gdy zmienisz skalę całej armatury, a potem przypiszesz do jej kości jakieś obiekty, to w trybie edycji zobaczysz je wszystkie odpowiednio pomniejszone. To może być trochę deprymujące (w *Object Mode* będą wszystkie wyglądały normalnie);

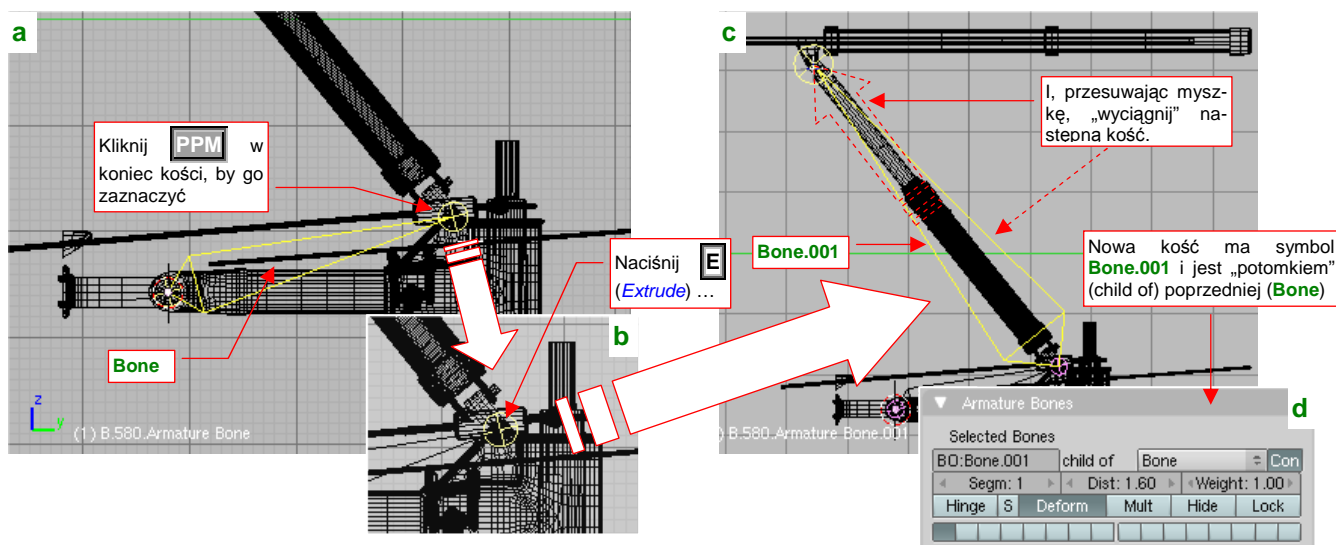
Gdy w oknie przycisków przełączysz się na zestaw *Editing* (**F9**), to zobaczysz dwie nowe panele: *Armature* (z ogólnymi ustawieniami aktualnej armatury) i *Armature Bones* — z ustawieniami aktualnie wybranej kości (Rysunek 14.21.3):



Rysunek 14.21.3 Właściwości obiektu *Armature* i jego kości (*Bone*)

Domyślnie pierwsza kość nosi nazwę **Bone**, a następne — **Bone.001**, **Bone.002**, itd. Oczywiście, można je nazwać inaczej, adekwatnie do konkretnej roli, którą pełnią (np. **ramię**, **przedramię** czy coś w tym stylu). Nazwy kości muszą być unikalne tylko w obrębie pojedynczej armatury. (W innej armaturze możesz użyć takich samych nazw). Tu także (Rysunek 14.21.3), w panelu *Link and Materials*, możesz nadać całej armaturze i obiektowi, do którego jest przypisana, odpowiednie nazwy (najlepiej takie same).

Aby dodać do armatury nową kość, która łączy się z poprzednią, należy użyć lokalnej odmiany operacji wytłaczania (*Extrude*). Kliknij najpierw **PPM** w koniec kości — punkt, w którym ma się zaczynać następna kość (Rysunek 14.21.4a):



Rysunek 14.21.4 Wytłoczenie (*Extrude*) kolejnej kości

W odpowiedzi Blender powinien podświetlić samą końcówkę kości (Rysunek 14.21.4b). (Jeżeli zaznaczył całą — kliknij w **PPM** w to samo miejsce jeszcze raz. Cała kość i końcówka są wybierane naprzemiennie). Następnie naciśnij przycisk **E** (lub wybierz z menu *Armature*→*Extrude*). W tym momencie do armatury zostanie dodana nowa kość, której koniec możesz przesunąć myszką w odpowiednie miejsce (Rysunek 14.21.4c). Gdy klikniesz **LPM** — kość zostanie ostatecznie dodana. Gdybyś chciał z niej jednak zrezygnować — w trakcie przesuwania wystarczy nacisnąć **Esc**.

Nowa kość otrzymuje automatycznie taką samą nazwę, jak kość z której powstała, z dodaną numeryczną końcówką (np. z **Bone** wytłoczysz **Bone.001**, a z **Bone.001** — **Bone.002**). Blender przypisuje ją także do poprzedniej kości jako „rodzica” (zobacz pole *child of* na panelu *Armature Bones* — Rysunek 14.21.4d).

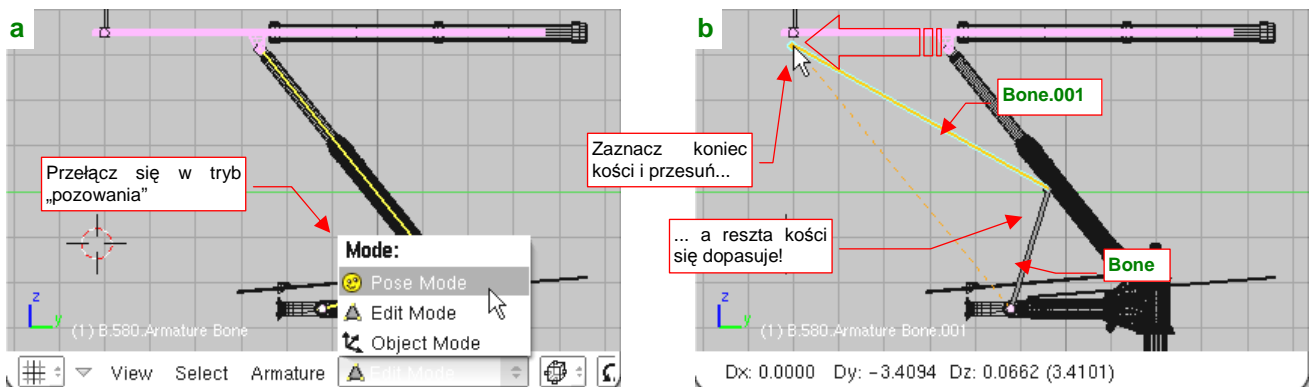
„Rozpięliśmy” już dwie kości, odpowiadające wahaczowi i amortyzatorowi kółka ogonowego. Inna sprawa, że są rysowane przez Blender jako dość duże ośmioboki (*Octahedron*). Na szczęście w panelu *Armature* masz do wyboru kilka innych sposobów reprezentacji kości. Przełącz się na przykład na „pręty” (*Stick* — Rysunek 14.21.5):



Rysunek 14.21.5 Dostosowanie ogólnych ustawień armatury

Przy okazji włącz w panelu *Armature* opcję *Auto IK* (Rysunek 14.21.5a). To spowoduje, że cały układ kości zacznie się zachowywać jak prawdziwy zespół powiązanych przegubami prętów¹.

Zobaczymy, jak to wygląda w działaniu. Przełącz się w specjalny tryb „pozowania” — *Pose Mode*. Wystarczy wybrać *Pose Mode* z listy trybów pracy w nagłówku okna widoku 3D (Rysunek 14.21.6a):



Rysunek 14.21.6 „Pozowanie” armatury

- Zawartość listy trybów pracy zależy od rodzaju aktywnego obiektu. Tryb *Pose Mode* pojawia się na niej tylko wtedy, gdy aktywnym obiektem jest jakaś armatura.

W trybie pozowania można eksperymentować, ustawiając armaturę na najróżniejsze sposoby. Gdy przełączysz się z powrotem w tryb edycji, wszystko powróci do pozycji początkowej (*Rest position*).

Spróbujmy poruszać naszą armaturą. W *Pose Mode* kliknij **PPM** w kość **Bone.001**, aby ją zaznaczyć (powinna się podświetlić na niebiesko). Następnie naciśnij **G** (*Pose* → *Transform* → *Grab*) i przesunij koniec zaznaczonej kości w nowe miejsce (Rysunek 14.21.6b). Czy widzisz? Obydwie kości poruszają się całkiem realistycznie, jak dwa połączone przegubowo segmenty.

¹ *IK* to skrót od angielskiego *inverse kinematics*. W języku polskim nosi nazwę „kinematyki odwrotnej” i pochodzi z automatyki/robotyki. Zazwyczaj chodziło o ramię robota, złożone z kilku ruchomych segmentów. „Zwykła” kinematyka pozwalała rozwiązać prostsze zagadnienie: „jeżeli pierwszy segment obrócimy o 30°, a drugi względem pierwszego o 20°, to gdzie się znajdzie koniec ramienia?”. Kinematyka odwrotna znajduje odpowiedź na pytanie postawione bardziej praktycznie: „o jaki kąt należy odwrócić każdy z segmentów, by koniec ramienia znalazł się tam, gdzie chcemy?”

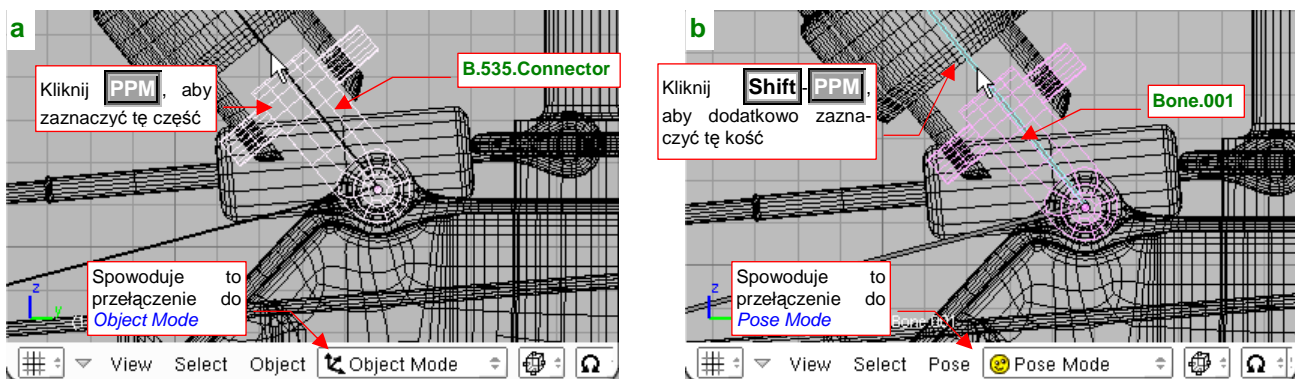
Teraz wystarczy przyłączyć do tych kości odpowiednie obiekty. Upewnij się, że jesteś w *Pose Mode*.

- Aby zaznaczyć obiekt i kość tak, jak to jest pokazane poniżej, musisz zacząć zaznaczanie w trybie *Pose Mode*. Inaczej nie uda Ci się wskazać konkretnej kości.

Jeżeli chodzi o amortyzator, to w hierarchii obiektów jego „rodzicem” powinien być niewielki **B.355.Connector**. Zaznacz go (**PPM**).

- Jeżeli w *Pose Mode*, mimo klikania **PPM**, Blender nie chce obiektu podświetlić, to oznacza to, że masz zaznaczoną jakąś kość. Wyłącz ją (klawisz **A** — *Select/Deselct All*) i spróbuj jeszcze raz.

Zwróć uwagę, że to zaznaczenie przełączyło tryb pracy w *Object Mode* (Rysunek 14.21.7a):

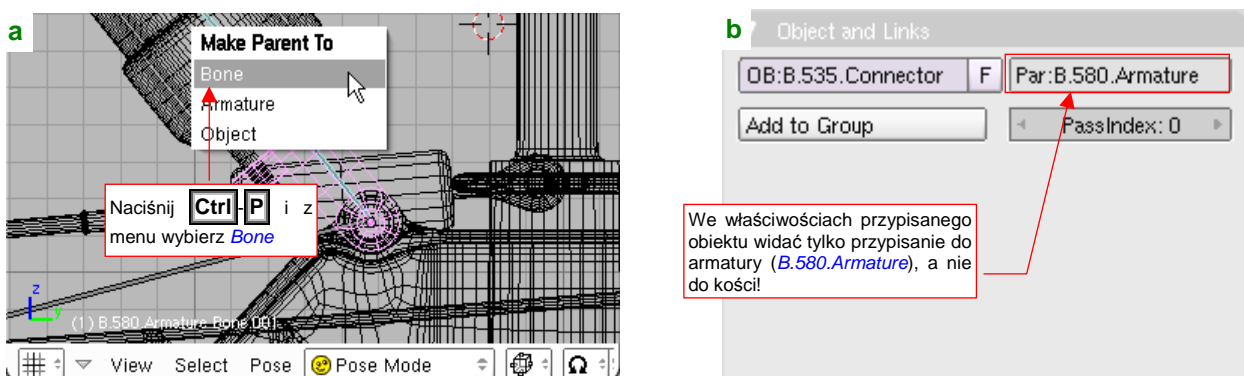


Rysunek 14.21.7 Zaznaczenie początku amortyzatora i kości (Bone.001)

Do złączki **B.535.Connector** powinny być hierarchicznie przypisane (relacją „parent - child”): cylinder (**B.540.Damper**) i tłok (**B.550.Piston**). W przypadku obiektów przyłączonych do kości, bardzo ważny jest środek. Najlepiej, aby się znajdował w tym samym miejscu, co początek kości. Zwróć uwagę, że właśnie tak jest zlokalizowany środek obiektu **B.535.Connector**.

Teraz dodaj do wybranych elementów odpowiednią kość — w tym przypadku to **Bone.001**. Kliknij w nią **PPM**, trzymając wciśnięty klawisz **Shift**. Spowoduje to przełączenie Blendera w *Pose Mode*. Powinieneś w tym momencie widzieć: obiekt, który chcesz przypisać do kości w kolorze bladnoróżowym („pozostałe zaznaczenie”) i kość w kolorze niebieskim (Rysunek 14.21.7b).

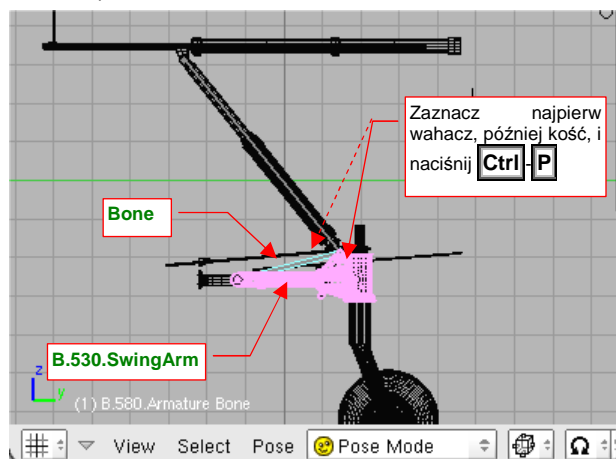
Gdy dla tak wskazanych obiektów naciśniesz **Ctrl-P**, pojawi się menu *Make Parent To*, z którego należy wybrać opcję *Bone* (Rysunek 14.21.8a):



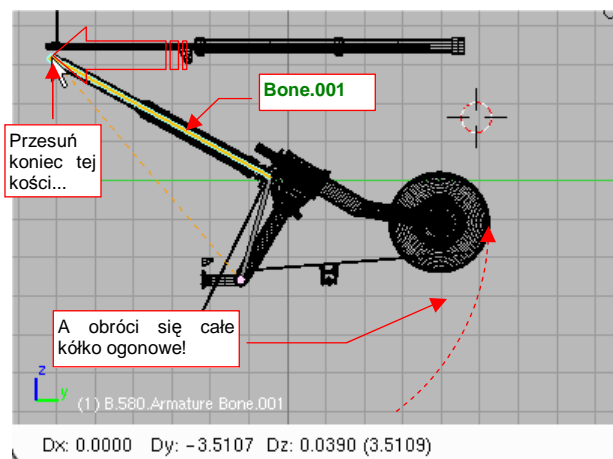
Rysunek 14.21.8 Przypisanie amortyzatora do kości (Bone.001)

To wszystko — złączka amortyzatora jest przypisana do kości (a wraz z nią — wszystkie jej obiekty „potomne”: cylinder i tłok, czyli cały amortyzator). Co prawda we właściwościach obiektu **B.535.Connector** wygląda, jak gdyby jego „rodzicem” była cała armatura (Rysunek 14.21.8b), ale to powiązanie jest bardziej wysublimowane.

W ten sam sposób zrób pierwszą kość armatury — **Bone** — rodzicem wahacza (**B.530.SwingArm**) (Rysunek 14.21.9):



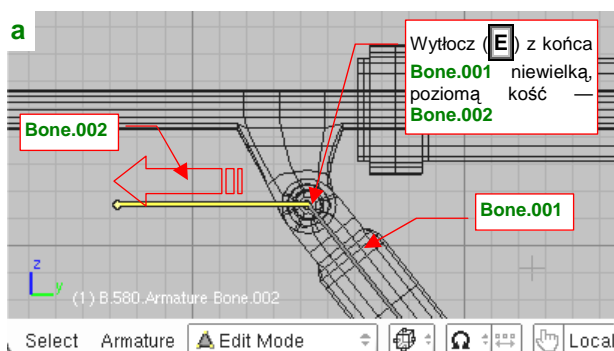
Rysunek 14.21.9 Przypisanie wahacza do kości **Bone**



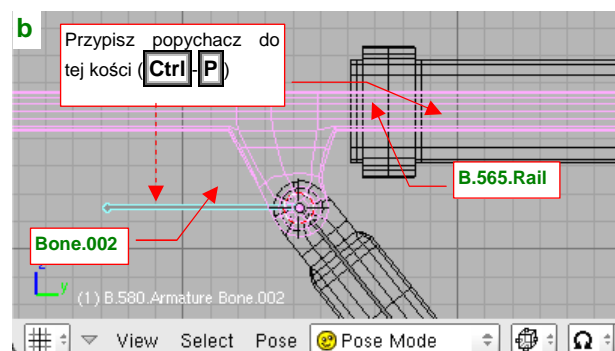
Rysunek 14.21.10 Rezultat — składające się kółko ogonowe

Zwróć uwagę, że środek wahacza znajduje się tam, gdzie początek kości **Bone** — w osi, do której wahacz jest przyczepiony. Teraz wystarczy w *Pose Mode* powtórnie przesunąć poziomo (**G**, **Y**) koniec kości amortyzatora (**Bone.001**) — i cały zespół kółka ogonowego składa się i rozkłada jak pierwowzór (Rysunek 14.21.10)!

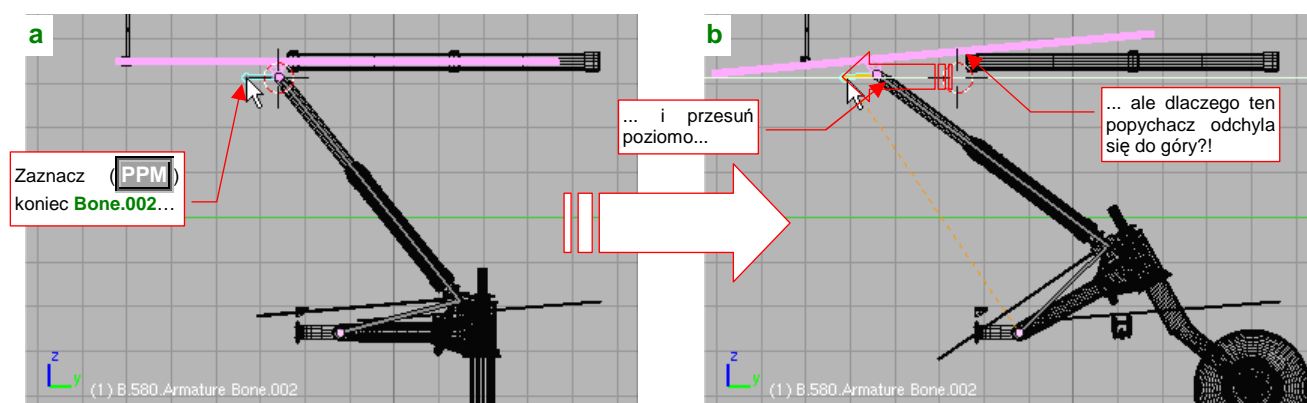
Skoro nam tak dobrze idzie, włączmy w ten mechanizm popychacz (**B.565.Rail**). W prawdziwym samolocie ten element przesunął koniec amortyzatora do tyłu i do przodu, wymuszając otwieranie i zamykanie podwozia. Wytłocz z końca **Bone.001** kolejną, poziomą kość (**Bone.002**) (Rysunek 14.21.11a):



Rysunek 14.21.11 Wytłoczenie trzeciej kości (**Bone.002**) i przypisanie jej do popychacza



Przypisz do kości **Bone.002** popychacz **B.565.Rail** (Rysunek 14.21.11b). Teraz wystarczy zaznaczyć tę ostatnią kość (Rysunek 14.21.12a) i przesunąć:

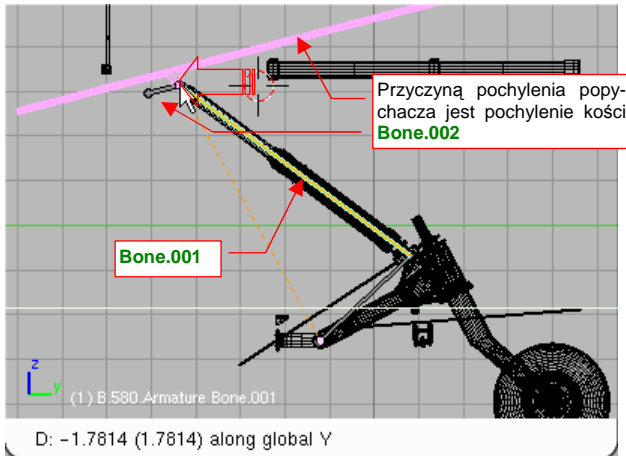


Rysunek 14.21.12 Niespodziany rezultat poziomego przesunięcia kości **Bone.002**

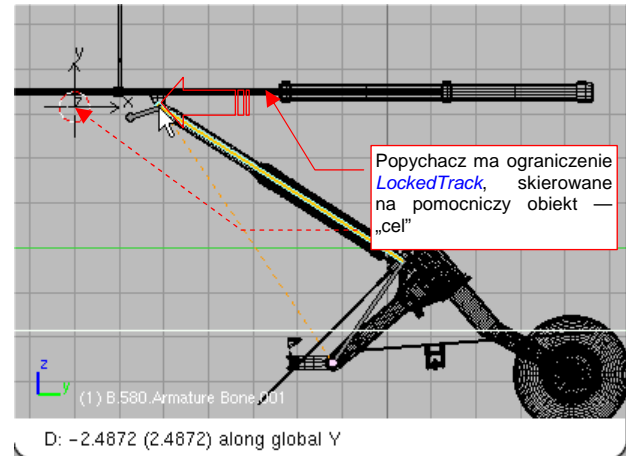
Ale zaraz, dlaczego w miarę, jak przesuwasz koniec **Bone.002** w lewo, cały popychacz coraz bardziej odchyła się do góry (Rysunek 14.21.12b)!!?

Jeżeli podczas przesuwania przyjrzyj się dokładniej zachowaniu kości **Bone.002**, zauważysz że w miarę oddalania się od „pozycji spoczynku” coraz bardziej odchyła się do dołu. A skoro kość się odchyła, to robi to także jej obiekt potomny — popychacz.

W takim razie, może jeżeli przesuwając będziemy koniec poprzedniej kości — amortyzatora (**Bone.001**) — to następna (**Bone.002**) podczas ruchu zachowa poziome położenie? Spróbujmy (Rysunek 14.21.13):



Rysunek 14.21.13 Podobny efekt dla kości **Bone.001**



Rysunek 14.21.14 Rozwiązanie problemu — dodanie do popychacza ograniczenia **Locked Track**.

Niestety, nic z tego. Pozostawiona „samej sobie” swobodna kość **Bone.003** odchyła się jeszcze bardziej niż poprzednio. Takie są, niestety, efekty działania kinematyki odwrotnej (*Auto IK*).

Ten problem można rozwiązać za pomocą ograniczenia:

- wstaw na linii ruchu środka popychacza (osi obrotu tłoka amortyzatora) pomocniczy, pusty obiekt (**Add→Empty** — szczegóły na str. 791). Ten obiekt będzie pełnić rolę „celu”;
- dodaj do popychacza ograniczenie **Locked Track** (szczegóły — zob. str. 823). W parametrach ograniczenia lokalna oś **Z** obiektu na śledzić umieszczony z przodu „cel”, a obrót może następować wyłącznie wokół lokalnej osi **X**;

Rysunek 14.21.14 przedstawia rezultat tego rozwiązania. Popychacz ignoruje, dzięki dodanemu ograniczeniu, poprzeczne obroty kości **Bone.002**. Gdy przesuwasz poziomo koniec kości **Bone.001**, cały zespół prawidłowo składa się i rozkłada.

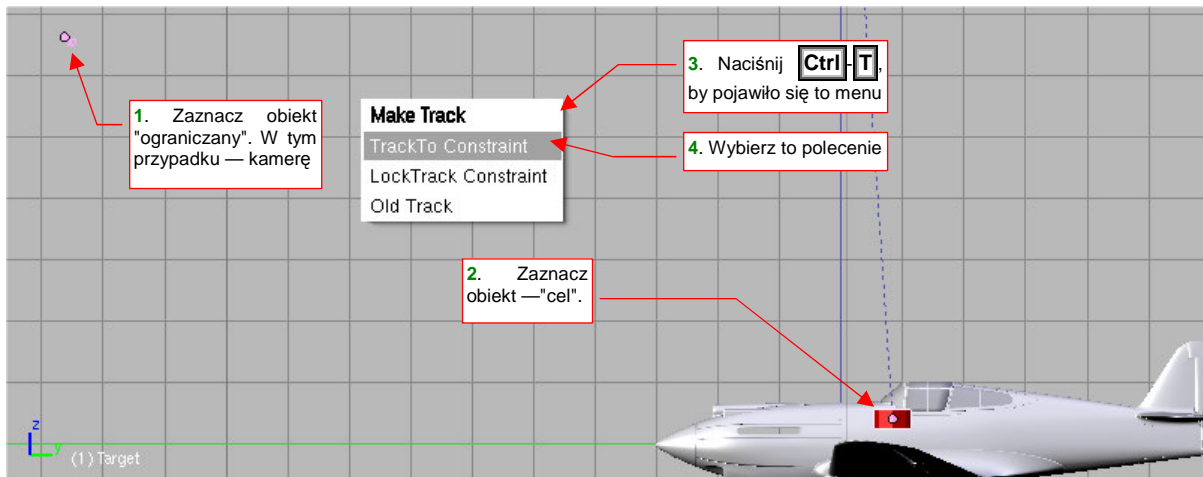
14.22 Przepisanie ograniczenia *Track To*

Obiekt z ograniczeniem *Track To* jest zawsze zorientowany w przestrzeni tak, by jego oś była skierowana na środek wskazanego celu (innego obiektu). Domyślnie w cel kierowana jest lokalna oś **Z**, ale oczywiście we właściwościach ograniczenia można wskazać inną (**X** lub **Y**).

Najszybsza metoda utworzenia ograniczenia *Track To* to:

- zaznaczenie obiektu/obiektów podlegających ograniczeniu;
- zaznaczenie obiektu - celu (to musi być ostatni zaznaczony — tzn. to musi być obiekt aktywny!);
- naciśnięcie skrótu **Ctrl-T** i wybranie z podręcznego menu polecenia *Track To Constraint*:

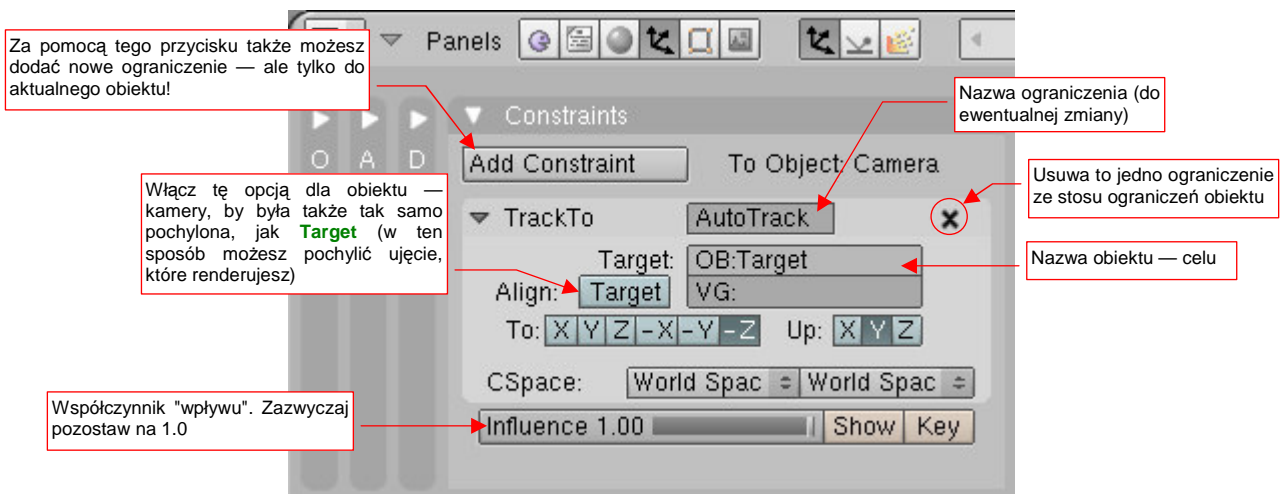
Rysunek 14.22.1 pokazuje, jak to zazwyczaj wygląda:



Rysunek 14.22.1 Dodanie ograniczenia *Track To* za pomocą skrótu *Make Track*

Bardziej ogólna metoda, pozwalająca dodać każde ograniczenie, polega na zaznaczeniu obiektów "ograniczanych" oraz "celu", i wywołaniu polecenia *Object→Constraints→Add Constraint...* (**Ctrl-Alt-C**). Lista rozwijalna, która się wówczas pojawi, zawiera wszystkie typy ograniczeń.

Właściwości ograniczenia *Track To* wybranego obiektu można zmienić w zestawie *Object* (**F8**), panelu *Constraints* (Rysunek 14.22.2):



Rysunek 14.22.2 Właściwości ograniczenia *Track To*

W przypadku tworzenia powiązania kamery z celem, nie zapomnij włączyć przełącznika *Target*.

- Najprostszą drogą do usunięcia ograniczenia **Track To** jest użycie skrótu **Alt-T**. W menu, które wówczas się pojawi, warto wybrać opcję **Clear Track and Keep Transform**. Po jej wybraniu obiekt/obiekty zostaną "uwolnione", ale nie zmienią swojego położenia ani orientacji w przestrzeni.

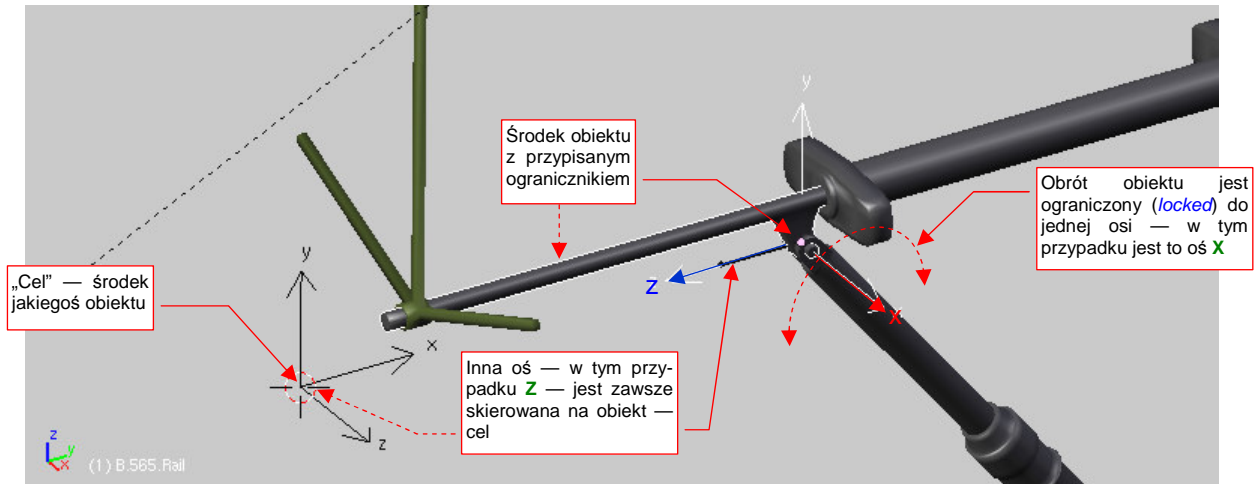
Innymi drogami do usunięcia ograniczenia są:

- wywołanie polecenia **Object→Constraints→Clear Constraints**;
- usunięcie pojedynczego ograniczenia ze stosu ograniczeń aktualnego obiektu (Rysunek 14.22.2).

Te metody nie gwarantują jednak zachowania aktualnej transformacji obiektu, jak to ma miejsce przy użyciu skrótu **Alt-T**.

14.23 Przypisanie ograniczenia *Locked Track*

Ograniczenie *Locked Track* działa podobnie do *Track To* (por. str. 814). Także tutaj obiekt obraca się zawsze tak, by wskazana w parametrach ograniczenia oś lokalnego układu współrzędnych była skierowana na środek wskazanego celu. Jednak w tym przypadku obiekt może się obracać tylko wokół jednej osi (Rysunek 14.23.1):

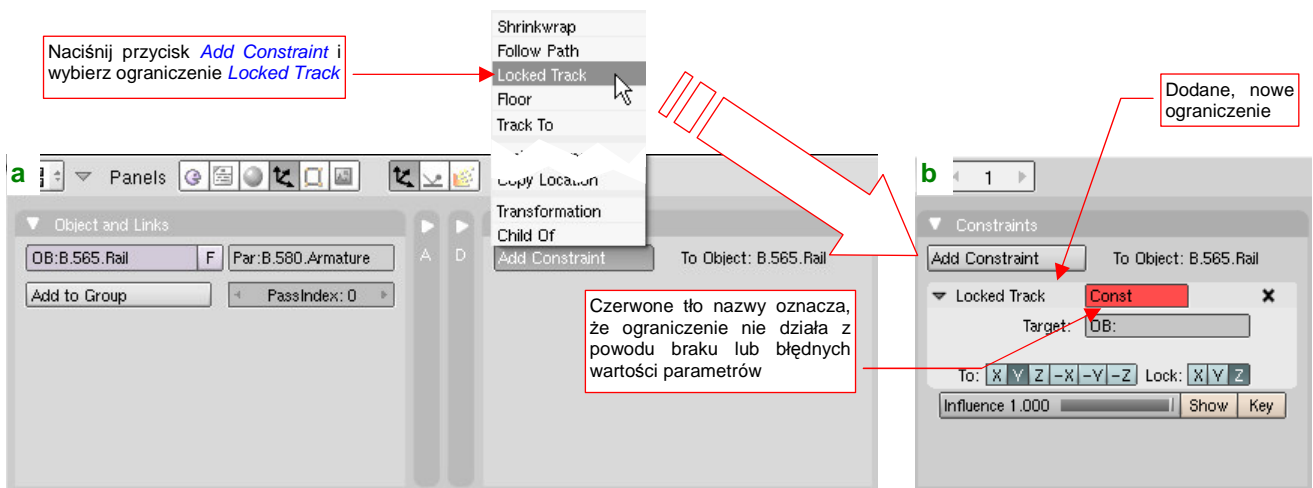


Rysunek 14.23.1 Zasada działania ograniczenia *Locked Track*.

Oznacza to, że wybrana oś może nie być dokładnie skierowana na cel, (bo np. cel jest przesunięty w bok, poza płaszczyznę obrotu). Tym niemniej obiekt z ograniczeniem *Locked Track* zawsze będzie się „starał wycelować” ją jak najlepiej — na tyle, na ile pozwala mu podana w parametrach oś obrotu.

- Ograniczenia typu *Locked Track* doskonale nadają się do modelowania mechanizmów. Prawie każdy element takich zespołów ma „jeden stopień swobody” — może się obracać wokół jednej osi.

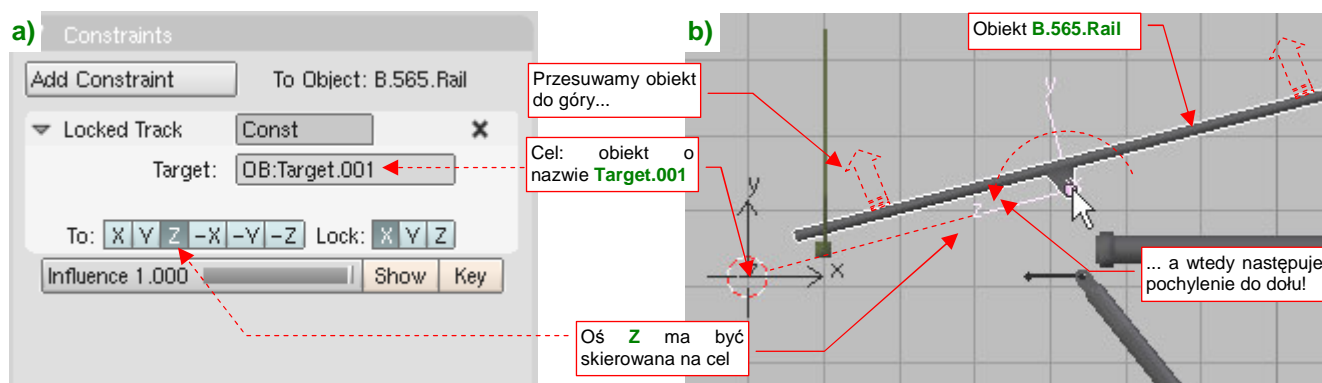
Aby dodać ograniczenie *Locked Track*, zaznacz odpowiedni obiekt, i w zestawie *Object Buttons*, panelu *Constraints*, naciśnij przycisk *Add Constraint* (Rysunek 14.23.2a):



Rysunek 14.23.2 Dodanie do obiektu ograniczenia *Locked Track*.

Z listy rozwijalnej, która się pojawi ponad przyciskiem, wybierz opcję *Locked Track*. Blender wstawi to ograniczenie na koniec listy („stosu”) ograniczeń przypisanych do obiektu. W pierwszej chwili pole *Locked Track* w panelu ograniczenia, (to z nazwą), jest czerwone (Rysunek 14.23.2b). To oznacza, że coś z parametrami tego ograniczenia jest nie tak. W tym przypadku w polu *Target* nie wpisaliśmy jeszcze nazwy obiektu — „celu”. Gdy to uzupełnimy, czerwony kolor zniknie z panelu.

Zróbmy to: wpisz w pole **Target** nazwę obiektu **Target.001** (którego używam wyłącznie w celach szkoleniowych). Oprócz tego na liście **To** zaznacz oś **Z**, a na **Lock** — **X** (Rysunek 14.23.3a):

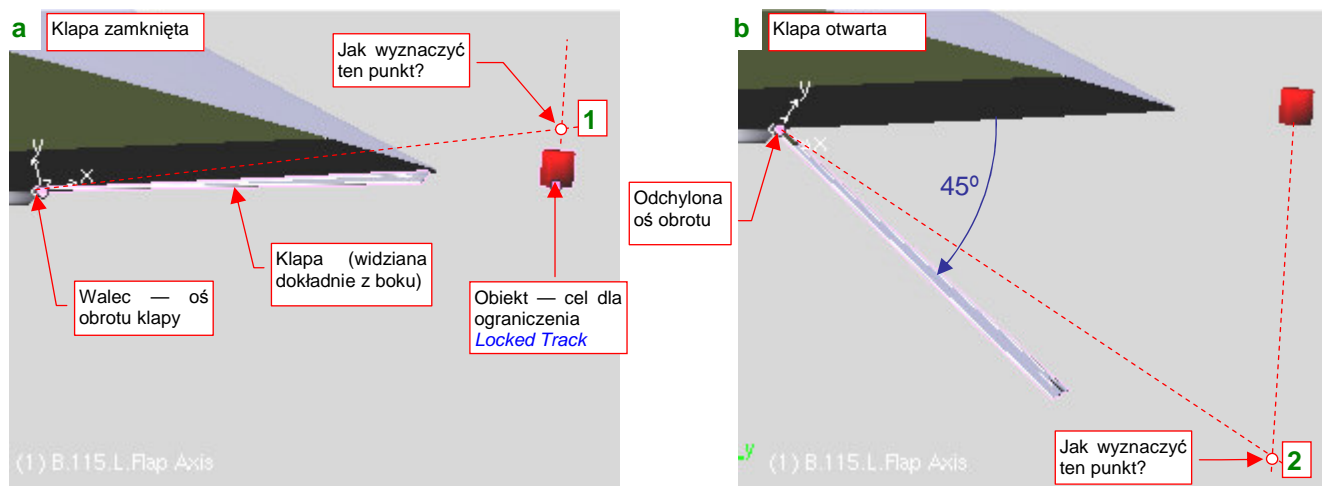


Rysunek 14.23.3 Przykład przypisania parametrów **Locked Track**.

Gdy teraz spróbujesz przesunąć ograniczony obiekt do góry — zobaczysz że zacznie się obracać tak, by jego oś **Z** nadal „celowała” w środek obiektu **Target.001** (Rysunek 14.23.3b). Gdybyś zaczął przesuwając ten obiekt **B.565.Rail** wzdłuż lokalnej osi **X**, nie spowodowałoby to żadnego dodatkowego obrotu (tym właśnie się różni to ograniczenie od ograniczenia **Track To**).

Ograniczenie **Locked Track** doskonale się nadaje do sterowania ruchem sterów, lotek lub klap samolotu. Zazwyczaj przypisuje się je do ich osi obrotu (zamodelowanych jako oddzielny obiekt — por. str. 133). Często jednak te osie nie są ustawione równoległe do globalnego układu współrzędnych. (Tak bywa, gdyż w czasie formowania samolotu nie zawsze byłes w stanie o to odpowiednio zadbać). Pojawia się wówczas problem: jak umieścić obiekt — „cel” ograniczenia **Locked Track** we właściwym położeniu?

Przykładem takiego problemu jest kłapa skrzydła, którą pokazuje Rysunek 14.23.4:

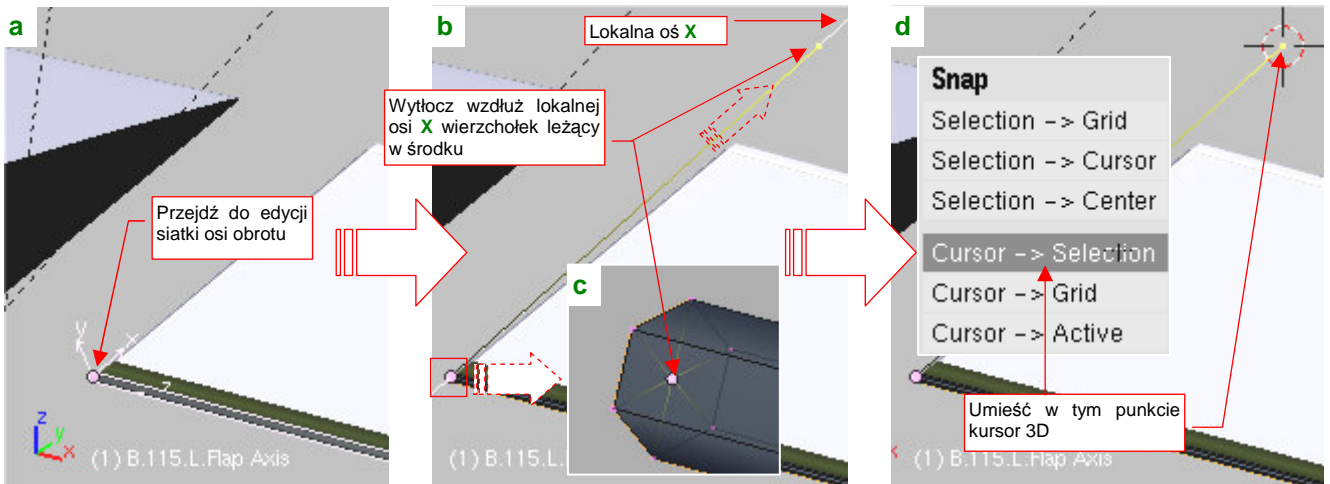


Rysunek 14.23.4 Problem: określenie dokładnego położenia „celu” ograniczenia **Locked Track**

Kłapa to z grubsza prostokątny fragment powierzchni. Jest przymocowana (relacją **Parent**) do długiego walca, imitującego jej zawiasy (por. str. 259). Kłapę należy „zmechanizować”: przypiszemy osi **X** zawiasów ograniczenie **Locked Track**, śledzące specjalnie w tym celu przygotowany obiekt — „cel” (Rysunek 14.23.4a). Obrót będzie się odbywał wokół lokalnej osi **Z**. Dzięki temu użytkownik będzie mógł sterować kłapą za pomocą łatwo dostępnego uchwytu („celu”), a nie ukrytej głęboko osi, którą trudno jest wskazać myszką.

Niestety, lokalna oś **X** zawiasów kłapy jest pochylona w stosunku do globalnego układu współrzędnych, w którym umieściliśmy obiekt — „cel”. Jak ustawić ten „uchwyt” dokładnie w punkcie **1** (Rysunek 14.23.4a), aby po włączeniu ograniczenia zamknięta kłapa się nie „poruszyła”? Jak wyznaczyć drugie położenie „uchwytu”, odpowiadające całkowicie wychylonej (w tym przypadku — o kąt 45°) kłapie (punkt **2**, Rysunek 14.23.4b)?

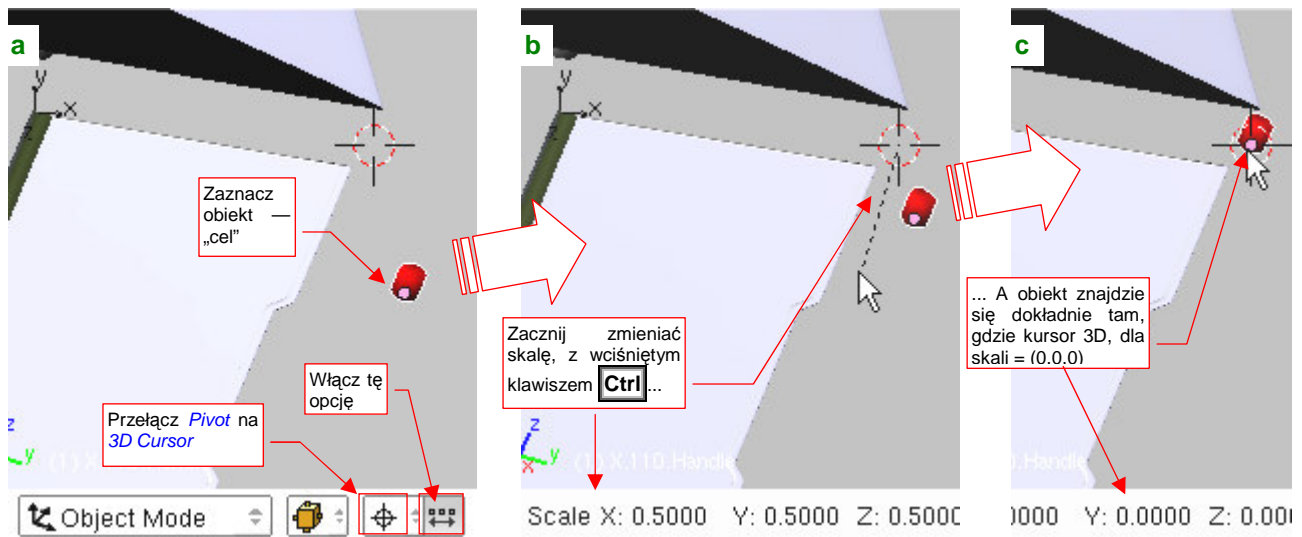
Aby wyznaczyć górne położenie „celu”, przejdź do edycji siatki zawiasów (Rysunek 14.23.5a):



Rysunek 14.23.5 Umieszczenie kursora 3D w górnym położeniu „celu”

W tym przypadku tak się szczęśliwie składa, że siatka ma wierzchołek w tym samym punkcie, co środek (punkt odniesienia) tego obiektu (Rysunek 14.23.5c). Zaznacz go, i wytłocz z niego nową krawędź wzdłuż lokalnej osi X (**E**, **X**, **X**). Odsuń jej koniec na jakąś większą odległość (Rysunek 14.23.5b). Powinna być na tyle duża, by końcowy wierzchołek znalazł się znacznie poza powierzchnią płata. Potem wystarczy wywołać polecenie, które umieści w tym punkcie kursor 3D: **Shift-S** (**Snap**, **Cursor**→**Selection**) (Rysunek 14.23.5d).

Umieściliśmy już w odpowiednim miejscu „punkt zaczepienia” — kursor 3D. Mógłbyś teraz odczytać jego współrzędne i wpisać je jako nowe położenie uchwytu. Znam jednak prostszą metodę, nie wymagającą przepisowywania, które zajmuje czas i podczas którego można się pomylić. To przesuwanie obiektów za pomocą zmiany skali (Rysunek 14.23.6):

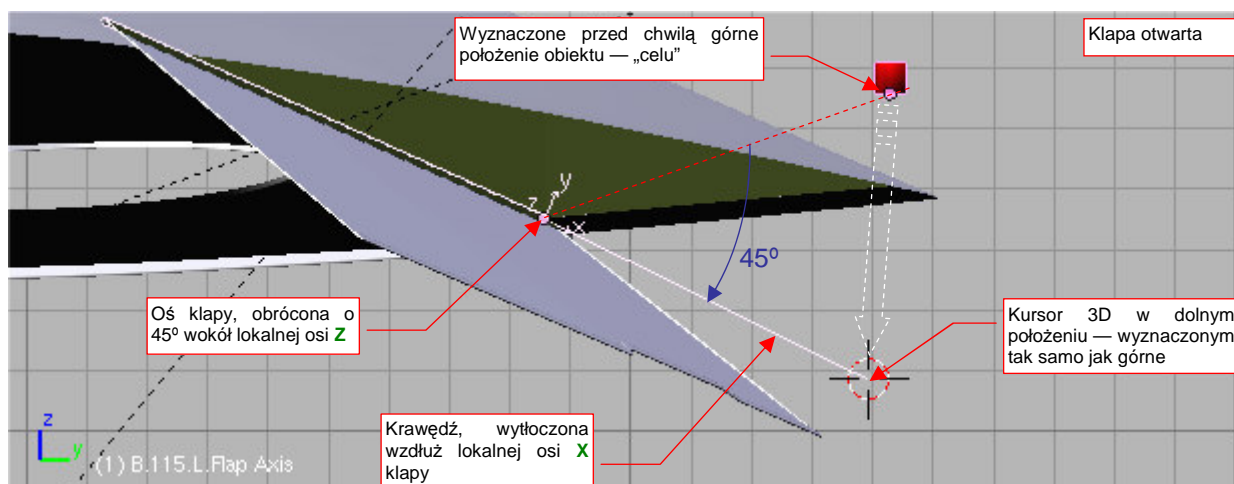


Rysunek 14.23.6 Umieszczenie środka obiektu — „celu” w punkcie wyznaczonym przez kursor 3D

Zaznacz obiekt, który ma być umieszczony tam, gdzie znajduje się kursor 3D. Następnie przełącz **Pivot** w tryb **3D Cursor** i włącz przełącznik **Move object centers only**, umieszczony obok (Rysunek 14.23.6a). (Jego włączenie powoduje, że zmiana skali i obrót stosuje się nie do całego obiektu, a tylko do jego środka). Potem naciśnij **S** (**Scale**), i trzymając wciśnięty klawisz **Ctrl** przesunij kursor myszki tak, by znalazł się ponad kursorem 3D. Nasz obiekt („uchwyt”) podąży w tym samym kierunku, a gdy skala transformacji osiągnie (0,0,0) — jego środek znajdzie się w docelowym miejscu (Rysunek 14.23.6a,b). Zatwierdź tę transformację.

- Po operacji nie zapomnij tylko o wyłączeniu przełącznika **Move object centers only**, bo inaczej nie będziesz mógł obrócić żadnego obiektu!

W analogiczny sposób możesz także określić drugie położenie „uchwyty”, odpowiadające klapie całkowicie otwartej (Rysunek 14.23.7):



Rysunek 14.23.7 Wyznaczenie dolnego położenia obiektu — „celu”

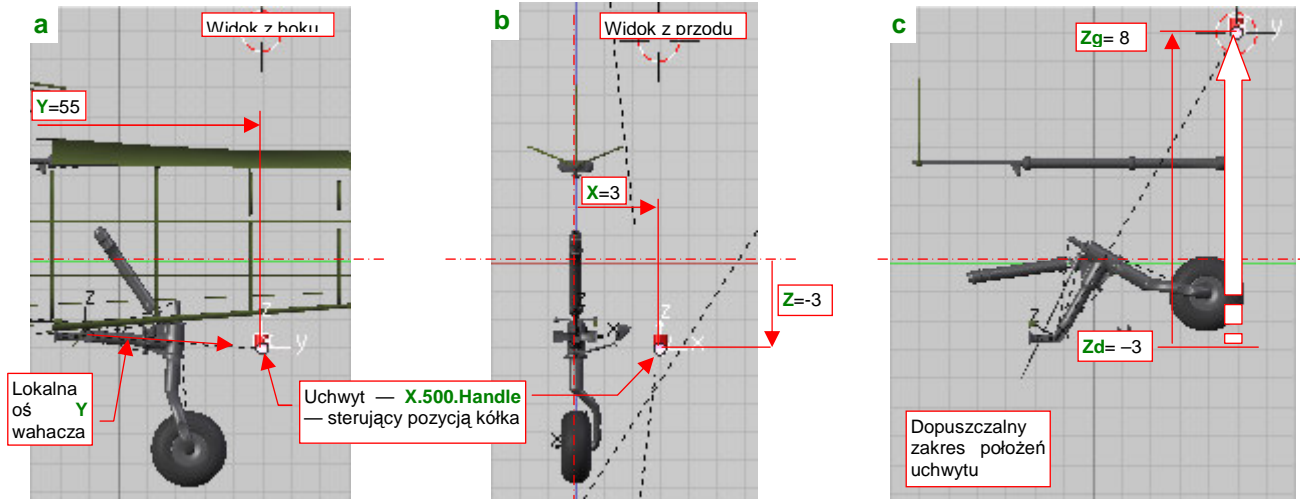
Wystarczy:

- obrócić oś (zawiasy) o odpowiedni kąt (tutaj: 45°);
- przejść do edycji siatki zawiasów;
- ponownie zaznaczyć nasz „roboczy” wierzchołek i umieścić na nim kursor 3D;
- zakończyć edycję siatki zawiasów;
- przesunąć „uchwyt” do kursora 3D;

Na koniec usuń, oczywiście, z siatki zawiasów wytłoczony podczas tej operacji wierzchołek!

14.24 Przypisanie ograniczenia *Limit Location*

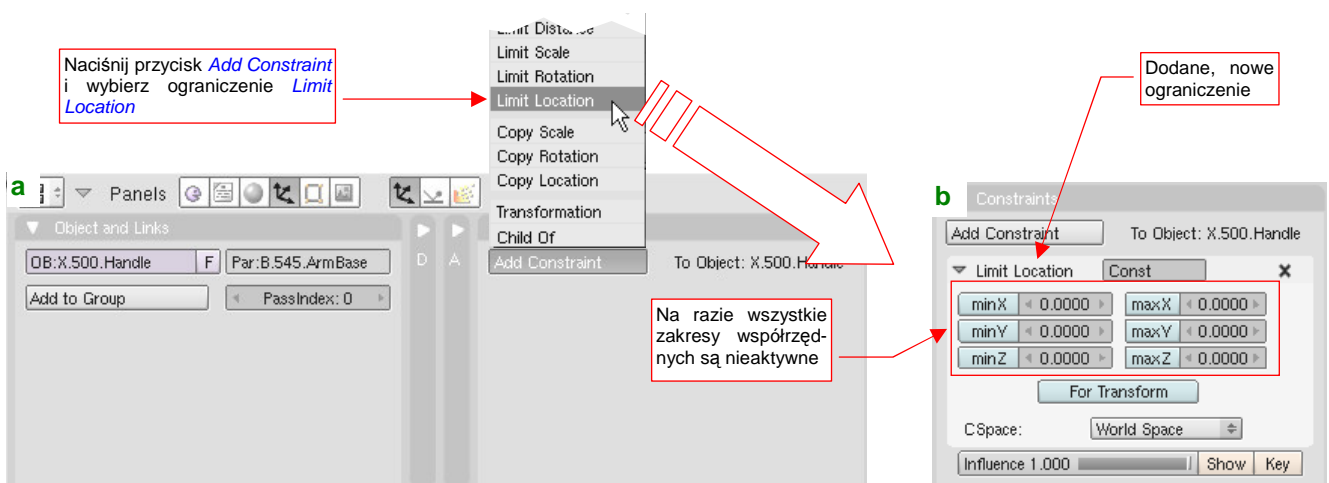
Aby części naszego modelu nie mogły się poruszać po całej scenie, tylko same „pilnowały” swoich właściwych położeń, można zastosować ograniczenie *Limit Location*. Oczywiście, opłaca się go użyć tylko dla kluczowych elementów. Na przykład — dla uchwytu odpowiedzialnego za chowanie i wysuwanie podwozia (Rysunek 14.24.1):



Rysunek 14.24.1 Położenie i dopuszczalny zakres ruchu „uchwytu”, sterującego pozycją kółka ogonowego.

O pozycji podwozia decyduje uchwyt **X.500.Handle**, którego środek jest śledzony przez lokalną oś **Y** wahacza (Rysunek 14.24.1a). (Ten efekt jest uzyskany pomocą ograniczenia *Locked Track* — por. str. 823). Gdy podwozie jest całkowicie otwarte, uchwyt znajduje się w punkcie **X=3, Y=55, Z=-3** (Rysunek 14.24.1a,b). Na razie uchwyt można omyłkowo przesunąć w każdą stronę, bez żadnych ograniczeń. Użytkownik może się więc pomylić, i ustawić podwozie w nienaturalnej pozycji. Ograniczymy to, pozwalając uchwytowi **X.500.Handle** przesuwać się wzdłuż osi **Z**, w zakresie od -3 (podwozie wysunięte) do +8 (podwozie schowane) (Rysunek 14.24.1c).

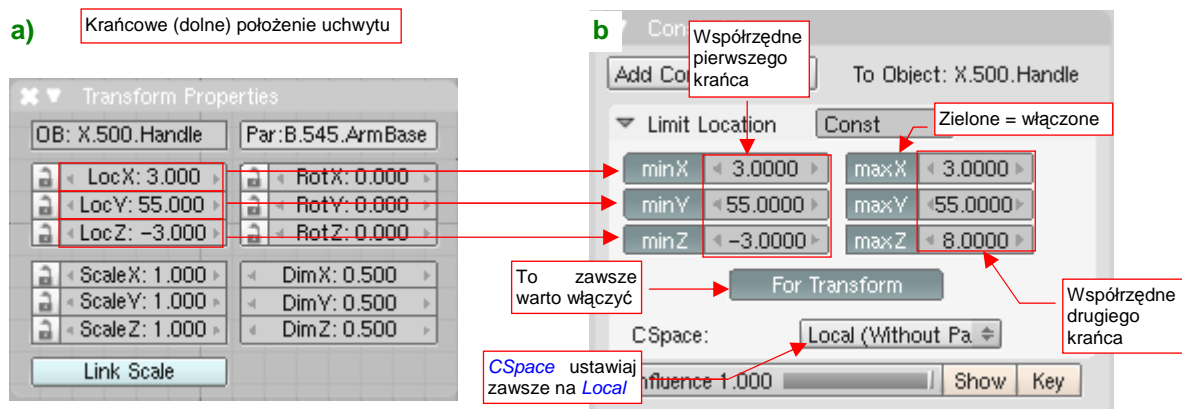
Aby dodać ograniczenie *Limit Location*, zaznacz obiekt (uchwyt), i w zestawie *Object Buttons*, panelu *Constraints*, naciśnij przycisk *Add Constraint* (Rysunek 14.24.2a):



Rysunek 14.24.2 Dodanie do obiektu ograniczenia *Limit Location*.

Z listy rozwijalnej, która się pojawi ponad przyciskiem, wybierz opcję *Limit Location*. Blender wstawi to ograniczenie na koniec listy („stosu”) ograniczeń przypisanych do obiektu. W pierwszej chwili wszystkie pola *min#* i *max#* są wyłączone (Rysunek 14.24.2b) — oznacza to, że nie obowiązuje jeszcze żadne z nich (tzn. ograniczenie nie jest aktywne).

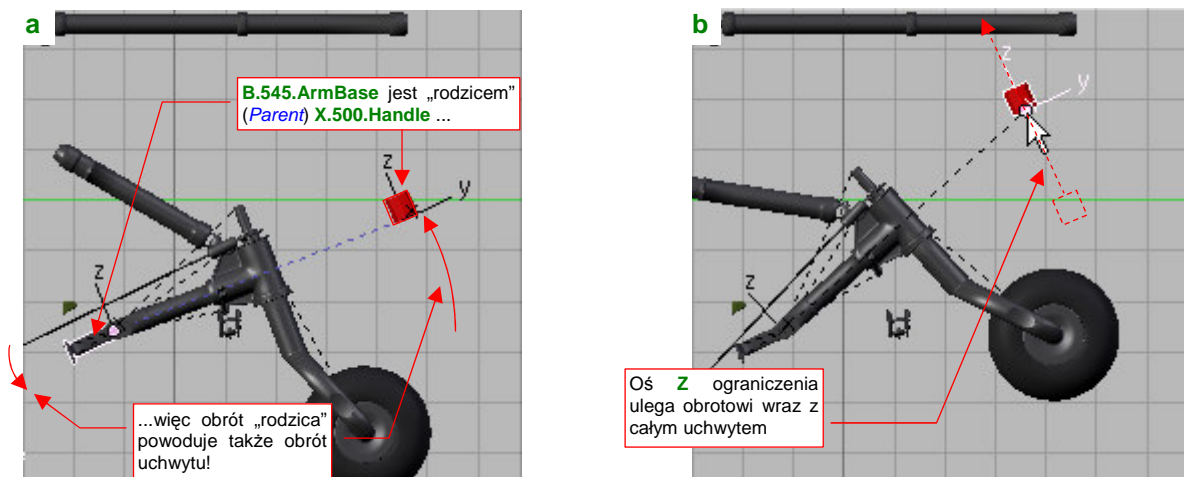
Zacznij od przepisania w pola **minX**, **maxX**, **minY**, **maxY**, **minZ**, **maxZ** aktualnych współrzędnych uchwytu, np. z okna **Transform Properties** (N) (Rysunek 14.24.3):



Rysunek 14.24.3 Przykład ustawienia parametrów ograniczenia **Limit Location**.

Gdy je przepiszesz, możesz kliknąć **LPM** w te przełączniki, by zmienić ich kolor na zielony (tj. włączyć ograniczenia) (Rysunek 14.24.3b). W pole **maxZ** wpisz wartość dopuszczalnego, górnego położenia uchwytu. Poza tym — warto jeszcze włączyć opcję **For Transform**. (Jeżeli tego nie zrobisz, po ewentualnym zdjęciu ograniczeń pozycja obiektu może Cię zaskoczyć). Przełącz także układ współrzędnych (**CSPACE**), w którym obliczane są ograniczenia na **Local (Without Parent)**. Dlaczego właśnie ten? Także po to, by uniknąć niespodzianek. W przyszłości, podczas kompozycji sceny, możesz chcieć przechylić lub przesunąć model. Uchwytów powinny być wkomponowane w hierarchię jego zespołów, i podążać za ruchem całego samolotu. Przełączenie ich ograniczeń na lokalny układ współrzędnych to właśnie zapewnienie.

Na przykład **X.500.Handle** jest przypisany do nieruchomego okucia wahacza: **B.545.ArmBase** (Rysunek 14.24.4a):

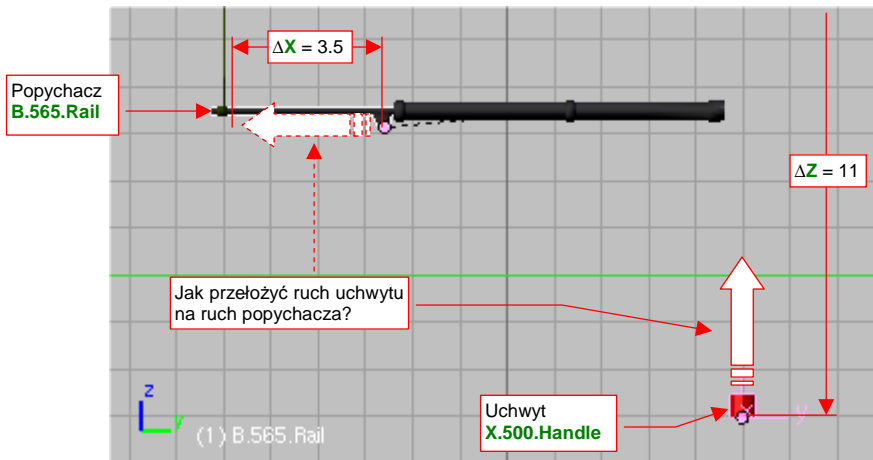


Rysunek 14.24.4 Rezultat wyboru dla **Limit Location** lokalnego układu współrzędnych (**Local**).

Gdy pochyłisz to okucie (tak jak mógłbyś pochylić cały samolot, np. aby wyglądał jak podczas nurkowania), pochyli się odpowiednio także „sztywno” powiązany z nim uchwyt (Rysunek 14.24.4a). Gdybyś w ograniczeniu **Limit Location** pozostawił jako **CSPACE** domyślny układ współrzędnych **World Space**, to ograniczenie działałoby dalej pionowo, i podwozie „samo” zaczęłoby Ci się podczas obrotu okucia przesunąć. (Taki problem miałem we wcześniejszych wersjach Blendera — do 2.44 — w których **Limit Location** było zawsze obliczane w **World Space**). Na szczęście przełączenie **CSPACE** na **Local** powoduje, że osie ograniczenia są „przymocowane” do lokalnego układu współrzędnych uchwytu. Dzięki temu ich położenie względem reszty modelu nie ulega zmianie (Rysunek 14.24.4b) i w czasie animacji samolotu nie jesteś narażony na takie zaskoczenia, jak samoczynne otwieranie się podwozia.

14.25 Przypisanie ograniczenia *Transform*

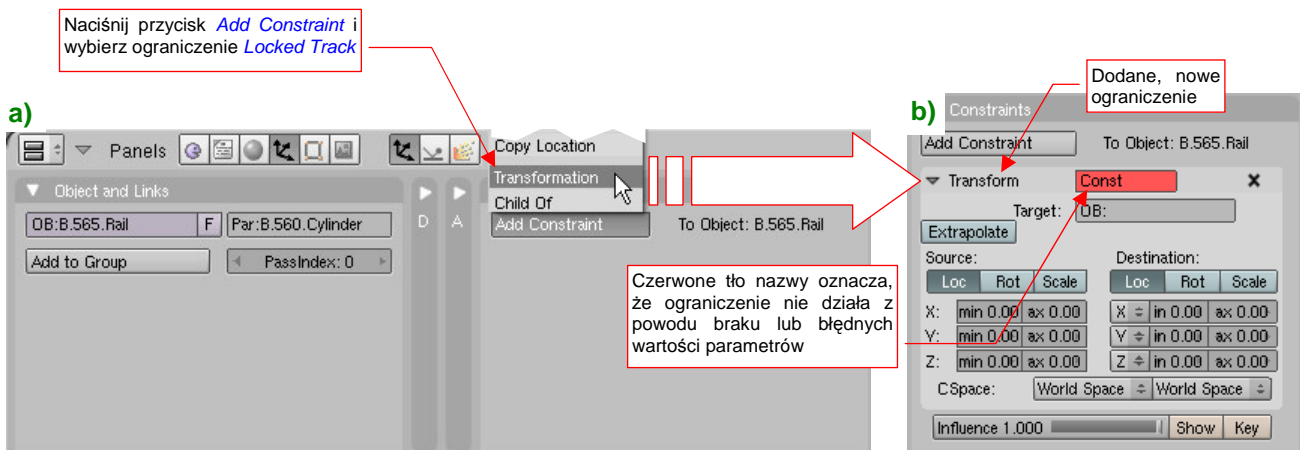
Ograniczenie *Transform* można traktować jako rodzaj „uniwersalnej przekładni”. Pokażę je na przykładzie przełożenia ruchu pionowego uchwyty (X.500.Handle) na poziome przesunięcie popychacza (B.565.Rail) (Rysunek 14.25.1):



Rysunek 14.25.1 Przykład typowego zagadnienia dla ograniczenia *Transform*.

Gdy użytkownik przesuwa uchwyt do góry, popychacz powinien przesuwać się w lewo, przy czym każdy z nich ma inną prędkość ruchu. W czasie gdy uchwyt X.500.Handle przesunie się o 11 jednostek, popychacz B.565.Rail ma przebyć tylko 3.5 (Rysunek 14.25.1). Aby uzyskać ten efekt, przypiszemy do popychacza ograniczenie *Transform*.

Aby dodać ograniczenie *Transformation*, zaznacz obiekt (uchwyt), i w zestawie *Object Buttons*, panelu *Constraints*, naciśnij przycisk *Add Constraint* (Rysunek 14.25.2a):

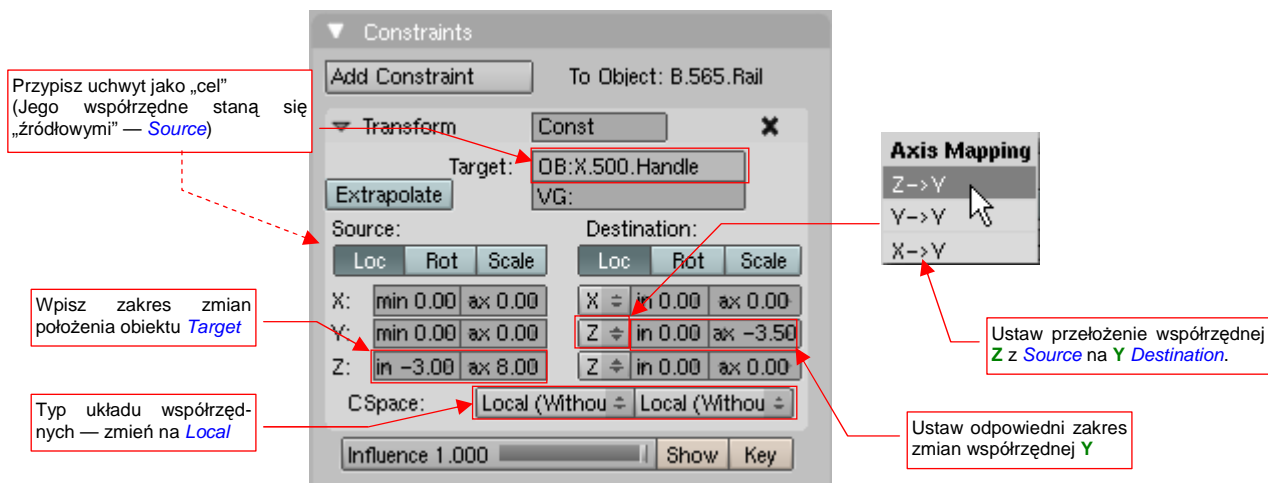


Rysunek 14.25.2 Dodanie do obiektu ograniczenia *Transformation*.

Z listy rozwijalnej, która się pojawi ponad przyciskiem, wybierz opcję *Transformation*. Blender wstawi to ograniczenie na koniec listy („stosu”) ograniczeń przypisanych do obiektu. W pierwszej chwili pole *Transform* w panelu ograniczenia, (to z nazwą), jest czerwone (Rysunek 14.23.2b). To oznacza, że coś z parametrami tego ograniczenia jest nie tak. W tym przypadku w polu *Target* nie wpisaliśmy jeszcze nazwy obiektu — „celu”. Gdy to uzupełnimy, czerwony kolor zniknie z panelu.

Zwróć uwagę, że domyślnie włączone jest przełożenie położenia (*Loc*) obiektu — „celu” (*Source*) na położenie (*Loc*) obiektu *Destination* (obektu, do którego jest przypisane ograniczenie). Możliwe są jednak i inne przełożenia — np. obrotu (*Rot*) na położenie (*Loc*), skali (*Scale*) na obrót (*Rot*), itp. Wszystko zależy wyłącznie od tego, jaką kombinację tych opcji zaznaczysz.

Ustawienia transformacji dla popychacza **B.565.Rail** przedstawia Rysunek 14.25.3:



Rysunek 14.25.3 Przykład ustawienia parametrów ograniczenia *Transform*.

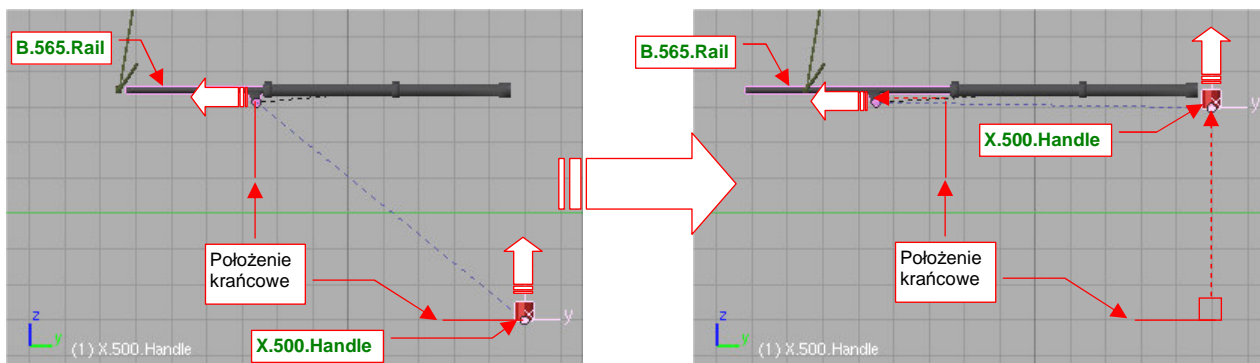
W pole *Target* wpisałem nazwę uchwytu — **X.500.Handle**. W zespole *Source* pozostawiłem śledzenie położenia (*Loc*), i wpisałem pełen zakres dopuszczalnych ruchów uchwytu (**Z**: *min* = -3.00, *max* = 8.00). Ten ruch przekładam na zmianę współrzędnej **Y** popychacza. Dlatego w zespole *Destination* wybrałem dla współrzędnej **Y** z listy rozwijalnej *Axis Mapping* kombinację **Z→Y** (Rysunek 14.25.3). W towarzyszące jej pole *min* pozostawiłem 0.00, a w pole *max* wpisałem wartość mniejszą: -3.50. W ten sposób zapewniłem popychaczowi ruch z prawej na lewo. Oczywiście, zmieniłem także typ obydwu (*Source* i *Destination*) układów współrzędnych na *Local*, by to ograniczenie było niezależne od orientacji modelu w przestrzeni (por. str. 828).

Ustawienia, które pokazuje Rysunek 14.25.3, przekazują Blenderowi prostą zależność:

- dla położenia uchwytu pomiędzy **Z** = -3.00 ... +8.00 współrzędna **Y** popychacza ma się proporcjonalnie zmniejszać, o wartości od 0.00 do -3.50;

Wartości przesunięcia dla położenia pośrednich są zawsze wyliczane proporcjonalnie (gdy uchwyt jest „w połowie drogi” — **Z** = +2.50 — to popychacz także jest już przesunięty o połowę dystansu: **Y** = -1.75)

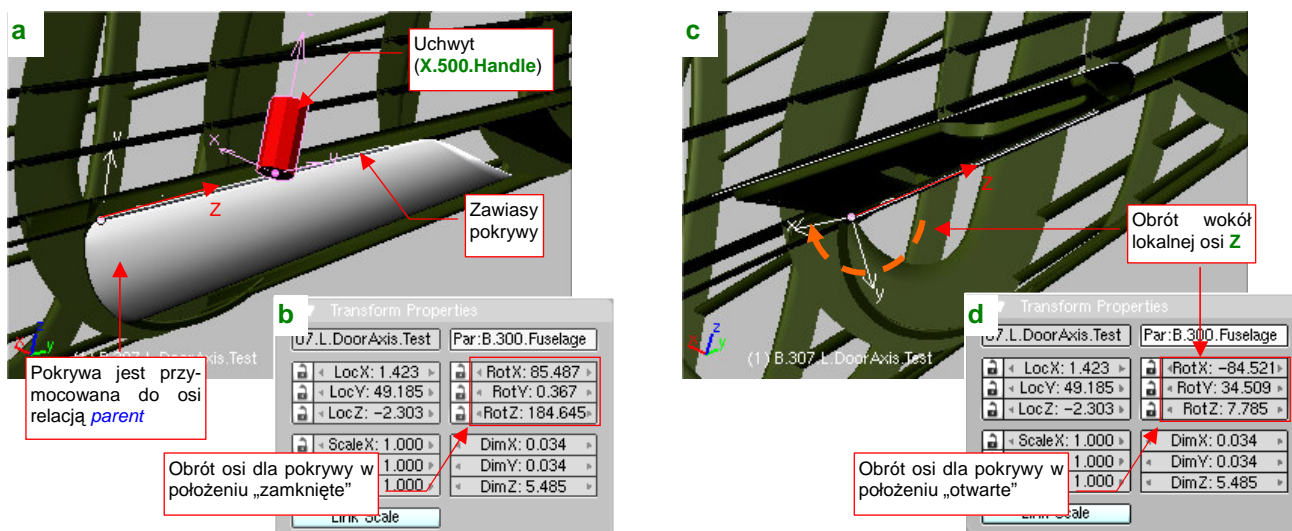
Rysunek 14.25.4 przedstawia rezultat działania ograniczenia *Transform* — ruch popychacza **B.565.Rail** wymuszony przez przesunięcie uchwytu **X.500.Handle**:



Rysunek 14.25.4 Działanie ograniczenia *Transform*.

Proporcjonalny sposób działania ograniczenia *Transform* powoduje, że nie nadaje się do „przełożenia” bardziej złożonych ruchów, np. równoczesnego obrotu względem wszystkich trzech osi. Mimo pozorów, taki ruch może wystąpić dość często — chociażby podczas obrotu wokół jednej z osi lokalnych, gdy obiekt jest obrócony względem swojego „rodzica”.

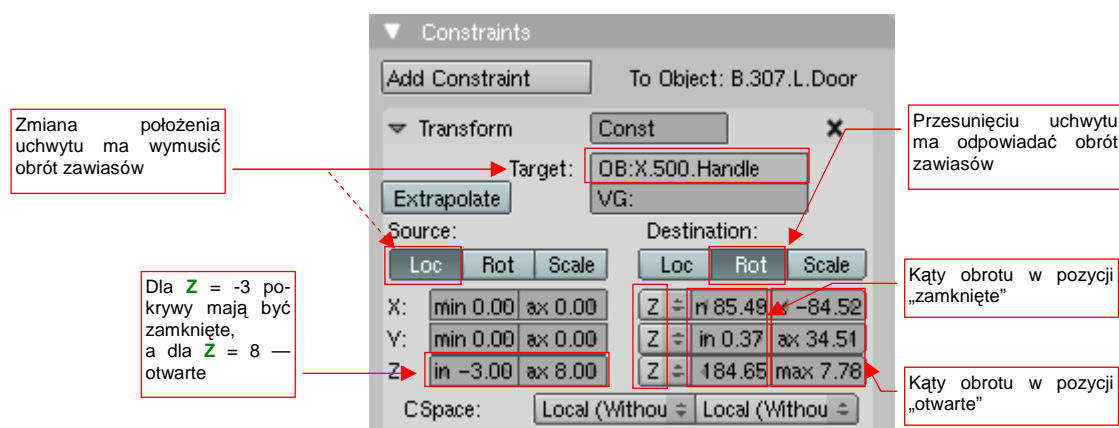
Dla przykładu, spróbujmy przełożyć za pomocą ograniczenia *Transform* przesunięcie uchwytu na obrót pokrywy podwozia (Rysunek 14.25.5):



Rysunek 14.25.5 Ruch złożony: obrót (osi) pokrywy podwozia wokół lokalnej osi Z.

Zawiasy pokrywy, które należy obrócić, to zwykle cylindry, wyciągnięte do odpowiedniego rozmiaru wzdłuż lokalnych osi Z (Rysunek 14.25.5a). Ich „rodzicem” jest kadłub. Są względem niego obrócone — wokół wszystkich trzech osi (por. okno *Transformation Properties* — Rysunek 14.25.5b). Do zawiasów jest „przymocowana” (relacją *Parent*) sama pokrywa. Aby ją otworzyć, należy obrócić zawias wokół lokalnej osi Z (Rysunek 14.25.5c). Porównaj jednak w oknie *Transformation Properties* obroty wokół osi X, Y, Z przed i po tej transformacji (por. Rysunek 14.25.5b z Rysunek 14.25.5d). Zauważ, że każdy z nich uległ zmianie: *RotX* (z 85.487 na -84.521), *RotY* (z 0.367 na 34.509), *RotZ* (z 184.645 na 7.785). (W oknie *Transform Properties* podawane są obroty względem osi „rodzica”).

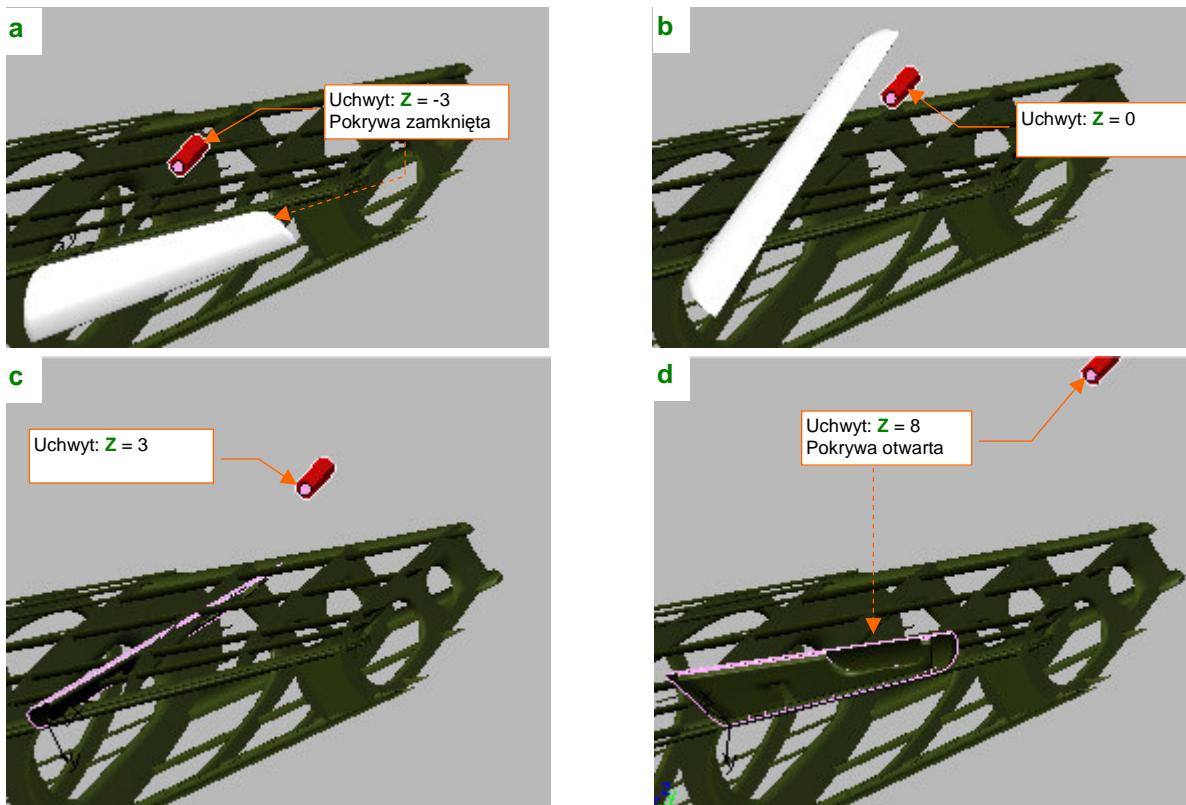
No dobrze, skoro zawiasy mają się obrócić wokół wszystkich trzech osi, to przypiszmy im ograniczenie *Transform*. Powiążmy je ze zmianą położenia (*Loc*) uchwytu (*Target: X.500.Handle*) wzdłuż osi Z (od Z = -3 do Z = 8) (Rysunek 14.25.6):



Rysunek 14.25.6 Ograniczenie, przypisane do zawiasów pokrywy podwozia

Ustalmy w obszarze *Destination*, że zmieniać się ma obrót (*Rot*). Przypiszmy zmianę każdej z trzech osi ze zmianą współrzędnej Z uchwytu („celu”). W odpowiednie pola *min* przepisujemy obroty *RotX*, *RotY*, *RotZ* z okna *Transformation Properties* dla położenia „zamknięte”. W pola *max* przepisujemy obroty *RotX*, *RotY*, *RotZ* dla położenia „otwarte”.

Po przypisaniu ograniczenia spróbuj przesunąć uchwyt. Zawiasy zaczynają się obracać, ale trajektoria pokrywy podwozia jest zupełnie nieoczekiwana (Rysunek 14.25.7a,b,c,d):

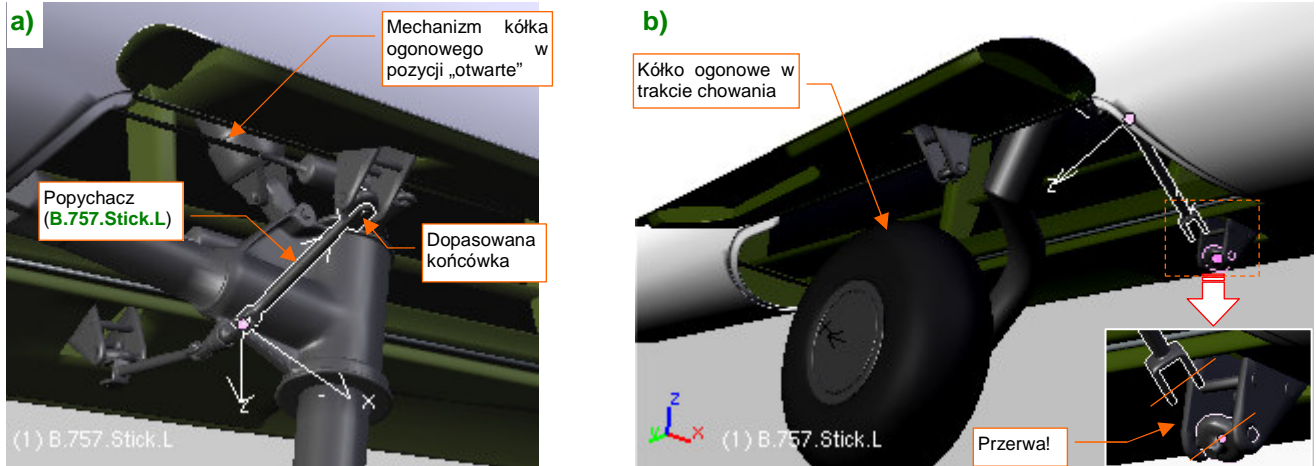


Rysunek 14.25.7 Rezultat — zaskakująca droga od jednego do drugiego położenia końcowego

O ile obydwie krańcowe pozycje są poprawne (por. Rysunek 14.25.7a,d — w końcu podaliśmy dokładne współrzędne), o tyle położenia pośrednie są bardzo dziwne (Rysunek 14.25.7b,c). Dlaczego? Dlatego, że poszczególne kąty obrotu wokół lokalnej osi **Z** zawiasów powinny się zmieniać nieliniowo (najpierw jeden z nich powinien rosnać szybciej niż pozostałe, potem inne to mają „nadrobić”). Takiego efektu nie można uzyskać za pomocą ograniczenia *Transform*. Do wywołania obrotu zawiasów lepiej jest użyć ograniczenia *Locked Track* (por. str. 823).

14.26 Przypisanie ograniczenia *Stretch To*

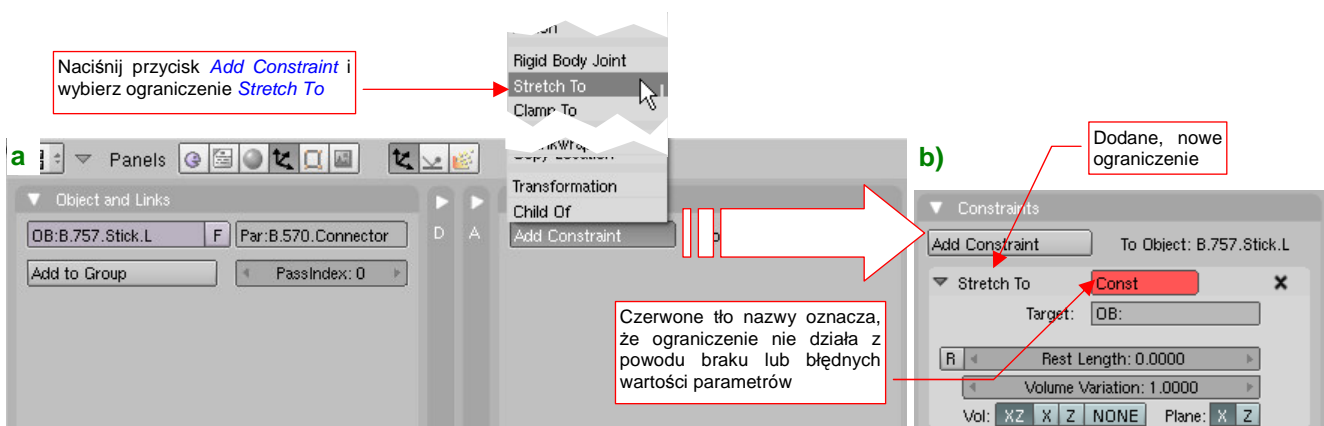
Ograniczenie *Stretch To* łączy w sobie dwie operacje: skierowania na “cel” określonej osi obiektu (przypomina w tym działanie *Locked Track*), i zmianę skali wzdłuż tej osi. Nadaje się doskonale do drobnych “naciągnięć”, maskujących pewne niedokładności w torach ruchu poszczególnych części mechanizmów. Popatrz, na przykład, na popychacze pokrywy kółka ogonowego (Rysunek 14.26.1):



Rysunek 14.26.1 Przykład typowego zagadnienia dla ograniczenia *Stretch To*.

Rysunek 14.26.1a) pokazuje mechanizm podwozia w pozycji “otwarte”. Popychacz (B.757.Stick.L) jest tu skierowany na okucie pokrywy ograniczeniem *Locked Track*. W tym położeniu wszystko wygląda poprawnie — w koniec popychacza wpasowana jest oś beczułka okucia. Gdy jednak zaczniesz składać podwozie, zauważysz że w szerokim zakresie położeni pośrednich popychacz “odrywa” się od okucia (Rysunek 14.26.1b). Na zmianę dystansu pomiędzy objektem i jego celem ograniczenie *Track To* nic poradzić nie może¹. Pomoże za to inne — *Stretch To*.

Aby dodać ograniczenie *Stretch To*, zaznacz obiekt (popychacz), i w zestawie *Object Buttons*, panelu *Constraints*, naciśnij przycisk *Add Constraint* i wybierz *Stretch To* (Rysunek 14.26.2a):

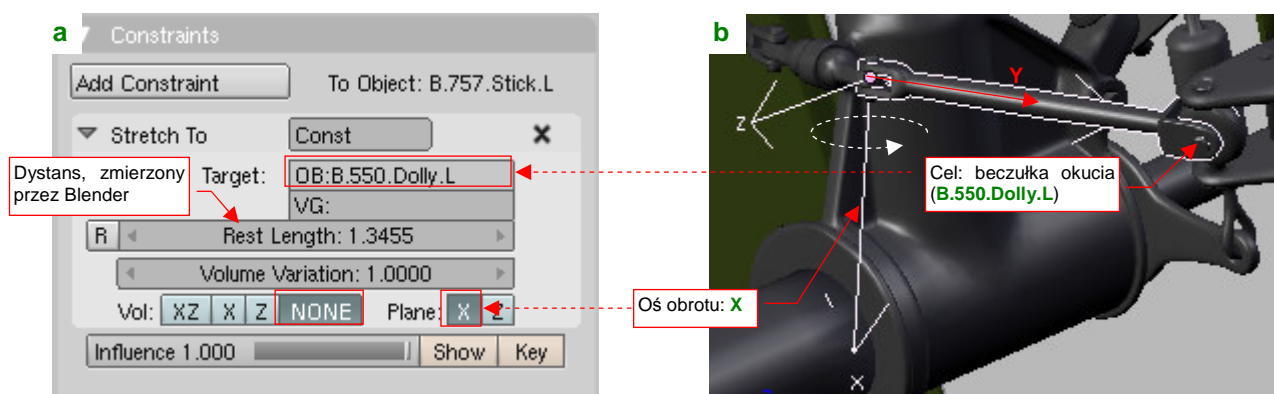


Rysunek 14.26.2 Dodanie do obiektu ograniczenia *Stretch To*.

W pierwszej chwili pole *Transform* w panelu ograniczenia, (to z nazwą), jest czerwone (Rysunek 14.23.2b). To oznacza, że coś z parametrami tego ograniczenia jest nie tak. W tym przypadku w polu *Target* nie wpisaliśmy jeszcze nazwy obiektu — „celu”. Gdy to uzupełnimy, czerwony kolor zniknie z panelu.

¹ Oczywiście, cały problem bierze się z niezbyt dokładnego odwzorowania obrotu pokrywy. Na etapie, który pokazuje Rysunek 14.26.1b), powinna być już bardziej obrócona. Niestety, zrezygnowałem z dokładnego odwzorowanie jej ruchu (jest bardzo nieliniowy, więc byłoby to bardzo pracochłonne). W efekcie muszę “tuszować” takie niedociągnięcia, jak przerwy pojawiające się między elementami mechanizmu.

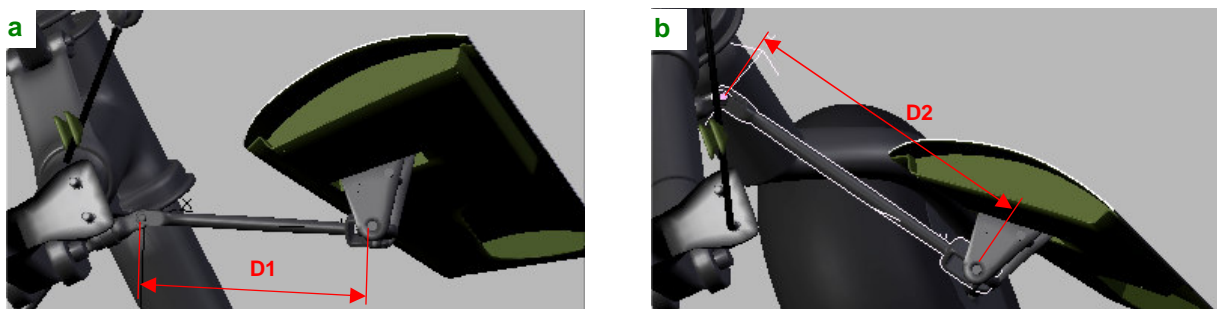
Wpisz w pole **Target** nazwę beczułki okucia (**B.550.Dolly.L**). Oprócz tego na liście **Plane** zaznacz oś **X**, a na **Vol** — **NONE** (Rysunek 14.26.3a):



Rysunek 14.26.3 Przykład przypisania parametrów **Stretch To**.

Ograniczenie **Stretch To** zawsze „celuje” we wskazany w polu **Target** obiekt lokalną osią **Y** (Rysunek 14.26.3b). (Nie można tego zmienić, tak jak w **Locked Track**). Jeżeli masz inaczej zorientowane osie — musisz wcześniej obrócić siatkę względem środka obiektu. Możesz nadal, podobnie jak w **Locked Track**, określić oś, która ma pozostać nieruchoma — **X** lub **Z**. (Traktuj ją jako „oś obrotu” — por. Rysunek 14.26.3b). Po przypisaniu obiektu Blender odnotowuje w polu **Rest Length** odległość do środka „celu”. Od tej chwili zaczyna zmieniać skalę obiektu wzdłuż osi **Y** o tyle, o ile różni się aktualny dystans od dystansu wpisanego w **Rest Length**.

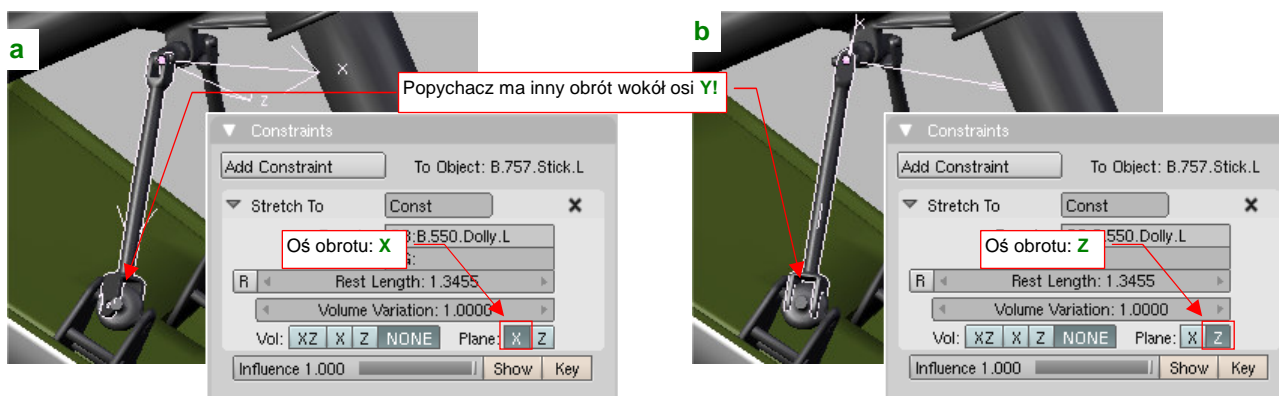
Rezultat działania ograniczenia pokazują ujęcia popychacza w dwóch kolejnych położeniach mechanizmu (Rysunek 14.26.4a i b są wykonane dokładnie w tej samej projekcji):



Rysunek 14.26.4 Rezultat ograniczenia **Stretch To**.

Koniec popychacza trzyma się „jak przyklejony” beczułki okucia. Przy okazji rozciąga się jak guma — zwróć uwagę, że dystanse **D1** i **D2** na ilustracjach bardzo się różnią!

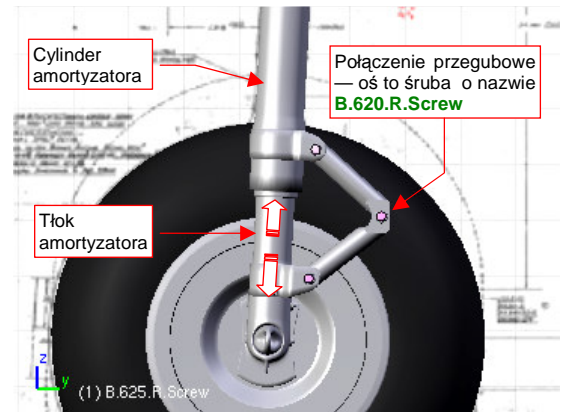
Przy pracy z ograniczeniem **Stretch To** zwracaj zawsze uwagę na oś podaną w parametrze **Plane**. Rysunek 14.26.5 pokazuje, jak może się różnić orientacja obiektu w przestrzeni dla każdej z opcji:



Rysunek 14.26.5 Rezultat przełączenia „blokady” osi obrotu z **X** na **Z**.

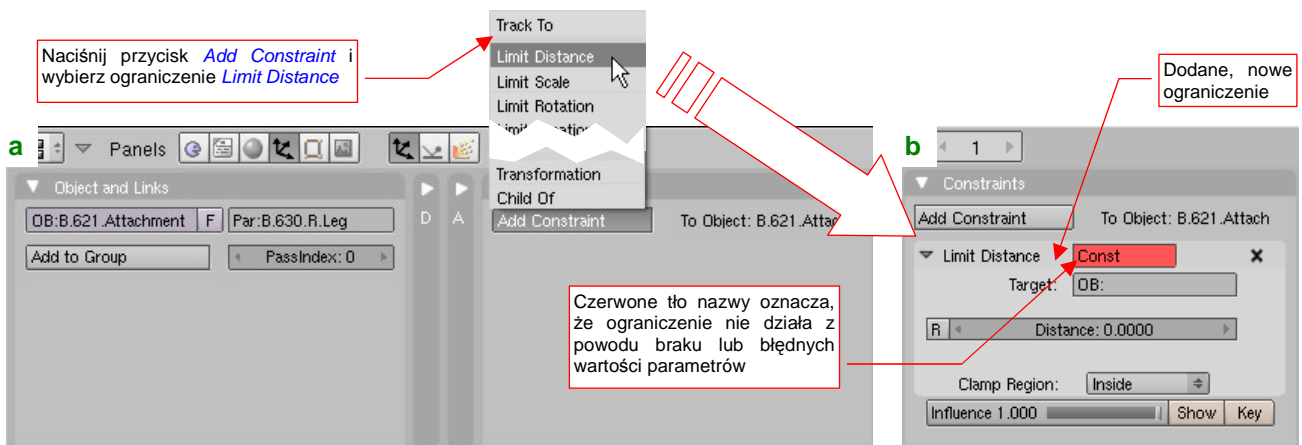
14.27 Przypisanie ograniczenia *Limit Distance*

Ograniczenie *Limit Distance* wymusza na obiekcie zachowanie odpowiedniej odległości od środka obiektu — celu. Ta właściwość przyda się do zamodelowania działania połączenia przegubowego, takiego jak u tych dwóch łączników na amortyzatorze goleni podwozia (Rysunek 14.27.1). To typowe rozwiązanie, stosowane w konstrukcji samolotów od początku lat czterdziestych: dwa składające się „jak nożyce” pręty, zapobiegające przypadkowemu obrotowi koła wokół goleni. Wydaje się, że najłatwiej odwzorować ten mechanizm za pomocą wewnętrznego szkieletu (*armatury* — por. str. 815). Można to także zrobić w inny sposób. Wystarczy użyć ograniczeń *Limit Distance* i *Stretch To*.



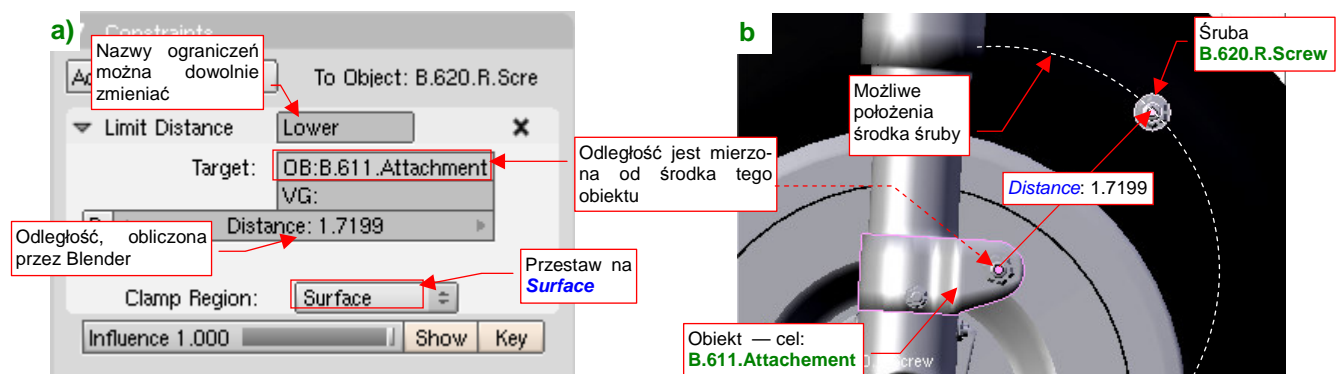
Rysunek 14.27.1 Połączenie przegubowe

Zacznijmy od dodania ograniczenia *Limit Distance*, do osi przegubu. W naszym przykładzie to śruba B.620.R.Screw (por. Rysunek 14.27.1). W zestawie *Object Buttons*, panelu *Constraints*, naciśnij przycisk *Add Constraint* (Rysunek 14.27.2a):



Rysunek 14.27.2 Dodanie do obiektu ograniczenia *Limit Distance*.

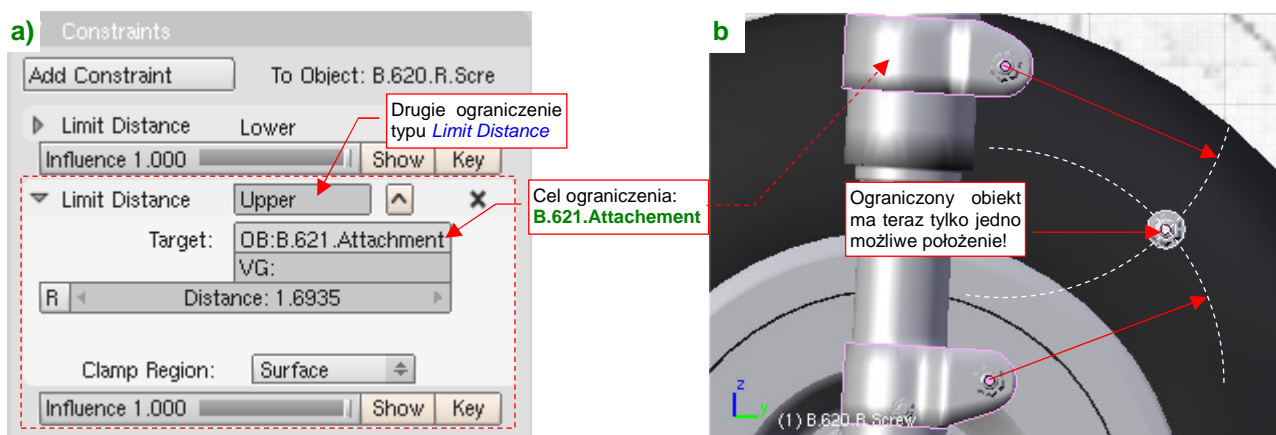
Z listy rozwijalnej, która się pojawi ponad przyciskiem, wybierz opcję *Limit Distance*. W pierwszej chwili pole *Limit Distance* w panelu ograniczenia, (to z nazwą), jest czerwone (Rysunek 14.27.2b). To oznacza, że coś z parametrami tego ograniczenia jest nie tak. Po prostu w *Target* nie wskazaliśmy jeszcze obiektu — „celu”. Wpisz tam nazwę okucia (**B.611.Attachment**), a czerwony kolor zniknie z panelu (Rysunek 14.27.3a):



Rysunek 14.27.3 Przykład przypisania parametrów *Limit Distance*.

Jednocześnie przełącz *Clamp Region* na *Surface*. To wymusza na śrubie zachowanie stałej odległości od środka okucia (Rysunek 14.27.3b). (Zauważ, że środek okucia nie musi być jego środkiem geometrycznym).

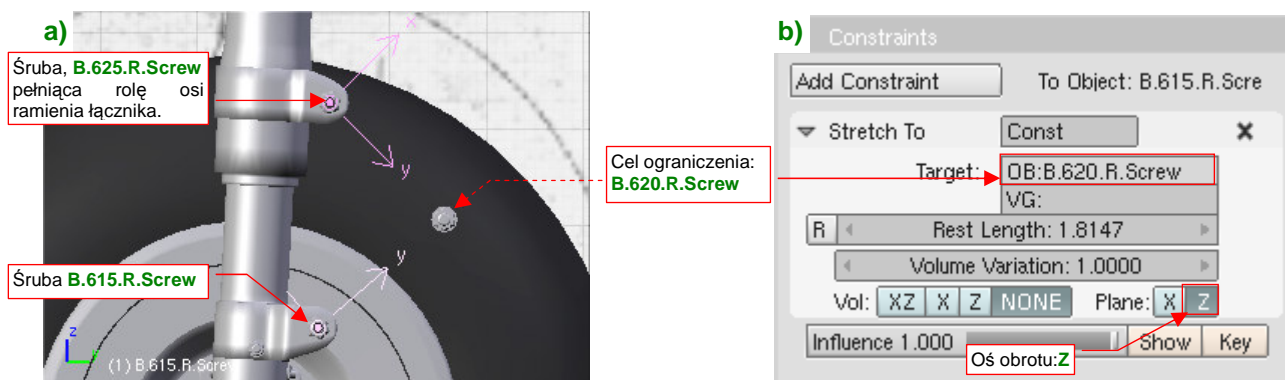
Z jednego ograniczenia *Limit Distance* nie ma jeszcze efektu, o który chodzi. Dodaj jednak do tej samej śruby (**B.620.R.Screw**) drugie ograniczenie tego samego typu — tyle że tym razem względem górnego okucia (**B.621.Attachment**) (Rysunek 14.27.4a):



Rysunek 14.27.4 Przykład zastosowania dwóch ograniczeń *Limit Distance*.

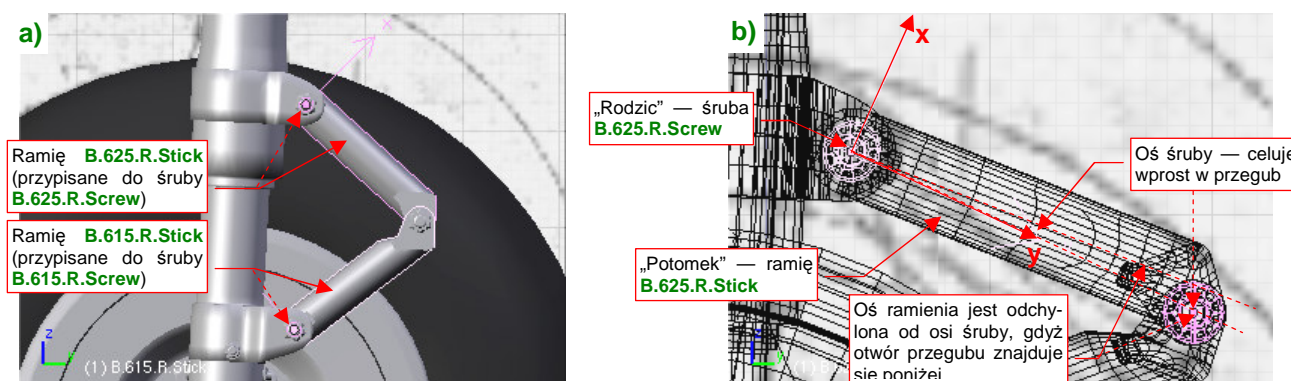
Rysunek 14.27.4b) pokazuje, że środek ograniczonej w ten sposób śruby może leżeć tylko w jednym, jedynym miejscu: tam, gdzie odległości od obydwu okuców są takie same. Ten punkt zmienia się, gdy okucia zbliżą się do siebie lub oddalą, w wyniku ruchu amortyzatora. W ten sposób ograniczenia *Limit Distance* zastępują ramiona łączników.

Teraz pozostaje tylko przypisać śrubom w osiach obrotu ramion ograniczenia *Stretch To* (por. str. 833) — tak, by ich lokalne osie **Y** „celowały” w oś przegubu (Rysunek 14.27.5a):



Rysunek 14.27.5 Przypisanie osiom łączników ograniczeń *Stretch To*.

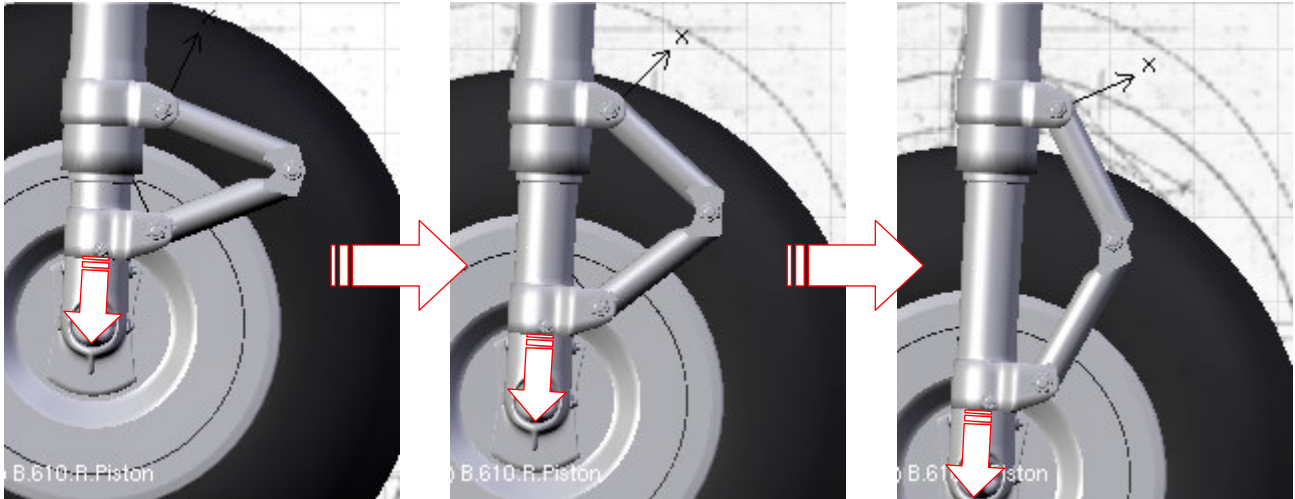
(Rysunek 14.27.5b) pokazuje parametry jednego z tych ograniczeń — jako *Target* jest wpisana śruba **B.620.R.Screw**, a osią obrotu jest lokalna oś **Z**). Potem pozostaje tylko przypisać do osi same ramiona (relacją *Parent*), by same się kierowały na oś przegubu (Rysunek 14.27.6a):



Rysunek 14.27.6 Przypisanie łączników do osi obrotu.

Zwróć uwagę, że ramiona są lekko odchylone od osi **Y** ich „rodziców” — śrub (Rysunek 14.27.6b).

Rysunek 14.27.7 przedstawia mechanizm, który stworzyliśmy, w działaniu:



Rysunek 14.27.7 Działanie mechanizmu połączenia przegubowego.

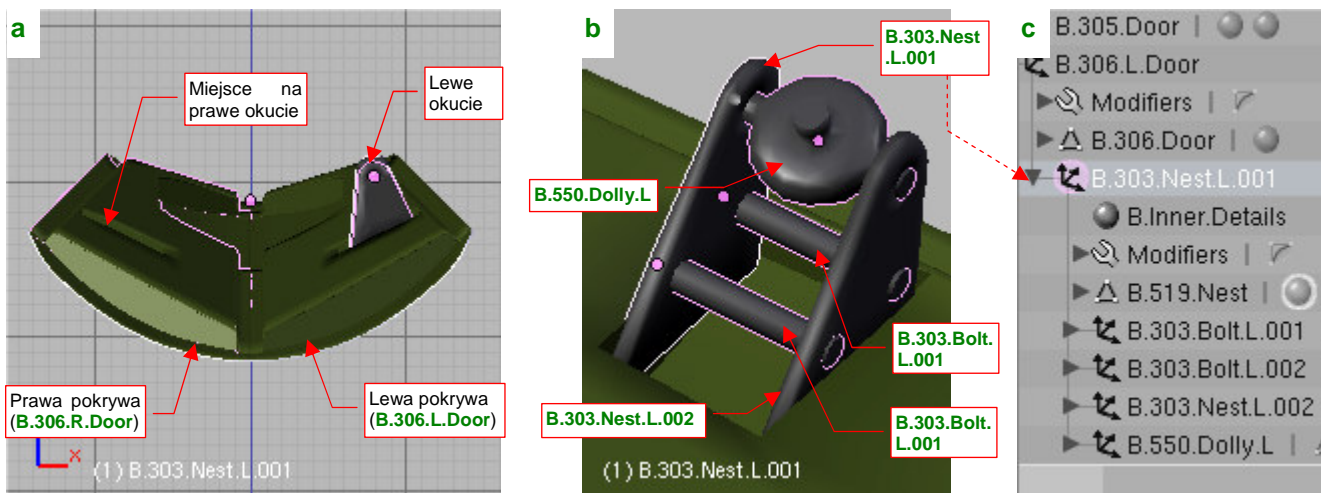
Wygląda to na poprawnie zginające się połączenie przegubowe. Na pierwszy rzut oka efektu nie można odróżnić od klasycznego zastosowania armatury (por. str. 815 i następane). Na koniec tej sekcji warto odnotować, że:

- Jako cel ograniczeń *Limit Distance* można wskazać wprost śruby w osiach obrotu ramion łącznika (chodzi o obiekty **B.625.R.Screw**, **B.615.R.Screw** — por. str. 836, Rysunek 14.27.5). W ten sposób mechanizm byłby złożony z mniejszej liczby obiektów, czyli — prostszy. Nie zrobiłem tego jednak ze względów „porządkowych”. (Blender wtedy wyświetla od czasu do czasu w konsoli ostrzeżenia o wzajemnej referencji — *circular reference* — tych dwóch śrub). Zamiast tego użyłem ich „rodziców” — okuć, którym specjalnie ustawiłem środki w osiach obrotu ramion.
- Gdyby otwory na oś przegubu **B.620.R.Screw** leżały na osi ramiona łączników (**B.625.R.Stick**, **B.615.R.Stick**) — ograniczenia *Stretch To* zostałyby przypisane bezpośrednio do nich. Wtedy ramiona mogłyby być bezpośrednimi „potomkami” okuć, a śruby, leżące w ich osiach obrotu, nie byłyby w mechanizmie potrzebne. Zastosowałem je wyłącznie dlatego, że oś przegubu leży poniżej lub powyżej osi podłużnej ramienia łącznika, więc nie można jej wycelować prosto w jego środek (por. str. 836, Rysunek 14.27.6b).
- Teoretycznie, do skierowania ramion na oś przegubu, zamiast ograniczeń *Stretch To* można by było zastosować ograniczenia *Locked Track* (por. str. 823). Jednak wtedy końcówki ramion mogą w pewnych położeniach „odstawać” od przegubu. Wygląda na to, że Blender wyznacza położenie wynikające z ograniczeń *Limit Distance* w sposób przybliżony (iteracja?), co prowadzi do powstania takich odstępów. Jedynym lekarstwem na to są ograniczenia *Stretch To*, które same dbają o to, by końcówki ramion były „dociągnięte” do zadanego punktu. (Niewielkie zmiany skali wzdłuż lokalnych osi **Y** śrub są przenoszone na obiekty potomne — ramiona. I to wystarcza).

14.28 Odbicie lustrzane (*Mirror*) i uboczne efekty ujemnej skali obiektu

Transformacja *Mirror* jest właściwie skrótem — specyficzną formą zmiany skali obiektu poprzez pomnożenie jej wartości względem jednej z lokalnych osi przez -1 . Można ją wywołać skrótem **Ctrl-M** lub poleceniem **Object→Mirror**. (Ten sam rezultat można uzyskać poprzez odpowiednią zmianę skali w jednym kierunku). W tej sekcji pokażę praktyczny przypadek zastosowania lustrzanego odbicia (*Mirror*). Omówię także pewien zakłajający efekt uboczny, który może wystąpić dla rezultatów tej transformacji — obiektów o skali ujemnej.

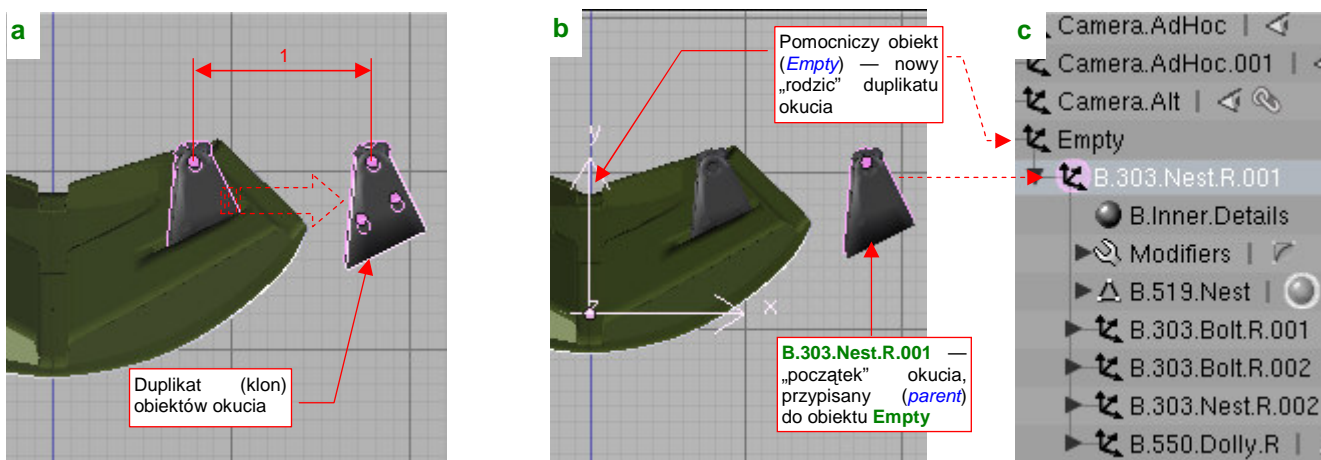
Naszym zadaniem jest skopiowanie i przypisanie do prawej pokrywy luku podwozia okucia, do którego ma być potem przyczepiony popychacz (Rysunek 14.28.1a):



Rysunek 14.28.1 Okucie pokrywy podwozia i jego struktura

Okucie składa się z dwóch trójkątnych blach, połączonych bolcami (nazwy poszczególnych części podaje Rysunek 14.28.1b). W hierarchii modelu (Rysunek 14.28.1c) jedna z tych blach (**B.303.Nest.L.001**) jest „rodzicem” (*parent*) pozostałych elementów okucia. Jednocześnie jej „rodzicem” jest lewa pokrywa podwozia (**B.306.L.Door**).

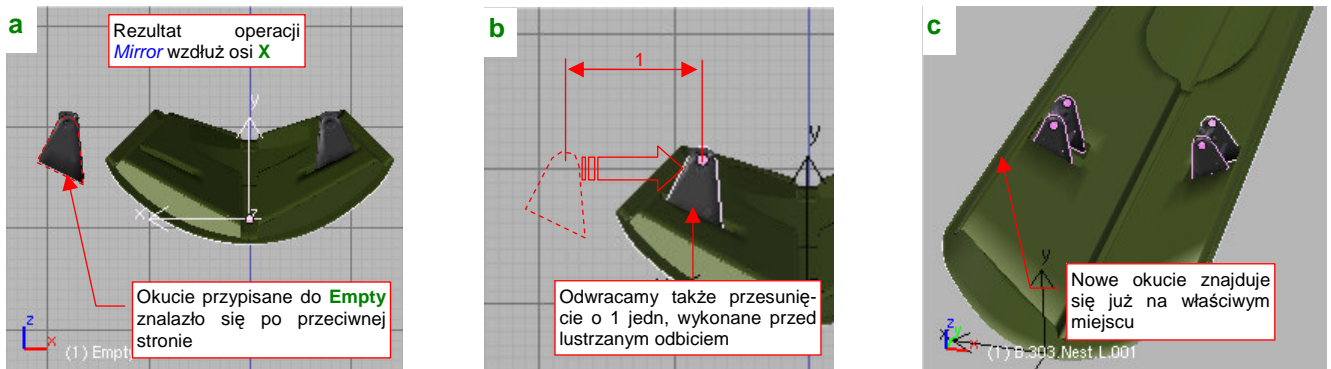
Zaznacz wszystkie elementy okucia i zduplikuj je (**Alt-D**, str. 796). Zaraz po stworzeniu przesuń te duplikaty w bok, na jakiś „równy” i łatwy do zapamiętania dystans — np. o 1 jednostkę Blendera (Rysunek 14.28.2a). (Robimy to wyłącznie po to, by łatwo było je zaznaczać). Przy okazji zmień w nazwach nowo utworzonych obiektów końcówki **L** na **R** (choć jeszcze na to nie wygląda, będzie to okucie prawej pokrywy):



Rysunek 14.28.2 Powiązanie duplikatu okucia z pomocniczym obiektem *Empty*

W związku z tym, że polecenie *Mirror* działa tylko wzdłuż lokalnych osi obiektu, musimy „obejść” to ograniczenie. Wstaw na chwilę w osi samolotu pomocniczy obiekt *Empty* (jak — str. 791). Uczyń go rodzicem blaszki **B.303.Nest.R.001** (Rysunek 14.28.2c).

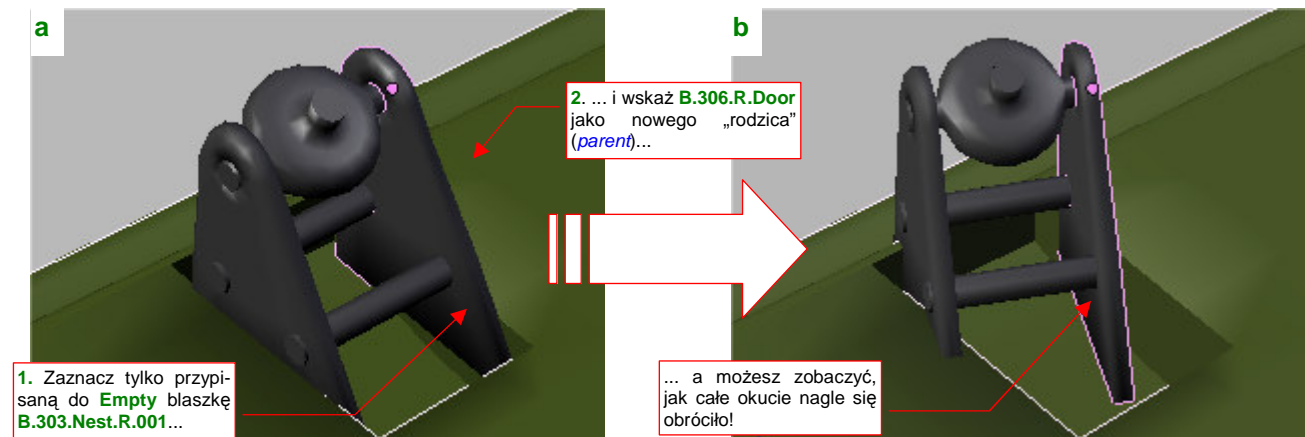
Teraz można wywołać transformację *Mirror* (**Ctrl-M**) dla obiektu **Empty**. To w istocie zmiana skali wzdłuż osi **X** z 1.0 na -1.0. „Rodzic” okucia — blaszka **B.303.Nest.R.001** — jest jednocześnie „dzieckiem” **Empty** (Rysunek 14.28.2c). Zmiana skali „rodzica” przekłada się na wszystkie obiekty do niego przypisane, stąd okucie znalazło się już po przeciwległej stronie osi samolotu (Rysunek 14.28.3a):



Rysunek 14.28.3 Przeniesienie duplikatu okucia ponad drugą pokrywę (*Mirror*)

Przesuń jeszcze okucie wzdłuż osi **X** o 1 jedn. (por. str. 838, Rysunek 14.28.2a) — tak, jak to pokazuje Rysunek 14.28.3b). W rezultacie mamy je już w miejscu docelowym (Rysunek 14.28.3c).

Pozostał do wykonania ostatni krok: zamiana „rodzica” okucia (a konkretnie: jego blaszki **B.303.Nest.R.001**) z tymczasowego obiektu **Empty** na pokrywę podwozia (**B.306.R.Door**) (Rysunek 14.28.4):



Rysunek 14.28.4 Niespodziewany efekt zmiany przypisania (*Parent*) obiektu

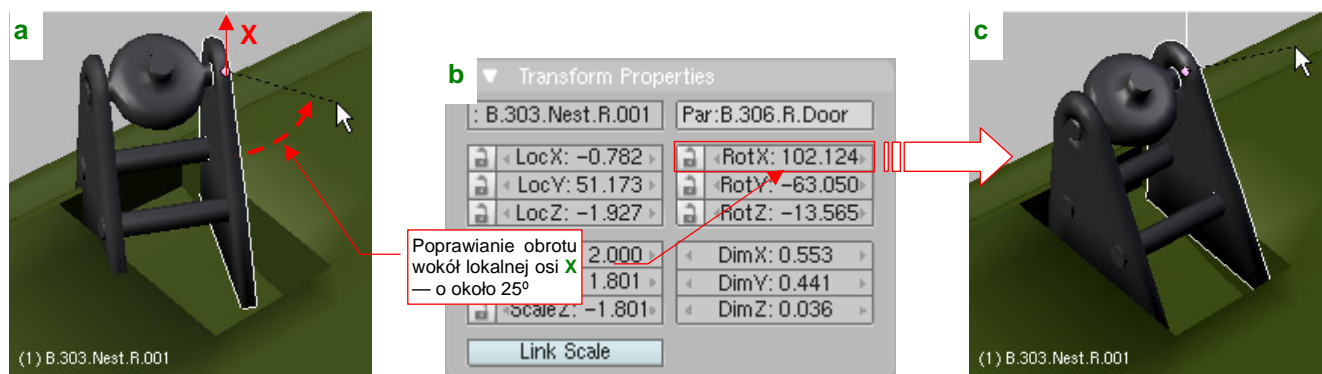
I nagle możesz stwierdzić, że nieoczekiwanie, tylko w wyniku zmiany „rodzica”, okucie obróciło się o jakiś dziwny kąt (Rysunek 14.28.4b)! Co się dzieje!?

No cóż, tak właśnie może się objawić efekt uboczny rezultatu operacji *Mirror* — zmiany skali na ujemną. Nim cokolwiek z tym zrobisz, przyjrzyj się uważnie transformacjom każdego z okuć (Rysunek 14.28.5):



Rysunek 14.28.5 Analiza różnic transformacji

Prawe okucie jest wyraźnie skrzywione, więc zacznijmy od kątów obrotu. Spróbuj najpierw obrócić blaszkę **B.303.Nest.R.001** wokół każdej z lokalnych osi: **X**, **Y**, **Z**. Wygląda na to, że zdecydowanie poprawia sprawę obrót wokół lokalnej osi **X** (Rysunek 14.28.6):

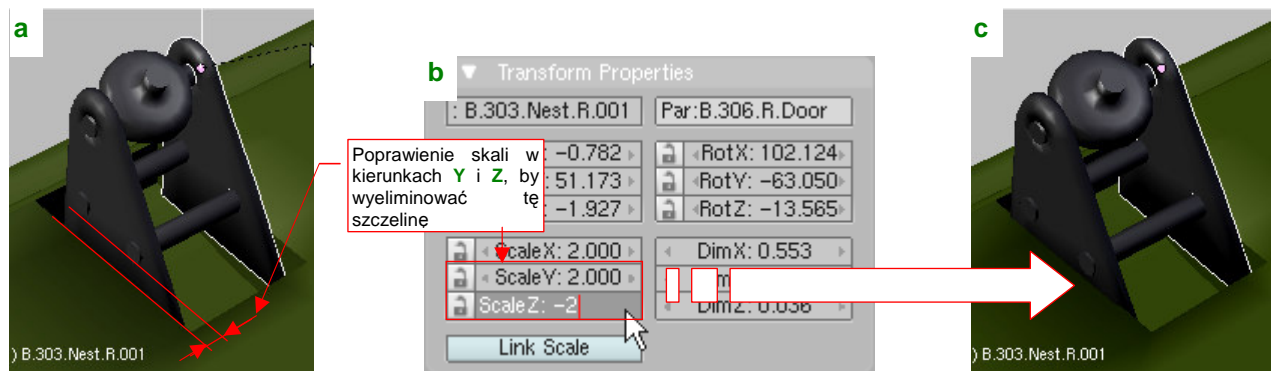


Rysunek 14.28.6 Pierwsza poprawka: obrót wokół lokalnej osi **X**

W wyniku operacji *Mirror* zmianie mogą ulec kąty obrotu obiektu. Ich zmiana zależy od osi, wzdłuż której nastąpiła transformacja. W tym przypadku (porównując Rysunek 14.28.5a i b) wygląda na to, że Blender zwiększył kąt **RotX** o 180° , zamiast zmienić mu znak na przeciwny.

- Nie potrafię znaleźć wyjaśnienia dla tak dziwnego zachowania Blendera. Przypuszczam, że jest to objaw jakiegoś błędu w programie. W chwili obecnej (czerwiec 2010) wszystkie wysiłki *Blender Foundation* są skierowane na prace nad nową wersją Blendera — 2.5. Gdy opublikowany zostanie Blender 2.5 beta, sprawdzę czy ten błąd w nim nadal występuje, i postaram się go zgłosić do poprawy.

Po wyprostowaniu okucia widać wyraźniej, że coś jest nie tak z jego rozmiarem — z boku otworu w pokrywie podwozia pojawiła się szczelina (Rysunek 14.28.7a):

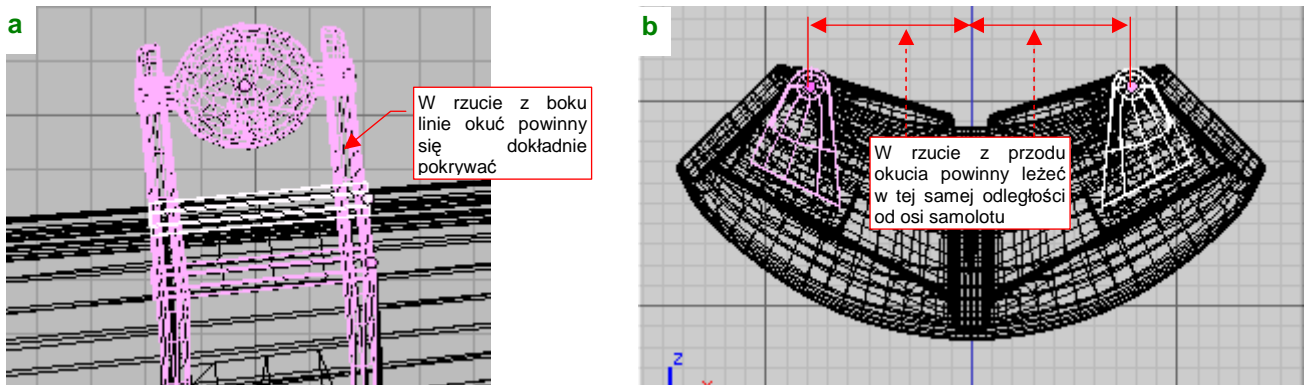


Rysunek 14.28.7 Druga poprawka: zmiana skali wzdłuż lokalnych osi **Y** i **Z**

Tym razem przyczynę łatwo dostrzec: z zupełnie niezrozumiałych powodów skala „rodzica” okucia — blaszki **B.303.Nest.R.001** — została zmniejszona w kierunkach **Y** i **Z** z 2.000 do 1.801 (por. Rysunek 14.28.5a). Skala „rodzica” jest dziedziczona przez obiekty potomne, stąd całe okucie jest w tych kierunkach mniejsze. Wystarczy wpisać w pola **ScaleY** i **ScaleZ** z powrotem poprawne wartości (Rysunek 14.28.7b), by „dociągnąć” krawędź okucia do krawędzi jego otworu w pokrywie podwozia (Rysunek 14.28.7c).

- Podczas poprawiania skali w oknie *Transform Properties* zmieniaj wartości, ale nie znaki! Jeżeli byś je zmienił, odwrócisz efekt transformacji *Mirror* (por. Rysunek 14.28.7b, znak w polu **ScaleZ**)

Na koniec warto jeszcze przełączyć się na tryb wyświetlania *Wireframe* (**Z**), i sprawdzić, czy w rzucie z boku linie obydwu okuć dokładnie się pokrywają (Rysunek 14.28.8a):



Rysunek 14.28.8 Weryfikacja poprawności — w rzucie z boku i z przodu

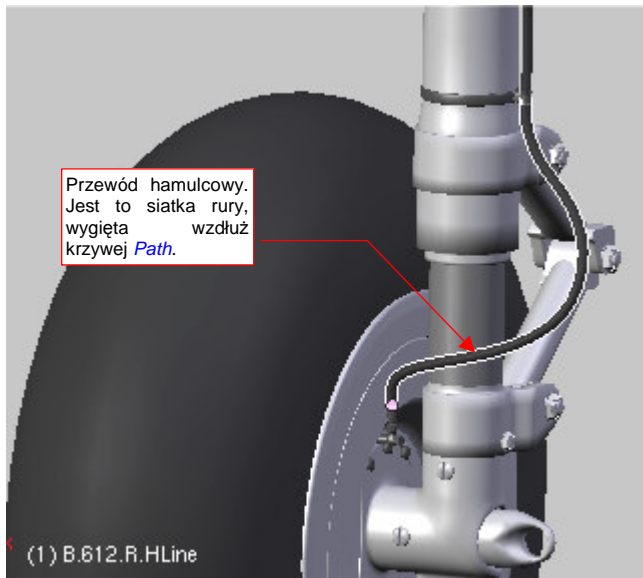
Nie zaszkodzi zerknąć także na rzut z przodu (Rysunek 14.28.8b), czy są w równych odległościach od osi samolotu. Rysunek 14.28.9 przedstawia ostateczny rezultat — prawidłowo ustawione okucia, przypisane do odpowiednich pokryw luku podwozia:



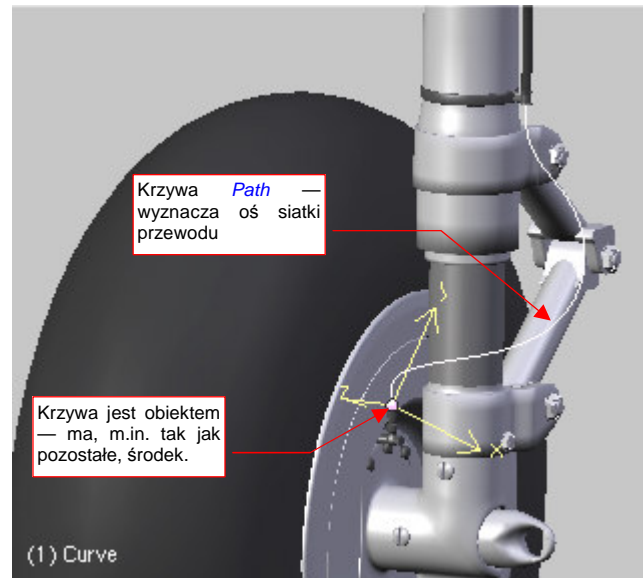
Rysunek 14.28.9 Ostateczny rezultat

14.29 Wstawienie krzywej (*Curve*)

Zastosowanie dla krzywych znajduję w Blenderze stosunkowo rzadko: gdy trzeba zamodelować jakiś wygięty kabel czy rurkę. Przykładem takiego elementu jest przewód z płynem hamulcowym, doprowadzany do piasty koła głównego (Rysunek 14.29.1):



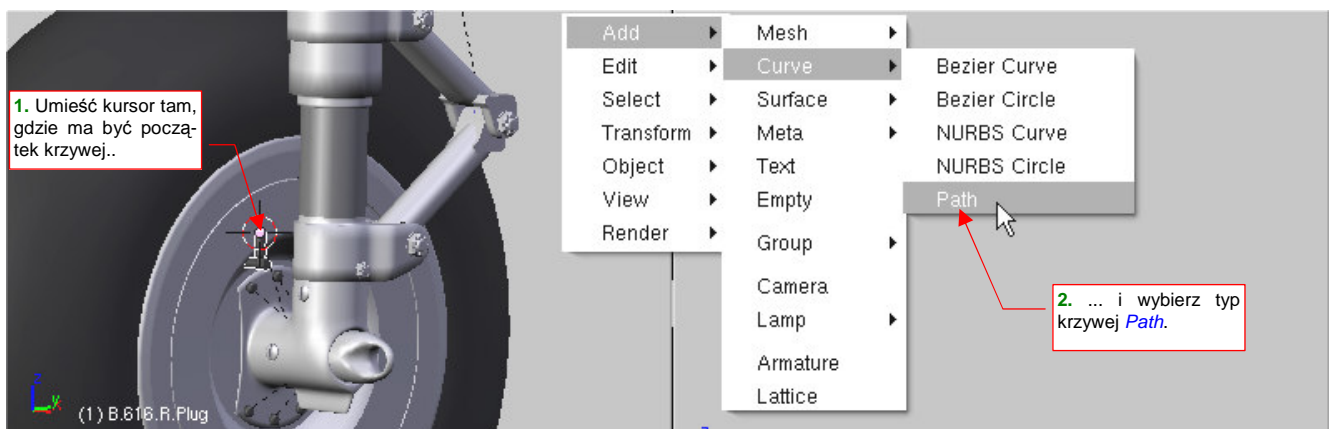
Rysunek 14.29.1 Przykład siatki, wygiętej wzdłuż krzywej



Rysunek 14.29.2 Krzywa, determinująca kształt siatki

Przewód, który pokazuje Rysunek 14.29.1, został uformowany jako prosta „rurka”. Następnie wygiąłem go, za pomocą odpowiedniego modyfikatora *Curve Deform* (por. str. 913) wzdłuż odpowiedniej krzywej przestrzennej (Rysunek 14.29.2). To jeden z typów linii dostępnych w Blenderze. Można byłoby tę siatkę „powyginać” bez pomocy krzywej, ale tak jest wygodniej modyfikować jej kształt. W dodatku mamy gwarancję, że przekrój poprzeczny tak wygiętej „rury” nie będzie w żadnym miejscu zdeformowany.

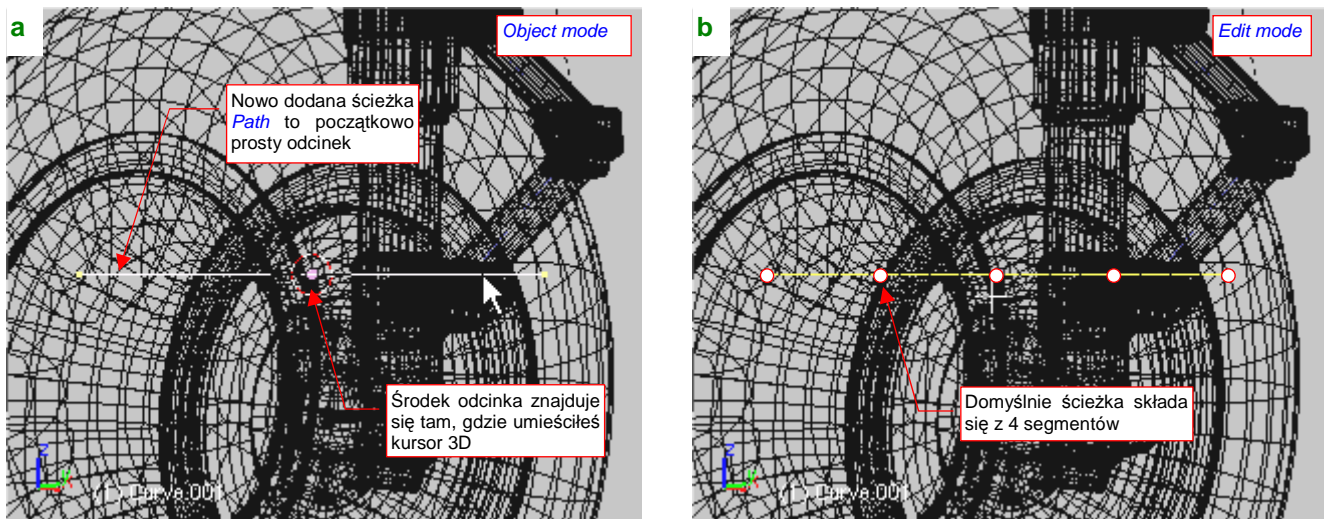
Aby dodać do rysunku krzywą (ścieżkę), umieść kursor 3D tam, gdzie ma być jej „punkt odniesienia” (np. początek). Następnie naciśnij **Space** (lub rozwiń menu *Add* — jest zaraz obok menu *File*). Wywołaj polecenie *Add→Curve→Path* (Rysunek 14.29.3):



Rysunek 14.29.3 Tworzenie krzywej typu *Path*.

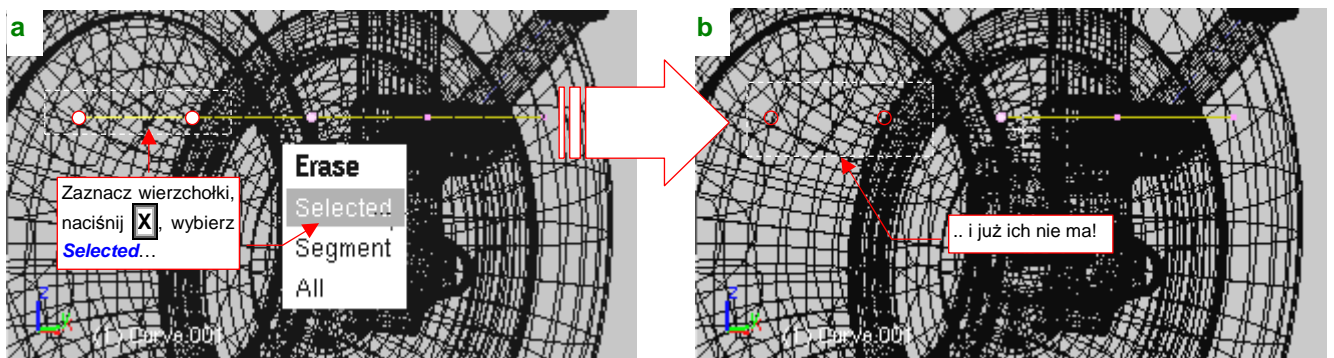
Zwróć uwagę, że masz do wyboru w Blenderze aż pięć rodzajów krzywych. Niech Cię to jednak nie kusi — wszystkie poza *Path* muszą być zupełnie „płaskie”. (To znaczy, że, wszystkie punkty krzywych Bezier czy Nurbs muszą leżeć na jednej płaszczyźnie, dowolnie zorientowanej w przestrzeni). Tylko *Path* jest prawdziwą krzywą przestrzenną

Blender zawsze wstawia obiekt typu *Path* jako zupełnie prosty, poziomy odcinek. Jego geometryczny środek znajduje się tam, gdzie kursor 3D (Rysunek 14.29.4a):



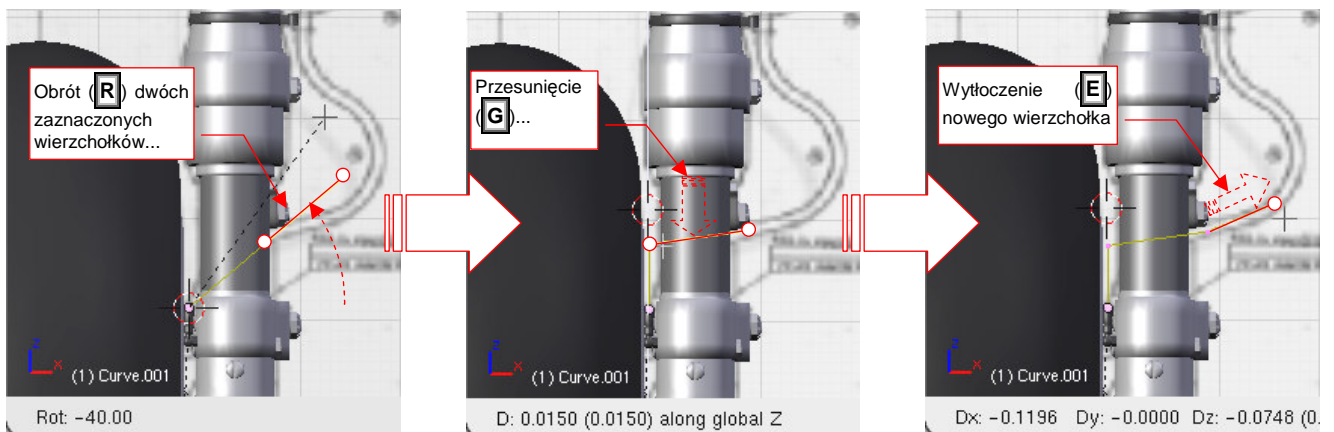
Rysunek 14.29.4 Nowo utworzona ścieżka (krzywa) *Path* — jako obiekt i w trybie edycji

(Rysunek 14.29.4 pokazuje obiekty w trybie wyświetlania *Wireframe*, gdyż część odcinka przenika do wnętrza innych powłok). Krzywą można zmieniać tak samo, jak zmienia się siatki. Wystarczy przełączyć się w tryb edycji (**Tab**). Zaraz po utworzeniu ścieżka składa się z czterech równych segmentów, łączących pięć wierzchołków (Rysunek 14.29.4b). Aby usunąć niepotrzebne wierzchołki, wystarczy je zaznaczyć i nacisnąć **X** (*Curve* → *Delete*) (Rysunek 14.29.5):



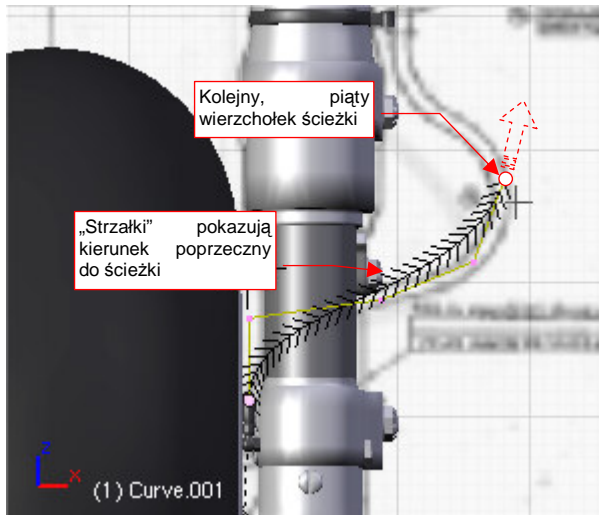
Rysunek 14.29.5 Usuwanie wierzchołków ze ścieżki

W menu, które się pojawi (Rysunek 14.29.5a), wystarczy wybrać opcję *Selected*. Wierzchołki ścieżki możesz obracać (**R**), przesuwać (**G**), czy wyłaczać (**E**) tak samo, jak wierzchołki siatki (Rysunek 14.29.6):

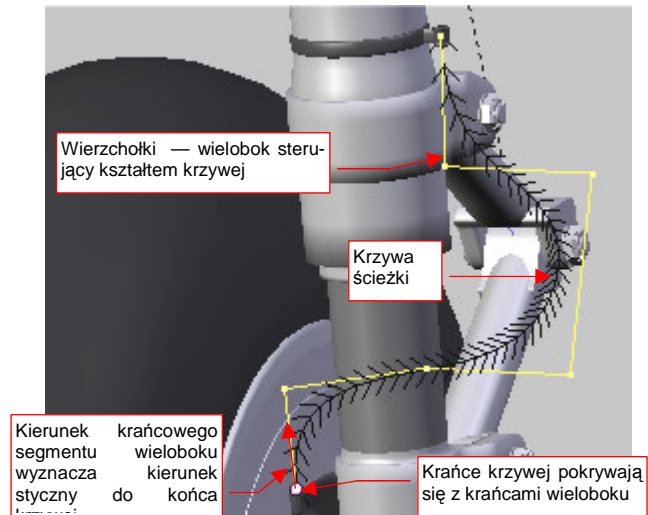


Rysunek 14.29.6 Dalsza edycja ścieżki

Po wyłoczeniu piątego wierzchołka ścieżka staje się linią krzywą (Rysunek 14.29.7):



Rysunek 14.29.7 Dalsza edycja ścieżki



Rysunek 14.29.8 Uformowana krzywa przestrzenna

Na krzywej ścieżki Blender rysuje pomocnicze „strzałki”. Pokazuje w ten sposób kierunek poprzeczny w każdym punkcie tej linii. (Ta informacja jest używana np. do orientacji kamery, przesuwanej wzdłuż ścieżki, lub innych efektów animacji). Wierzchołki, którymi manipulujemy, to punkty sterujące kształtem krzywej. Tworzą tzw. wielobok sterujący. Odpowiednio dobierając jego kształt, można uzyskać krzywą przestrzenną, wyglądającą jak wygięty kawałek elastycznego przewodu (Rysunek 14.29.8).

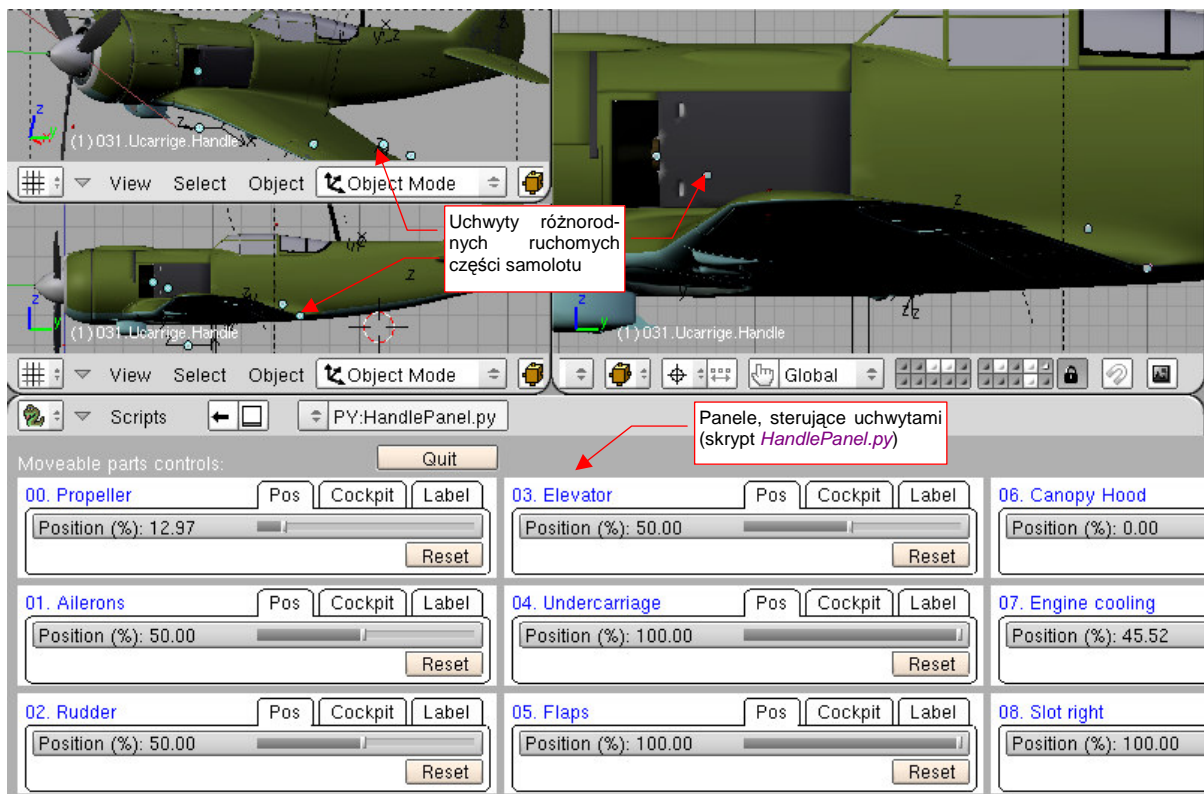
- Końce krzywej *Path* znajdują się zawsze w krańcowych wierzchołkach wieloboku sterującego. Kierunki krańcowych segmentów wieloboku wyznaczają kierunki styczne na końcach krzywej (Rysunek 14.29.8) .
- Minimalna liczba wierzchołków, przy której *Path* zaczyna być krzywą, wskazuje na to, że jest to tzw. krzywa czwartego stopnia. Jest to najmniejszy stopień krzywej przestrzennej, zapewniający ciągłość i „gładkie” zmiany promienia krzywizny wzdłuż jej całej długości.

Opis, jak wygiąć siatkę przewodu wzdłuż tej krzywej znajdziesz w sekcji o modyfikatorze *Curve Deform* (na str. 913 i następnych)

14.30 Panel sterowania ruchem obiektów (*HandlePanel.py*)

W ukończonym modelu znajdzie się około dziesięciu uchwytów (pomocniczych obiektów, służących do poruszania w określony sposób jakiejś części). Jeden służy do wysuwania i chowania podwozia, inny steruje obrotem steru kierunku, jeszcze inny — lotkami, kolejny — klapami, itd. Za jakiś czas, gdy zajrzysz do pliku z ukończonym modelem, możesz mieć problemy z szybkim zidentyfikowaniem, do czego służy konkretny element jego „chmurki” uchwytów. Odkrywanie na nowo ich funkcji „metodą prób i błędów” nie jest specjalnie zachęcającą perspektywą.

Dlatego opracowałem specjalny dodatek do Blendera. Jest to skrypt *HandlePanel.py*¹. Po uruchomieniu przeszukuje cały model i znajduje w nim wszystkie uchwyty. Dla każdego znalezionego uchwytu wyświetla panel z opisową nazwą i „suwakiem”, umożliwiającym wygodne określenie aktualnego położenia każdego z podzespołów samolotu (Rysunek 14.30.1):



Rysunek 14.30.1 Panel sterowania zespołami samolotu

Nazwy, które widzisz na ilustracji, zostały przypisane poszczególnym uchwytom za pomocą pola w zakładce **Label** (umieszczonej na każdej z paneli). Te teksty są przechowywane i zapisywane razem z całym modelem (jako dodatkowe dane poszczególnych obiektów).

¹ Skrypt powstał dla wersji 2.43 Blendera. W tej wersji ograniczenia *Limit Location*, ograniczające zakres ruchu uchwytów, były jeszcze niedopracowane. Używały współrzędnych mierzonych względem układu współrzędnych obiektu — „rodzica” uchwytu. W dodatku, gdy obracałeś cały model, kierunek działania tych ograniczeń nie obracał się wraz z nim, i wszystkie mechanizmy się „rozjeżdżały”. Aby temu zapobiec, gdy zakres ruchu uchwytu był już sprawdzony, wyłączałem jego współczynnik wpływu (*Influence* = 0). Za resztę odpowiadał skrypt *HandlePanel.py*. Odczytywał z ograniczeń uchwytu dopuszczalny zakres ruchu, i stosował go odpowiednio, nawet gdy model został obrócony. Dzięki wyłączeniu współczynnika wpływu ograniczenia żaden „rozjazd” mechanizmów modelu nie występował.

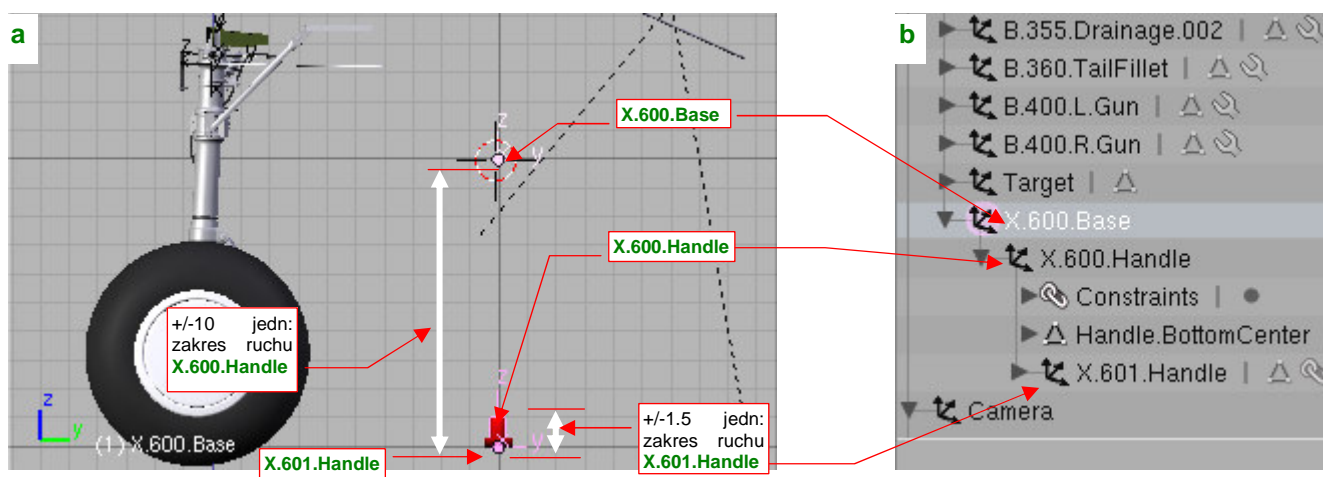
W wersji 2.49 kierunek ograniczeń *Limit Location* obraca się poprawnie, razem z obrotem „rodzica”. Jednocześnie zmieniono konwencję zapisu zakresu ograniczeń: teraz są wyrażone we współrzędnych globalnych, a nie lokalnych. Nową wersję skryptu *HandlePanel.py* dołączyłem do materiałów, towarzyszących tej książce. Uważaj tylko w modelu Ła-5: wymaga starej wersji skryptu! Robiłem go w Blenderze 2.43, i od tego czasu wartości jego ograniczeń są nadal względne. W tekstach (oknie *Text Editor*), towarzyszących temu modelowi, znajdziesz starą wersję *HandlePanel.py*. Uruchom ją (**Alt-P**) by wygodnie sterować tym modelem.

Skąd *HandlePanel.py* wie, co jest uchwytem, a co nim nie jest? To proste: program buduje i wyświetla panel dla każdego obiektu spełniającego następujące kryteria:

- jego nazwa musi mieć przyrostek ***.Handle**, np. **X.600.Handle** (uważaj więc na takie nazwy);
- ma przypisane ograniczenie *Limit Location*, z włączoną opcją *Local (without Parent)*;
- jest przyporządkowany (relacją *Parent*) do jakiegoś innego obiektu („rodzica”);

Program zakłada, że środek (tzn. punkt odniesienia — *origin*) „rodzica” znajduje się w położeniu neutralnym uchwyty. Położenie neutralne uchwyty powinno odpowiadać jakiejś szczególnej pozycji mechanizmu. (Na przykład steru kierunku jest w nim ustawiony dokładnie wzdłuż osi kadłuba, podwozie jest zupełnie wciągnięte, a klapy — schowane).

Przygotowałem dwa uchwyty, sterujące ruchem podwozia — **X.600.Handle** i **X.601.Handle** — tak, by skrypt mógł je odnaleźć (Rysunek 14.30.2):



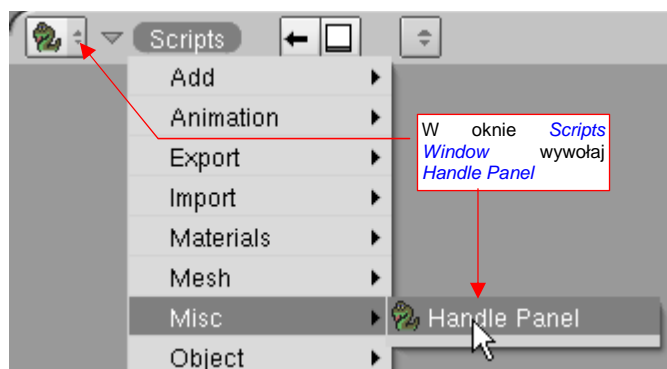
Rysunek 14.30.2 Przypisanie uchwyty „rodzica”

Uchwyt **X.600.Handle** steruje wysuwaniem i chowaniem podwozia. Ma przypisane ograniczenie *Limit Location*, wymuszające jego położenie do pionowego odcinka o długości 10 jednostki Blendera (Rysunek 14.30.2a). Gdy jest w położeniu górnym, podwozie jest schowane. W tym właśnie, górnym punkcie znajduje się jego „rodzic” (Rysunek 14.30.2b) — pusty (*Empty*) obiekt **X.600.Base**. Wyznacza w ten sposób domyślne położenie podwozia — w pozycji „złożone”.

Drugi uchwyt — **X.601.Handle** — służy do „ściskania” i „rozciągania” amortyzatorów kół głównych i pochylania wahacza kółka ogonowego. Ma przypisane ograniczenie *Limit Location*, wymuszające ruchu tego uchwyty wzdłuż pionowego odcinka o długości 1.5 jednostki Blendera. (Odpowiada to dokładnie skokowi amortyzatorów podwozia głównego). Rodzicem **X.601.Handle** jest uchwyt **X.600.Handle**. W dolnym położeniu, środek **X.601.Handle** pokrywa się z **X.600.Handle**. Położenie neutralne amortyzatora odpowiada więc całkowitemu rozciągnięciu (brakowi obciążenia).

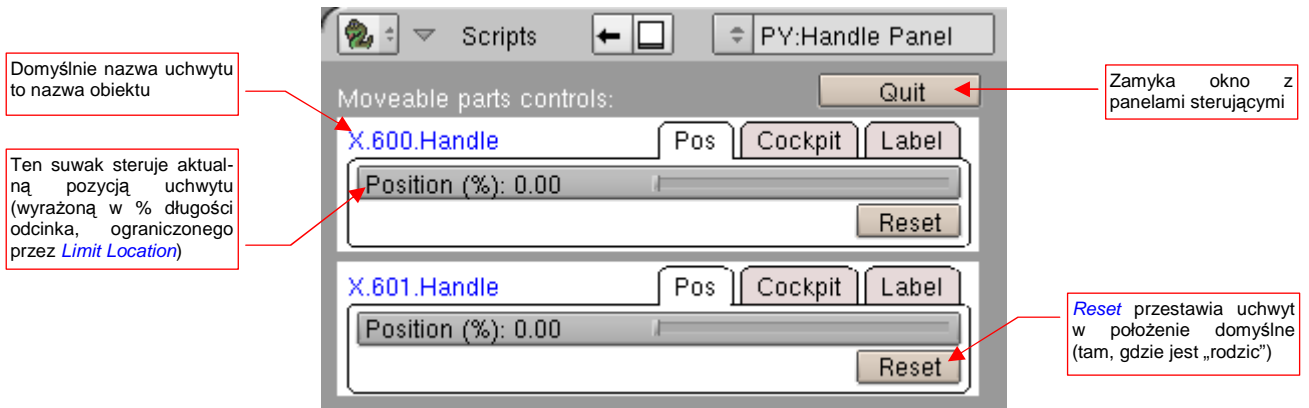
Po tych przygotowaniach można wywołać skrypt. Przełącz na chwilę okno przycisków (*Buttons Window*) na okno skryptów (*Scripts Window*). O ile ustawiłeś ścieżki tak, jak podałem to na str. 759, powinieneś go znaleźć w menu **Scripts → Misc → Handle Panel** (Rysunek 14.30.3).

Jeżeli nie znajdziesz go tam — załaduj do okna tekstowego (*Text Editor*) dostarczony wraz z tą książką plik *source/python/HandlePanel.py*. Potem uruchom go poleceniem **Text → Run Python Script**.



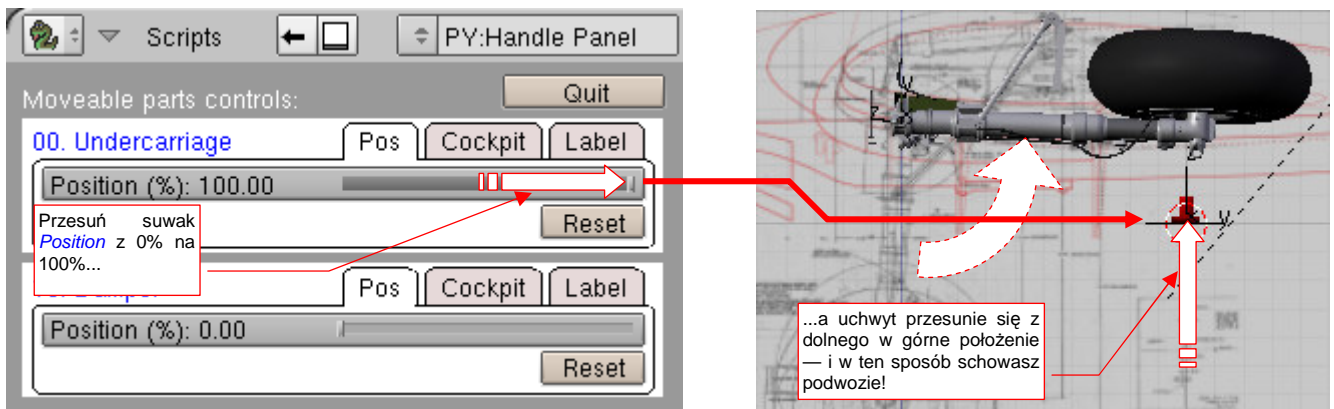
Rysunek 14.30.3 Wywołanie skryptu *Handle Panel*

W rezultacie pojawiają się dwie panele, z których każda odpowiada jednemu z uchwytów (Rysunek 14.30.4):



Rysunek 14.30.4 Panele sterowania modelem

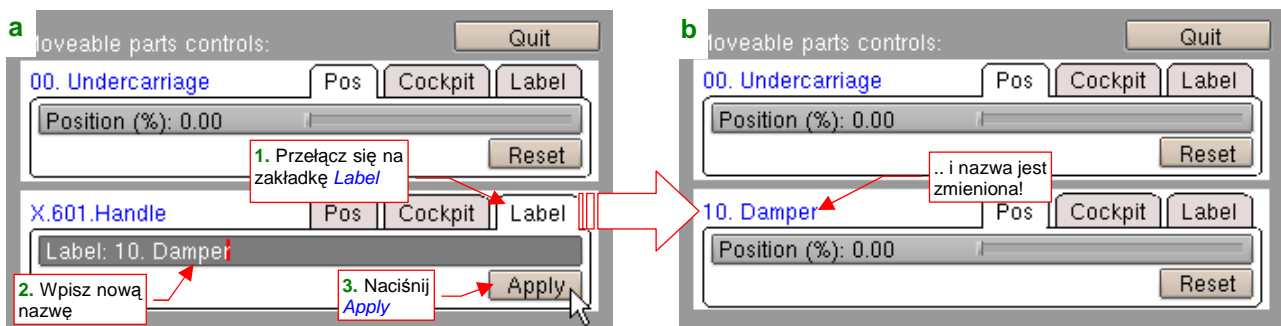
Przypuszczam, że wystarczył Ci rzut oka, by wiedzieć jak posługiwać się tym programem: należy przesunąć suwak **Position**, umieszczony na każdym panelu. (W tę kontrolkę możesz także wpisać dokładną wartość numeryczną). Gdy podwozie jest wysunięte (por. Rysunek 14.30.2), uchwyt **X.600.Handle** jest w najniższym położeniu (0%). Także uchwyt amortyzatora **X.601.Handle** jest w położeniu 0% (najniższym) — gdy amortyzator jest całkowicie rozciągnięty. Wystarczy którykolwiek z tych suwaków przesunąć, by wywołać ruch odpowiedniego mechanizmu (Rysunek 14.30.5):



Rysunek 14.30.5 Zmiana nazwy panelu

Dodatkowo, aby ułatwić ustawienie zespołu w jakiejś wyróżnionej pozycji neutralnej, dodałem przycisk **Reset**. Gdy go naciśniesz, środek uchwytu zostanie przesunięty w miejsce, w którym znajduje się środek jego obiektu — „rodzica” (*Parent*). W przypadku uchwytu z tego przykładu „rodzic” to pomocniczy obiekt **X.600.Base**.

Jak pokazuje Rysunek 14.30.4, domyślna nazwa uchwytu jest zapożyczona z nazwy obiektu. Warto ją zmienić na coś bardziej „reprezentatywnego”. Służy do tego zakładka **Label** (Rysunek 14.30.6):

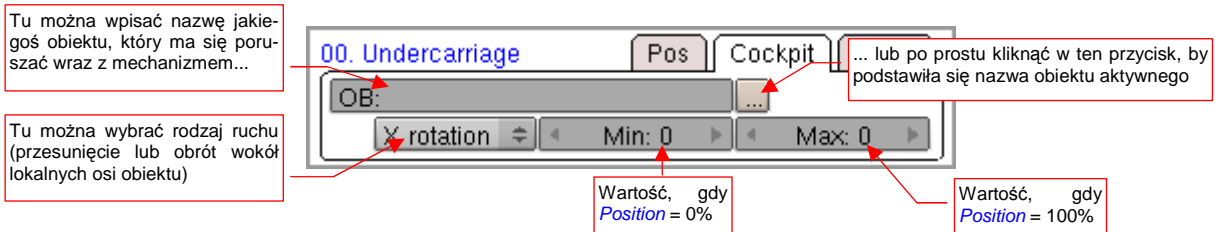


Rysunek 14.30.6 Zmiana nazwy panelu

Po naciśnięciu przycisku **Apply** nazwa, wyświetlana u góry panelu, ulegnie zmianie.

- Skrypt ustawia zawsze panele wg ich nazw, w kolejności alfabetycznej. Warto więc poprzedzić je jakimś numerycznym przedrostkiem, by amortyzator został umieszczony przy podwoziu, a ster wysokości (*Elevator*) przy sterze kierunku (*Rudder*).

Ostatnia zakładka panelu uchwytu — **Cockpit** — służyła do „przełożenia” transformacji mechanizmu na jakiś element w kabinie pilota (stąd nazwa) (Rysunek 14.30.7):



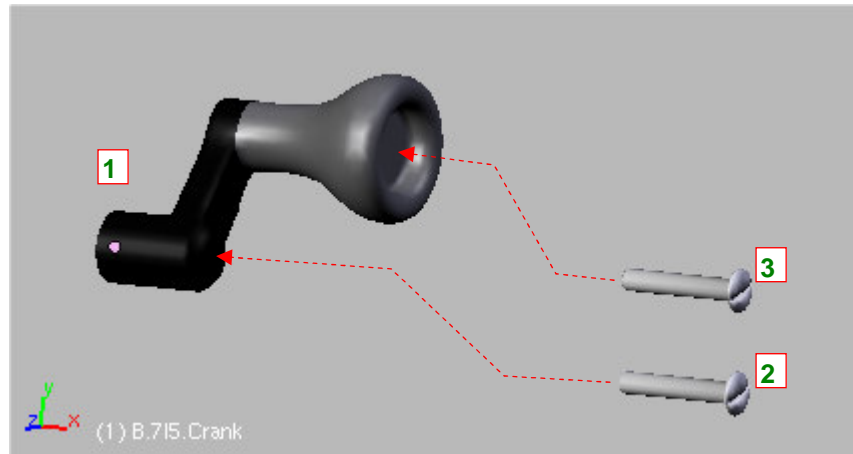
Rysunek 14.30.7 Zakładka **Cockpit**

Zazwyczaj chodziło o to, by np. drążek sterowy wychylał się wraz z ruchem sterów lub lotek. Miało to sens w starszych wersjach Blendera (np. 2.43 i kilku następnych), gdy jeszcze nie istniało ograniczenie *Transform*. Teraz nie ma już chyba potrzeby używać zakładki **Cockpit**. (Adaptując ten skrypt do Blendera 2.5/2.6 na pewno jej nie uwzględnię).

14.31 Dokładne umieszczanie obiektów w zadanym miejscu

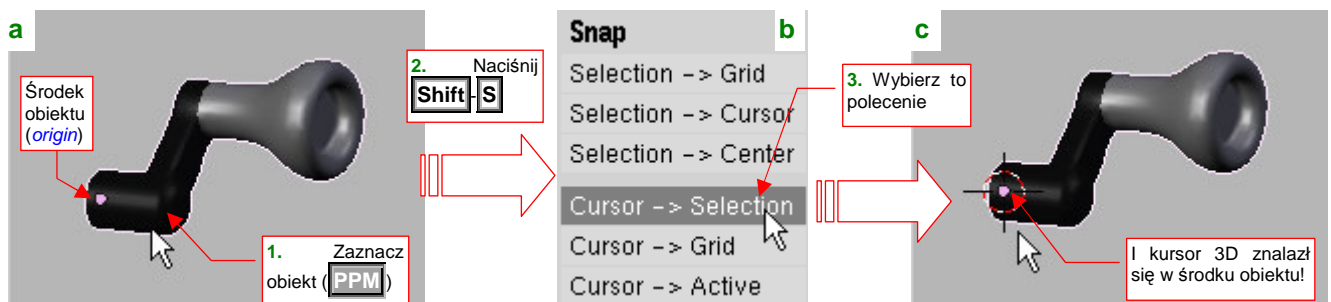
Podczas pracy nad modelem często powstaje sytuacja, gdy jakiś obiekt należy umieścić w dokładnie określonym miejscu. Jego aktualne położenie może być przypadkowe, a w każdym razie nieistotne. Tę czynność wykonuje się dość często wszędzie tam, gdzie modelujemy zespół złożony z wielu części — np. podwozie lub wnętrze kabiny. Ta sekcja pokazuje szybką i dokładną metodę wykonania takiej operacji.

Rysunek 14.31.1 przedstawia typowy przypadek opisanego powyżej problemu. Właśnie sklonowaliśmy dwie śruby (2 i 3). Śruba 2 ma się znaleźć w osi korbki 1, a śruba 3 — w centrum jej rączki. Jak to zrobić najszybciej i najłatwiej?



Rysunek 14.31.1 Typowy problem: jak najszybciej/najłatwiej osadzić śruby w osiach korbki?

Środek (punkt odniesienia — *origin*) korki leży w jej osi obrotu. W tym samym miejscu ma się znaleźć śruba 2. Skorzystajmy z tego faktu i umieśćmy w tym punkcie kursor 3D. Aby to zrobić: zaznacz korkkę (Rysunek 14.31.2a) i naciśnij **Shift-S**, by rozwinąć podręczne menu **Snap** (Rysunek 14.31.2b). Wybierz z niego polecenie **Cursor→Selection**:



Rysunek 14.31.2 Przeniesienia kursora 3D do punktu odniesienia obiektu (*origin*)

W rezultacie, kursor 3D znajdzie się w środku (punkcie odniesienia) korki (Rysunek 14.31.2c).

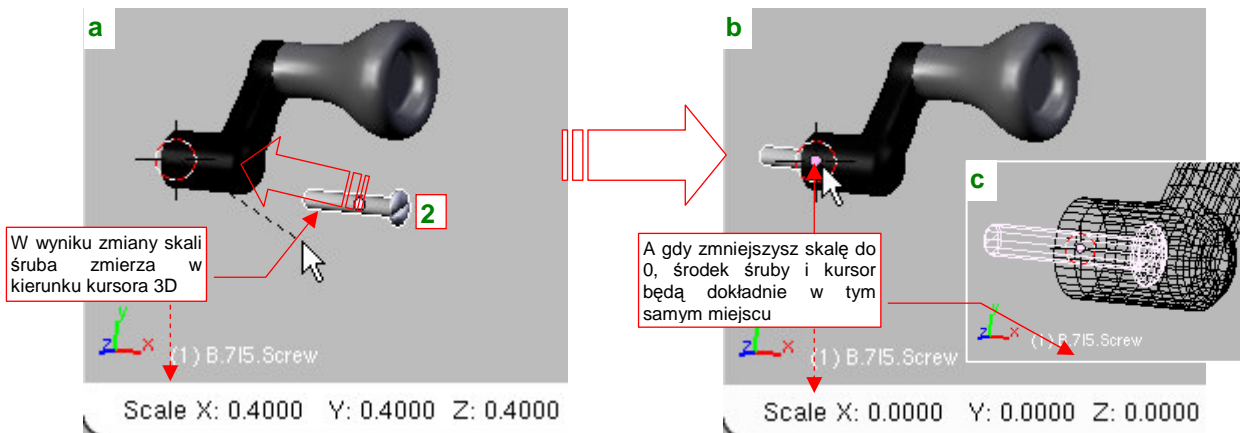
Teraz trzeba jeszcze odpowiednio ustawić tryb transformacji:

- przełącz tryb **Pivot** na **3D Cursor** (Rysunek 14.31.3a),
- włącz **Move object centers only** (Rysunek 14.31.3b):



Rysunek 14.31.3 Ustawienie odpowiednich trybów dla punktu odniesienia i transformacji

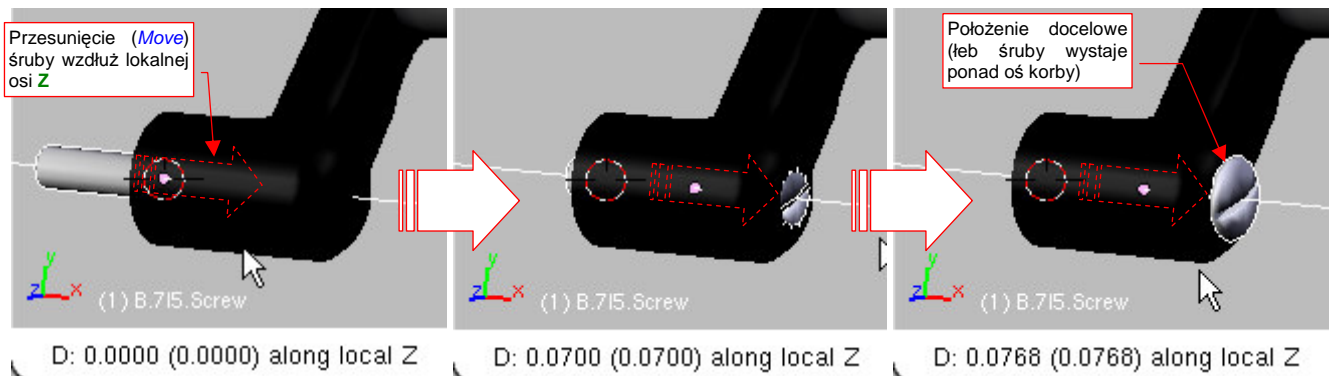
Zaznacz śrubę **2**, i naciśnij **[S]**, rozpoczynając operację zmiany skali. Gdy jest włączony tryb **Move object centers only**, rozmiar śruby nie ulegnie zmianie. Gdy podczas zmiany skali tylko jej środek zacznie zmierzać w kierunku środka transformacji - kursora 3D. I o to chodzi! (Rysunek 14.31.4a):



Rysunek 14.31.4 Umieszczenie śruby w zadanym punkcie (poprzez zmianę skali)

Gdy przesuwając myszkę wciśniesz jednocześnie klawisz **[Ctrl]**, szybko i łatwo zmniejszysz skalę do zera. Dla tej wartości środek śruby **2** znajdzie się dokładnie w tym samym miejscu, co kursor 3D — czyli w osi korbki (Rysunek 14.31.4b). Tyle tylko, że został „zagłębiony” w materiale (Rysunek 14.31.4c).

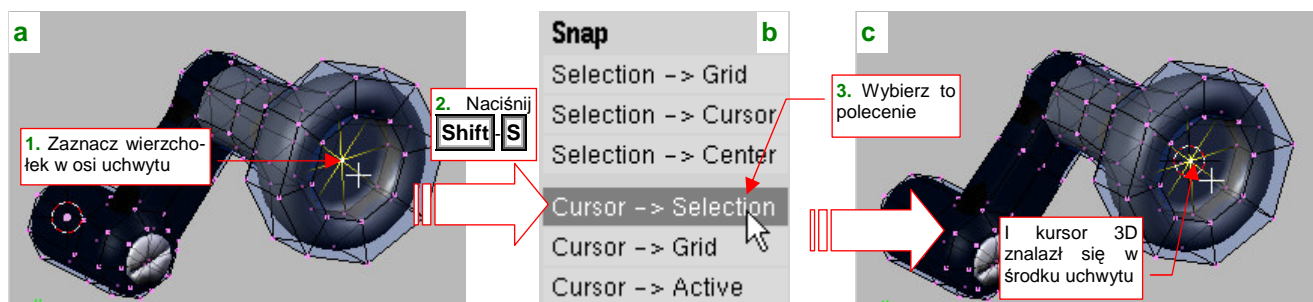
To na szczęście można szybko skorygować: wystarczy przesunąć śrubę wzdłuż jej lokalnej osi **Z** (**[G][Z][Z]**) dopóki jej łeb nie „wynurzy” się z korbki (Rysunek 14.31.5):



Rysunek 14.31.5 Finalne dopasowanie (przesunięcie wzdłuż osi śruby)

Śrubę **3** umieścimy w korbce w ten sam sposób. Musimy tylko umieścić w innym miejscu kursor 3D.

W tym celu przełączymy się na chwilę w tryb edycji siatki (**[Tab]**), i zaznaczymy wierzchołek leżący w centrum uchwyty (Rysunek 14.31.6a):

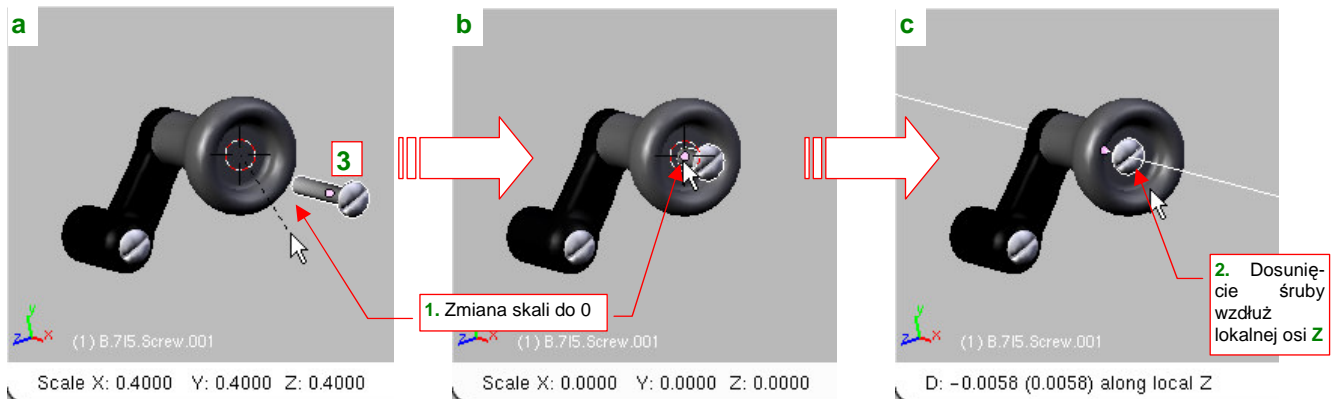


Rysunek 14.31.6 Przeniesienia kursora 3D do środka uchwyty

Potem postępujemy tak samo, jak poprzednio: **[Shift][S]**, by przywołać menu **Snap**, i wybrać z niego polecenie **Cursor->Selection**. W efekcie kursor 3D zostanie umieszczony w podświetlonym wierzchołku.

- Wierzchołki siatki bardzo się przydadzą do ustawienia kursora 3D w różnych kluczowych miejscach. Na przykład, jeżeli nawet nie masz wierzchołka w środku otworu, wystarczy zaznaczyć dwa przeciwległe wierzchołki na jego obwodzie. Polecenie *Cursor*→*Selection* umieści kursor 3D dokładnie pomiędzy nimi.

Gdy mamy już kursor 3D w odpowiednim miejscu, można się znowu przełączyć w *Object Mode* (**Tab**). Potem ze śrubą **3** postępujemy tak samo, jak ze śrubą **2**: zmiana skali do zera (Rysunek 14.31.7a, b):



Rysunek 14.31.7 Umieszczenie śruby w centrum uchwyty

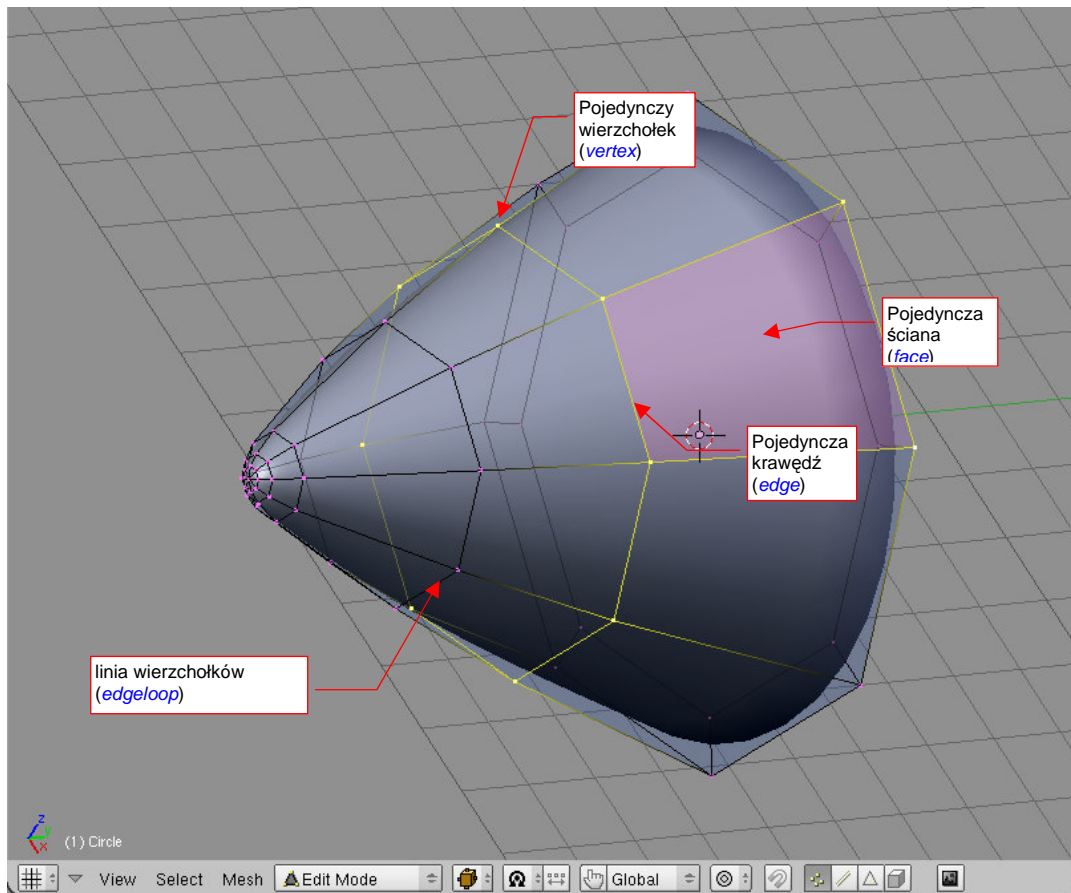
Na koniec dosuwamy łeb śruby do powierzchni korbki, przesuwając ją wzdłuż lokalnej osi **Z**. Rysunek 14.31.7c przedstawia gotową korbkę, z dwoma śrubami osadzonymi we właściwych miejscach.

Rozdział 15. Blender — edytor siatki (Edit Mode)

Prawie każdy obiekt w Blenderze zawiera siatkę wierzchołków. W ten sposób w Blenderze odwzorowany jest kształt. Tryb edycji służy do zmiany siatki pojedynczego obiektu.

15.1 Pojęcia podstawowe

Prawie każdy obiekt Blendera zawiera "w środku" siatkę (*mesh*). Siatka to zbiór elementarnych ścian (*face*), ich krawędzi (*edge*) i wierzchołków (*vertex*). Rysunek 15.1.1 pokazuje przykład takiej siatki:



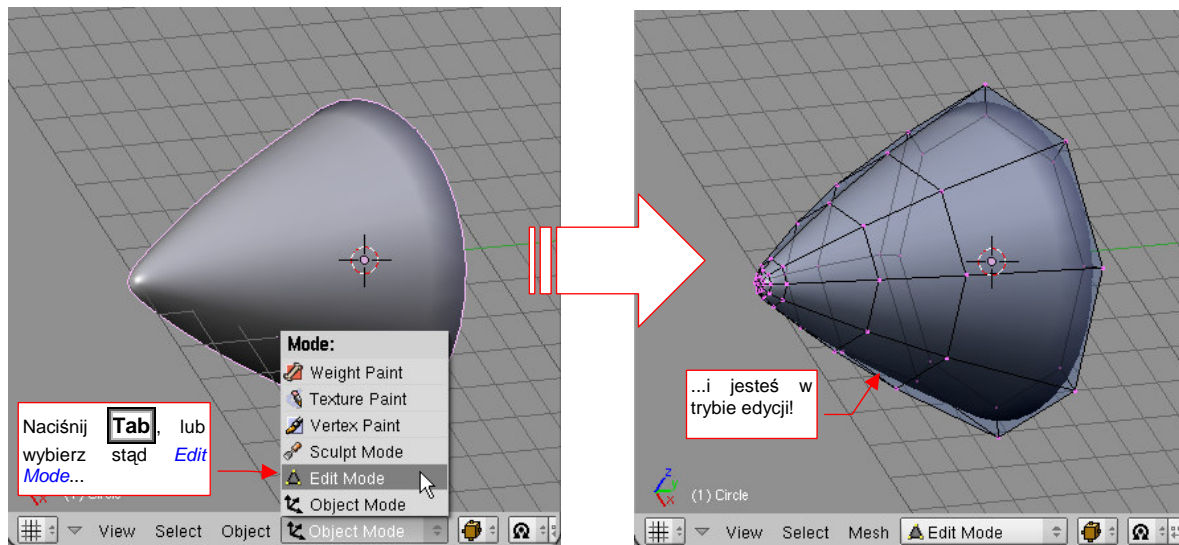
Rysunek 15.1.1 Siatka (*mesh*): elementy

Ściany w Blenderze mogą być kwadratowe lub trójkątne. W szczególnych przypadkach mogą w siatce mogą istnieć krawędzie bez ścian, czy wręcz "osamotnione" wierzchołki bez krawędzi.

W Blenderze występuje także pojęcie "linii wierzchołków" (*edgeloop* — Rysunek 15.1.1). Najłatwiej wskazać je na siatkach o ścianach prostokątnych. Blender potrafi jednak także znaleźć takie linie także na siatkach o ścianach trójkątnych.

15.2 Włączenie trybu edycji (*Edit Mode*)

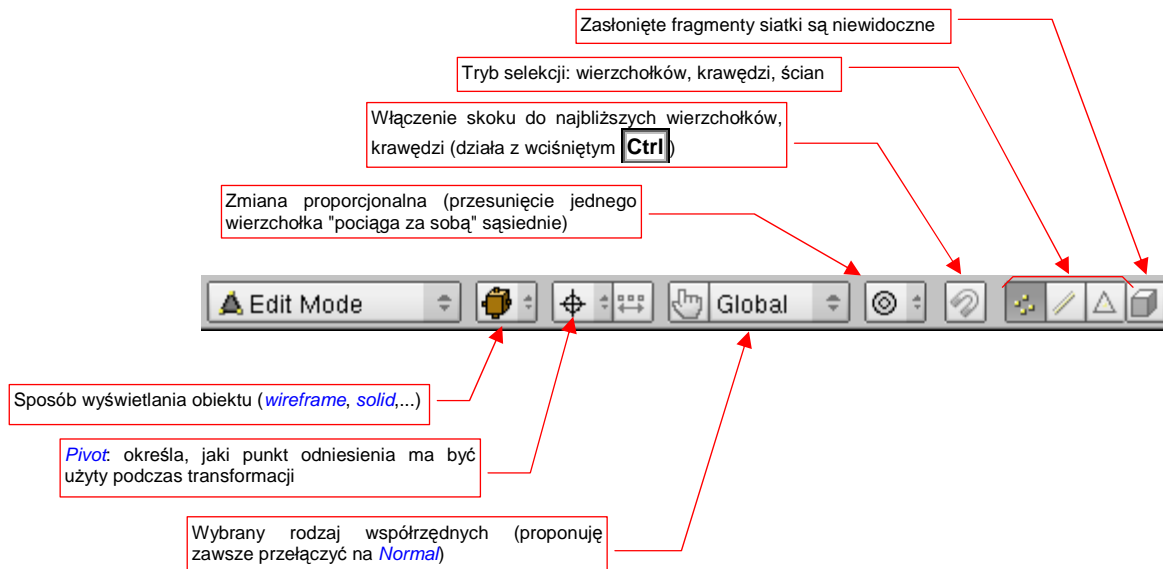
Tryb edycji siatki włączasz naciskając przycisk **Tab**. Alternatywnie możesz także zrobić to za pomocą menu z nagłówka okna widoku (Rysunek 15.2.1):



Rysunek 15.2.1 Okno widoku: włączenie trybu edycji

Powtórne naciśnięcie **Tab** przełączy Cię z powrotem do poprzedniego trybu (np. obiektu — *Object Mode*). Możesz także wybrać inny tryb z nagłówka widoku, listy rozwijalnej *Mode*.

Nagłówek okna *View 3D* ma w trybie edycji wiele przydatnych kontroltek:

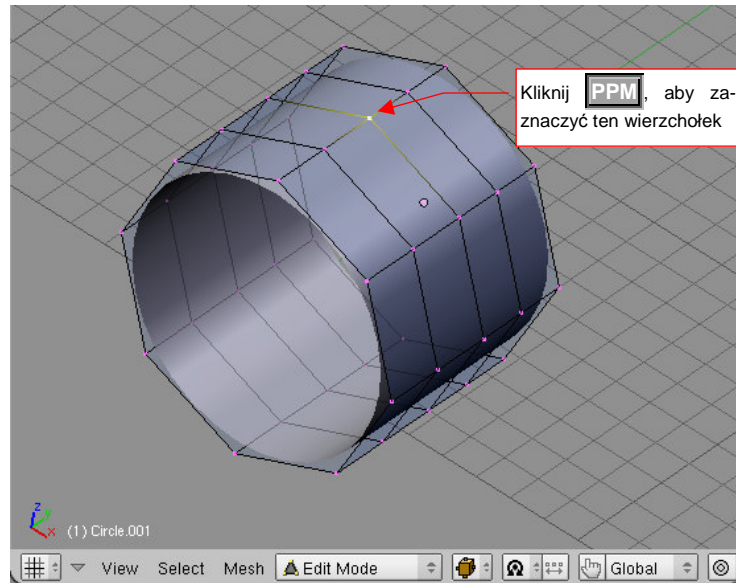


Rysunek 15.2.2 Okno widoku: kontrolki dostępne w trybie edycji

Mankamentem tego nagłówka jest brak możliwości włączania i ukrywania warstw. (Aby to zrobić, musisz się przełączyć w *Object Mode*)

15.3 Zaznaczanie elementów siatki

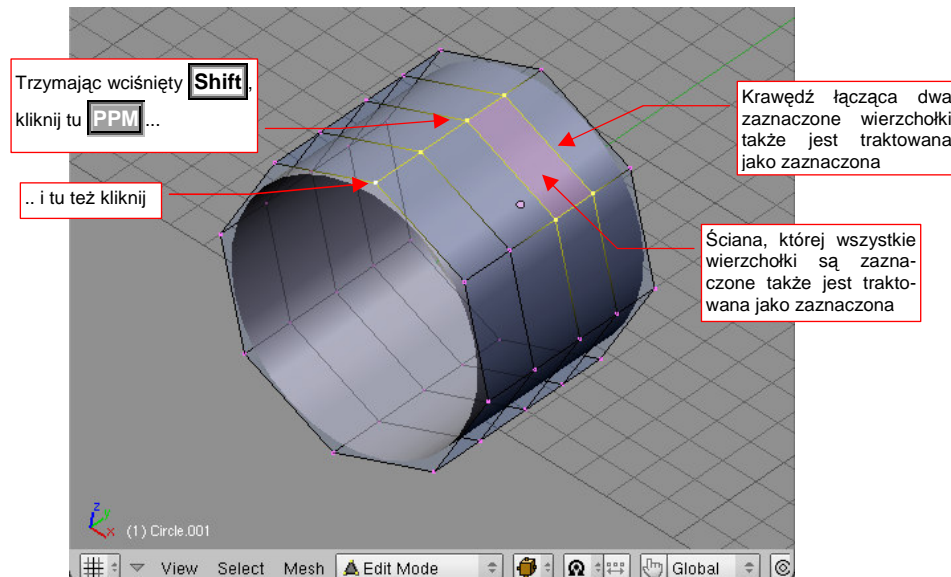
W zasadzie, wierzchołki można zaznaczać tak samo, jak całe obiekty (por. str. 72). Gdy klikniesz w jeden z nich **PPM** — stanie się zaznaczony, i zmieni swój kolor na żółty (Rysunek 15.3.1):



Rysunek 15.3.1 Zaznaczenie pojedynczego wierzchołka

Zwróć uwagę, że kolor krawędzi, wychodzących z zaznaczonego wierzchołka także ulega stopniowej zmianie. Im bliżej zaznaczenia, tym bardziej są żółte. Taka konwencja ułatwia szybkie odnalezienie zaznaczonego na siatce obszaru.

Aby **wybrać wiele wierzchołków**, jeden po drugim: trzymaj wciśnięty klawisz **Shift** i klikaj w nie po kolei **PPM** (Rysunek 15.3.2):

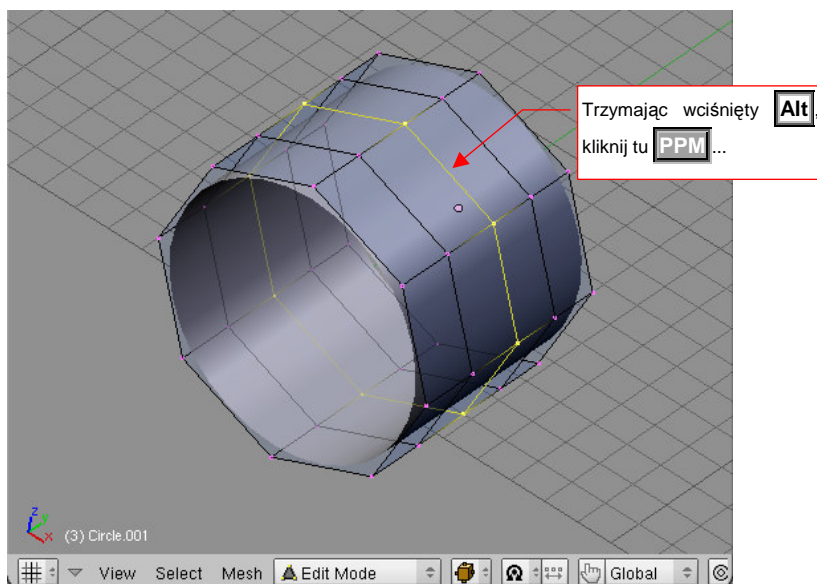


Rysunek 15.3.2 Wybór wielu wierzchołków

Kombinacja **Shift** - **PPM** działa jak przełącznik stanu zaznaczenia: pierwsze kliknięcie w wierzchołek włącza go do wybranych wierzchołków. Następne kliknięcie — wyklucza z tego zbioru.

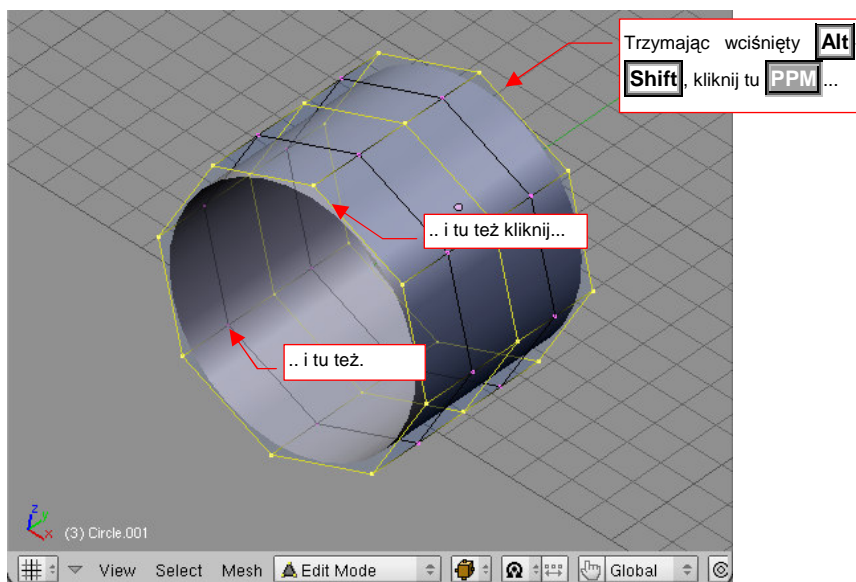
- Jeżeli chcesz szybko zrezygnować z aktualnego zaznaczenia — naciśnij klawisz **A** (*Select* → *Select/Deselect All*). Jeżeli chcesz zaznaczyć wszystko — naciśnij ten klawisz powtórnie. (Polecenie to działa jak przełącznik: "wszystko - nic").

Na siatkach Blendera można także szybko **wskazać cały rząd wierzchołków**, tworzących topologicznie linię (*edgeloop*). Trzymając wciśnięty **Alt**, kliknij **PPM** w jedną z krawędzi, należącą do tej linii (Rysunek 15.3.3):



Rysunek 15.3.3 Zaznaczanie linii wierzchołków

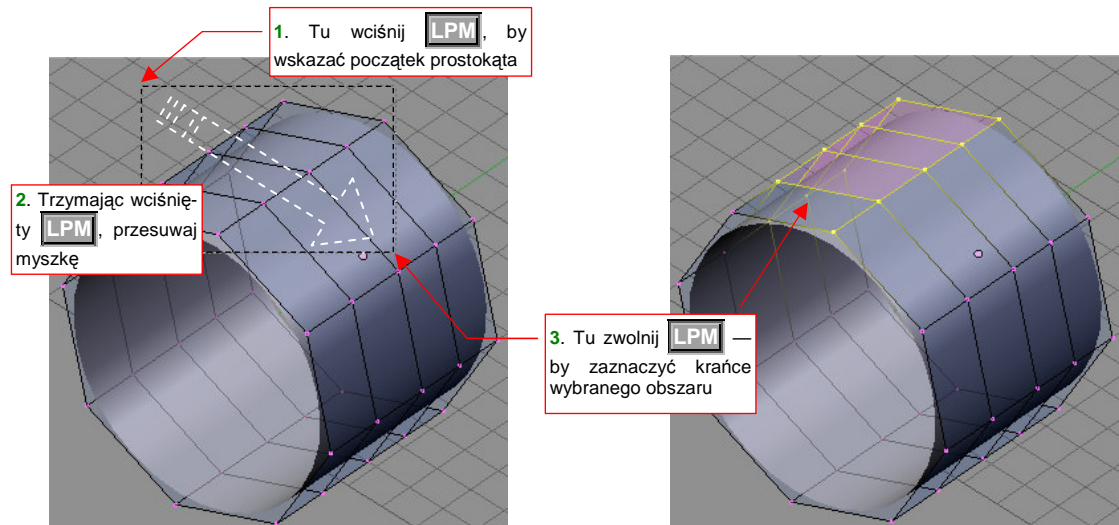
Analogicznie, jak w przypadku pojedynczych wierzchołków, możesz wyłączać/wykluczać z selekcji całe linie. Trzymaj na klawiaturze wciśnięte **Alt** - **Shift**, i klikaj w krawędzie odpowiednich linii **PPM** (Rysunek 15.3.4):



Rysunek 15.3.4 Zaznaczanie wielu linii wierzchołków

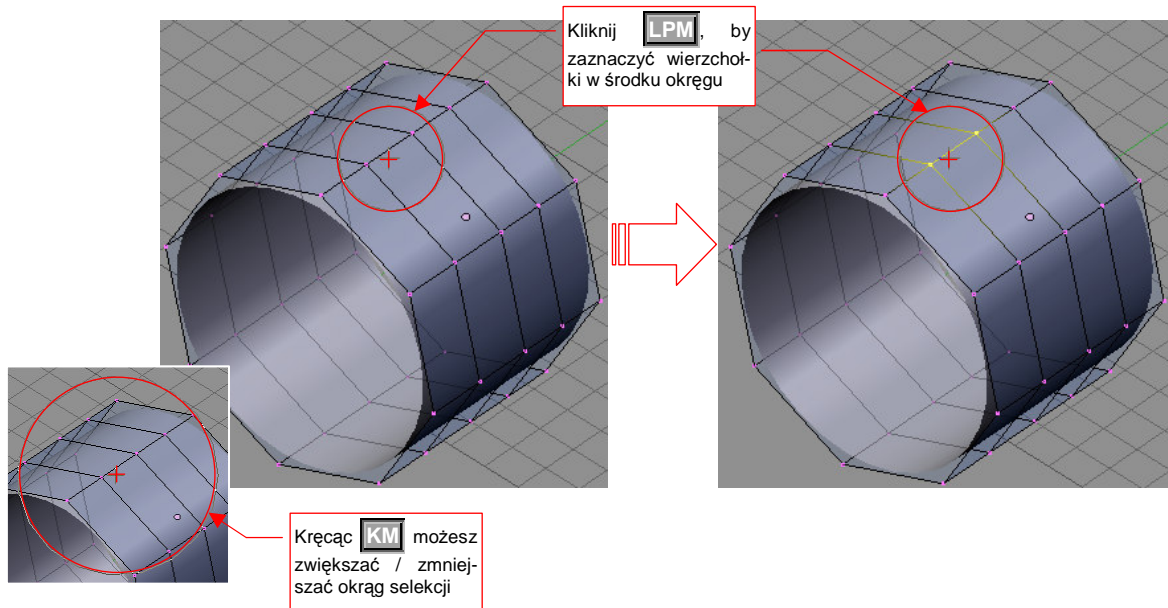
- Możesz także odwrócić selekcję (wybrać to, co nie jest zaznaczone): naciśnij **I** (*Select* → *Inverse*)

Wreszcie — w Blenderze jest także dostępne **wybijanie obszarem prostokątnym**. Naciśnij **B** (*Select*→*Border select*) (Rysunek 15.3.5).



Rysunek 15.3.5 Zaznaczanie obszarem prostokątnym

Osobiście wolę jednak posługiwać się inną odmianą wyboru obszarem: **wyborem okręgiem**. Aby przełączyć się w ten tryb, naciśnij powtórnie klawisz **B**. (Czyli — aby włączyć go "od zera", musisz nacisnąć **B** dwa razy. Możesz to zrobić szybko, traktując tę kombinację jako skrót — **B, B**). Wokół kursora na ekranie pojawi się okrąg (Rysunek 15.3.6):

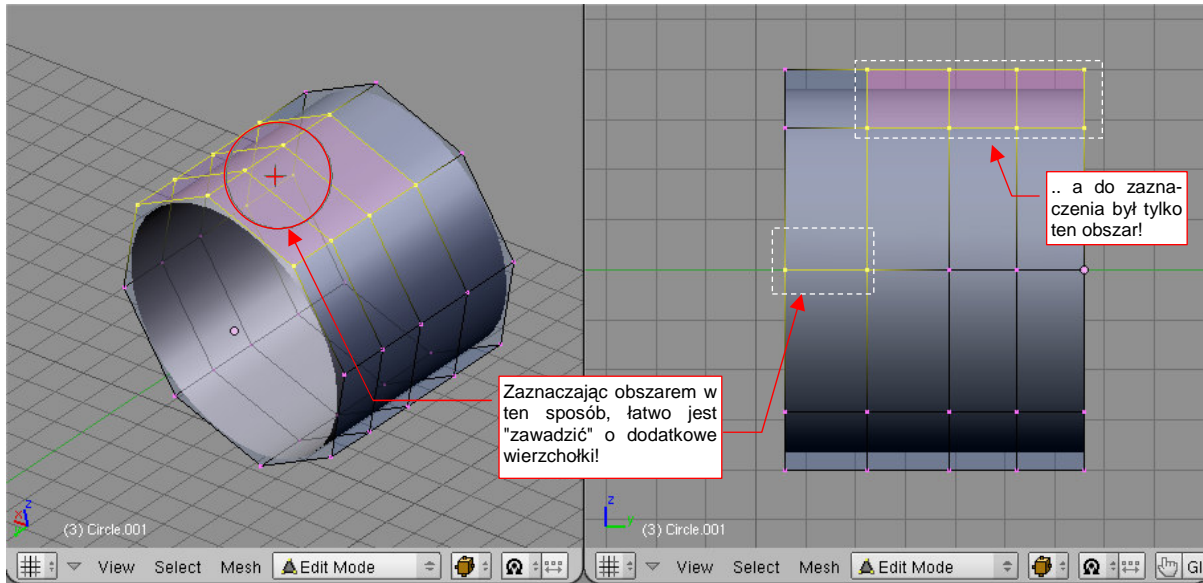


Rysunek 15.3.6 Zaznaczanie okręgiem

W tym trybie, gdy klikniesz **LPM**, zaznaczysz wierzchołki, które są wewnątrz okręgu. Rozmiar okręgu możesz zwiększać lub zmniejszać, obracając **KM**. Klikając w różne miejsca, możesz włączać kolejne wierzchołki do zbioru selekcji.

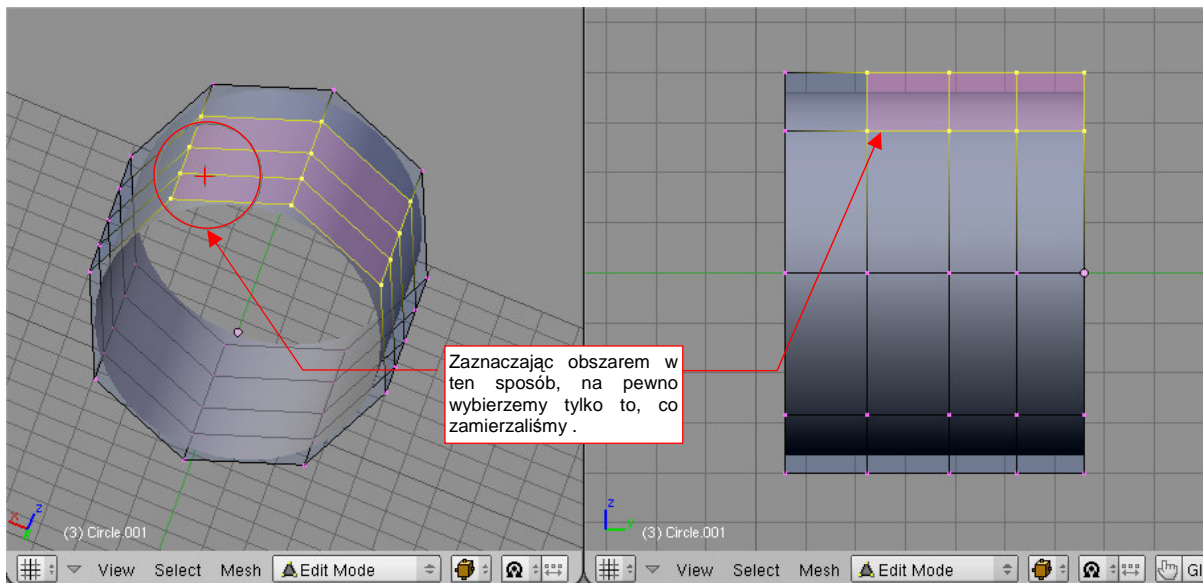
Jeżeli chciałbyś coś z tego zbioru wyłączyć — podczas kliknięcia trzymaj wciśnięty klawisz **Alt**. Tryb wyboru okręgiem możesz zakończyć, naciskając **Esc**.

Na koniec chciałbym przypomnieć, jak ważne podczas zaznaczania jest właściwe dobranie widoku. Rysunek 15.3.7 pokazuje, co się może stać, gdy zaznaczamy obszarem wierzchołki, za którymi są jeszcze dalsze punkty siatki:



Rysunek 15.3.7 Zaznaczanie obszarem — projekcja dobrana niewłaściwie

Rysunek 15.3.8 pokazuje, jak wygląda projekcja lepiej dobrana do tej sytuacji. Gdy nic nie leży poza zaznaczanymi wierzchołkami, nie ma możliwości zaznaczenia czegokolwiek dodatkowego:



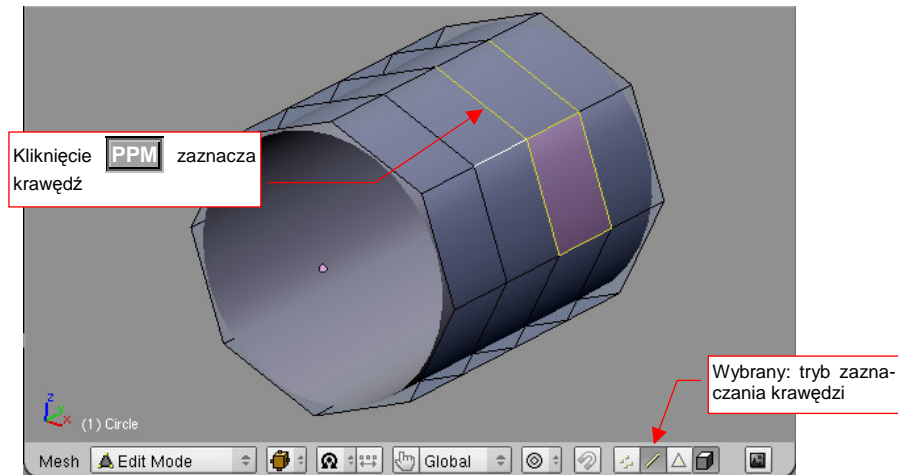
Rysunek 15.3.8 Zaznaczanie obszarem — projekcja dobrana właściwie

Do tej pory omówiłem szczegółowo tryb wybierania wierzchołków. Ale na tym możliwości programu się nie kończą. W nagłówku okna widoku znajduje się odpowiedni przełącznik (Rysunek 15.3.9):



Rysunek 15.3.9 Włączony tryb selekcji wierzchołków w nagłówku okna [View 3D](#)

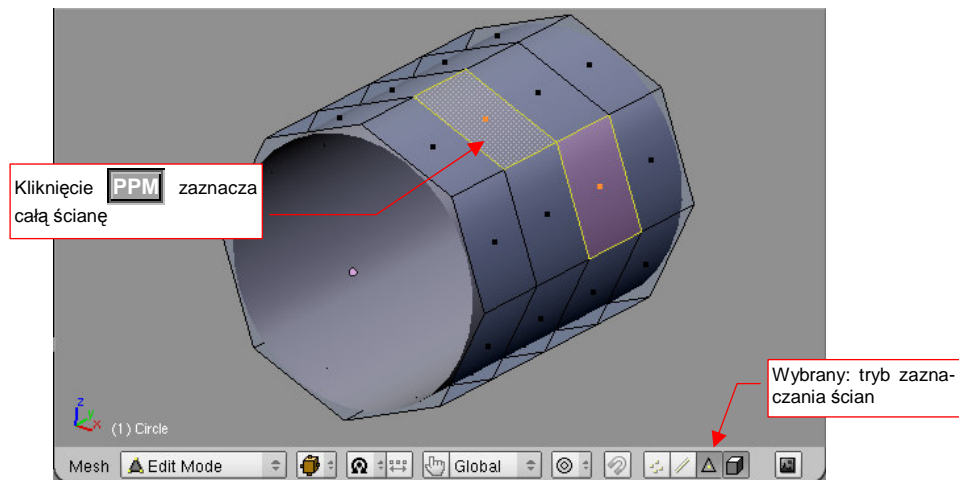
Gdy włączysz tryb zaznaczania krawędzi, zmieni się sposób wyświetlania siatki (Rysunek 15.3.10):



Rysunek 15.3.10 Tryb zaznaczania krawędzi

Tryb zaznaczania krawędzi jest przydatny np. wtedy, gdy chcesz usunąć ich grupę bez usunięcia wierzchołków. W trybie zaznaczania wierzchołków masz dwa razy więcej do "klikania"! W każdym trybie selekcji możesz stosować wszystkie metody zaznaczania, które opisałem na poprzednich stronach.

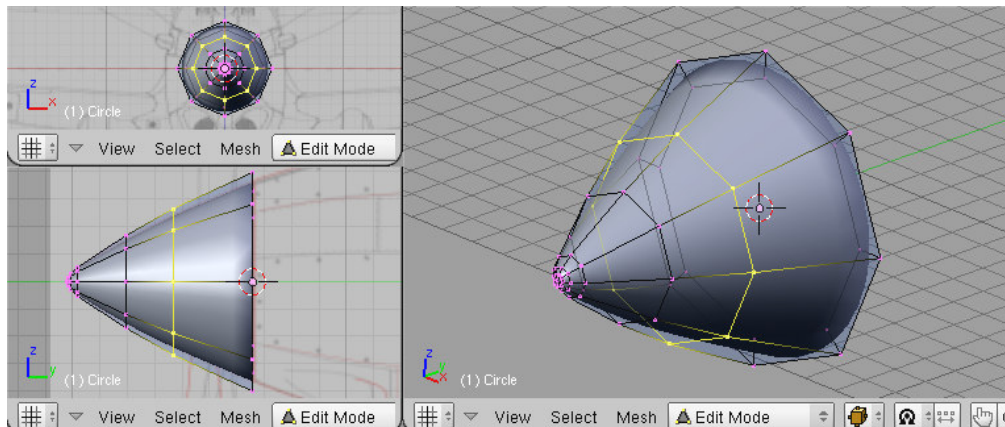
Ostatnim trybem zaznaczania jest zaznaczanie całych ścian (Rysunek 15.3.11):



Rysunek 15.3.11 Tryb zaznaczania ścian

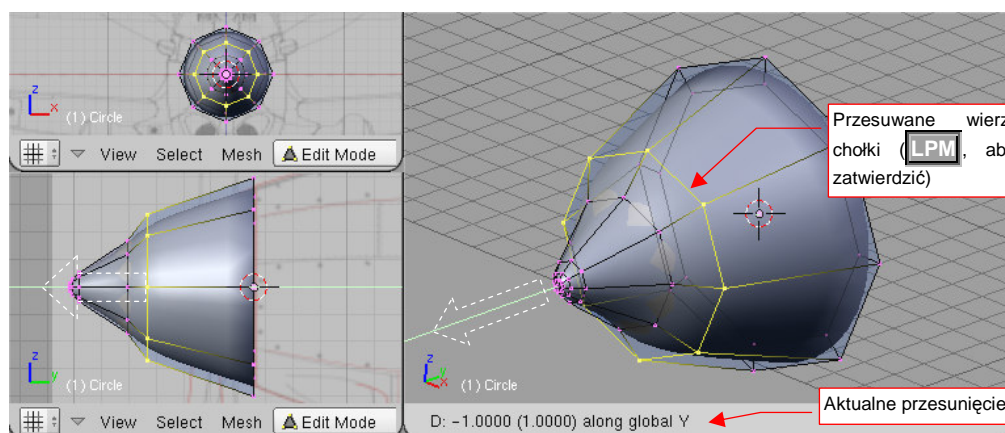
15.4 Przesunięcie (*Grab*)

Zaznacz wierzchołki, które chcesz przesunąć (Rysunek 15.4.1):



Rysunek 15.4.1 Zaznaczenie wierzchołków

Następnie naciśnij **G** (*Mesh* → *Transform* → *Grab/Move*). Przełączysz się w ten sposób w tryb przesuwania. Każde przesunięcie myszy będzie powodować przemieszczenie zaznaczonych wierzchołków. Dokonywane zmiany w kształcie siatki są pokazywane na bieżąco we wszystkich oknach Blendera (Rysunek 15.4.2):

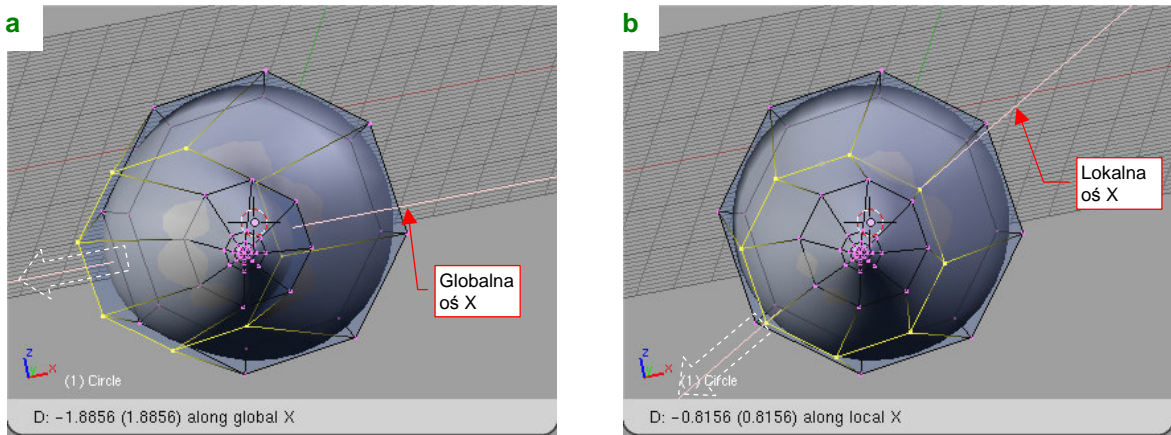


Rysunek 15.4.2 Przesuwanie grupy wierzchołków

W nagłówku aktywnego okna jest wyświetlane aktualne przesunięcie. Aby zatwierdzić nowe położenie (zakończyć przesunięcie), należy kliknąć **LPM**. Aby zrezygnować z operacji, możesz w każdej chwili nacisnąć **Esc**. Podczas przesuwania możesz dodatkowo trzymać wciśnięty klawisz **Ctrl**. Spowoduje to skokową zmianę odległości (np. co 0.1 jednostki). Jeżeli będziesz trzymał wciśnięty klawisz **Shift** — przesunięcia na ekranie "wyhamują", pozwalając na dokładniejsze ustalenie pozycji. Wreszcie kombinacja obydwu — **Shift-Ctrl** — umożliwi skokowe przesuwanie położenia o niewielkie odległości (np. co 0.01 jednostki).

Jeżeli w trakcie przesuwania naciśniesz **X**, **Y**, lub **Z** — ograniczysz swój ruch tylko do translacji wzdłuż jednej z tych osi. Po pierwszym naciśnięciu klawisza jest to oś globalna. Jeżeli jednak jeszcze raz naciśniesz ten sam klawisz — przełączysz się na lokalną oś obiektu. (Odpowiada to poleceniom *Mesh* → *Transform* → *Grab/Move on Axis*). To rozróżnienie pomiędzy osią globalną i lokalną ma swój sens wtedy, gdy obiekt jest w jakiś sposób obrocony.

Rysunek 15.4.3 pokazuje przykład takiej sytuacji. Stożek został najpierw (jako obiekt) obrócony o 30° w lewo. Następnie przełączyłem się w tryb edycji, zazaczyłem linię siatki, i nacisnąłem **G**, **X**. W efekcie mogę przesunąć wierzchołki wzdłuż globalnej osi X — na bok (Rysunek 15.4.3a). Jeżeli nacisnę jeszcze raz **X** — przełączę się na lokalną oś X obiektu (Rysunek 15.4.3b):

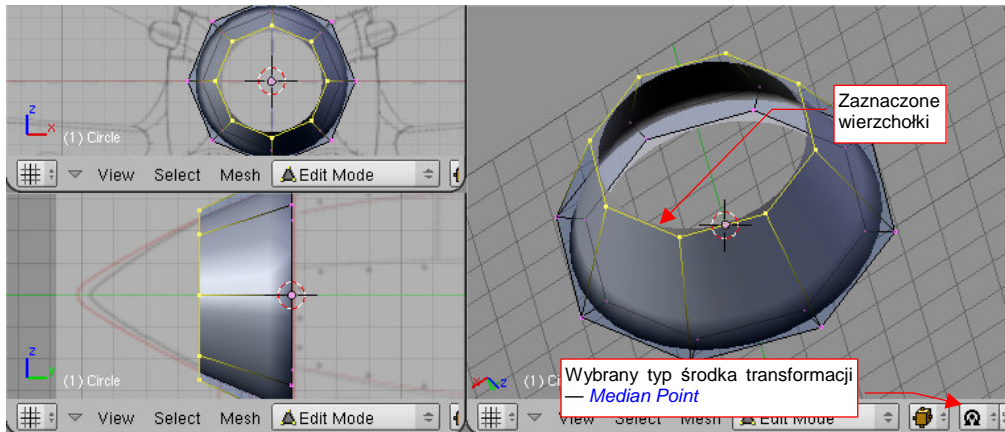


Rysunek 15.4.3 Przesuwanie wzdłuż globalnej osi X (**X**) i lokalnej osi X (**X**, **X**)

W związku z tym, że obiekt został wcześniej obrócony, jego lokalna oś X jest pochylona. Przesunięcie "na bok" zaznaczonej linii siatki oznacza w tym przypadku przemieszczenie pod kątem!

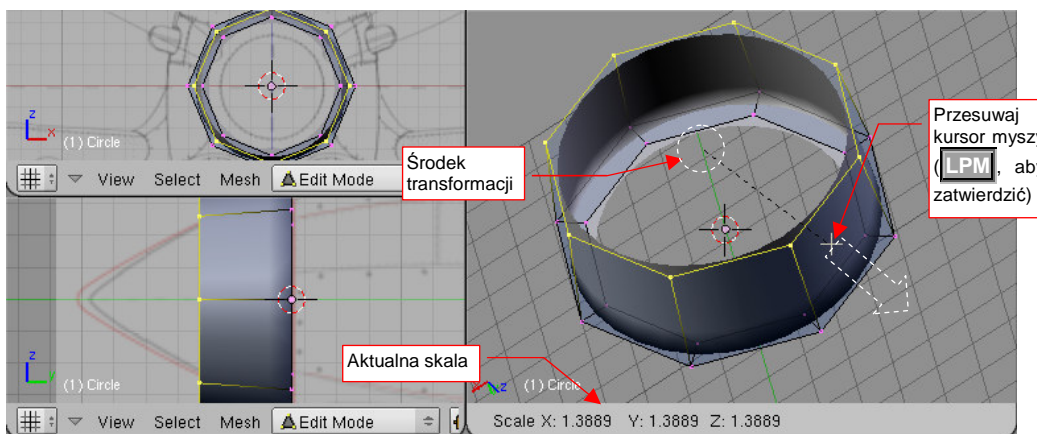
15.5 Skalowanie (**Scale**)

Zaznacz grupę wierzchołków, które chcesz powiększyć/pomniejszyć, oraz ustal środek transformacji (z listy **Pivot** — Rysunek 15.5.1). Praktycznie stosuję dwa — trzy rodzaje środków: **Bounding Box Center**, **Median Point** oraz **Cursor 3D**. **Median Point** to wypadkowy środek wszystkich zaznaczonych wierzchołków. (W przypadku trójkąta leży w "środku ciężkości" tej figury). Zaczniemy od przypadku, gdy środkiem skalowania jest właśnie **Median Point**. (Możesz go ustawić z klawiatury — **Shift-I**)



Rysunek 15.5.1 Zaznaczenie wierzchołków

Naciśnij **S** (**Mesh → Transform → Scale**). Przełączysz się w ten sposób w tryb skalowania. Na rysunku pojawi się kreskowana linia od kursora myszy do środka transformacji. Każde przesunięcie myszy będzie powodować zmianę skali zaznaczonych wierzchołków — proporcjonalnie do zmiany długości kreskowanej linii. Dokonywane zmiany w kształcie siatki są pokazywane na bieżąco we wszystkich oknach Blendera (Rysunek 15.5.2):



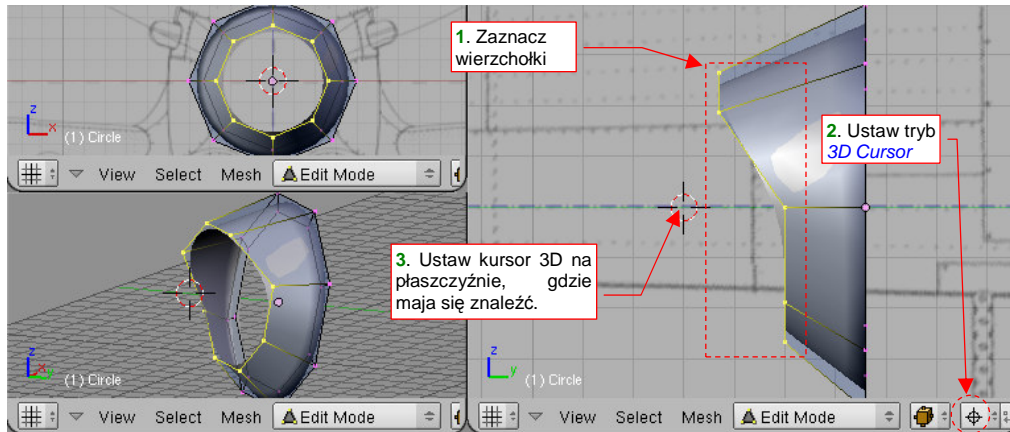
Rysunek 15.5.2 Powiększanie grupy wierzchołków

W nagłówku aktywnego okna jest wyświetlana aktualna zmiana skali wzdłuż każdej osi. Aby zatwierdzić nowe położenie wierzchołków (zakończyć transformację), należy kliknąć **LPM**. Aby zrezygnować z operacji, możesz w każdej chwili nacisnąć **Esc**.

Podczas zmiany skali możesz dodatkowo trzymać wciśnięty klawisz **Ctrl**. Spowoduje to skokową zmianę proporcji (np. co 0.1). Jeżeli będziesz trzymał wciśnięty klawisz **Shift** — zmiana skali na ekranie "wyhamuje", pozwalając na dokładniejsze ustalenie. Wreszcie kombinacja obydwu — **Shift-Ctrl** — umożliwi skokową zmianę skali o niewielkie wartości (np. co 0.01).

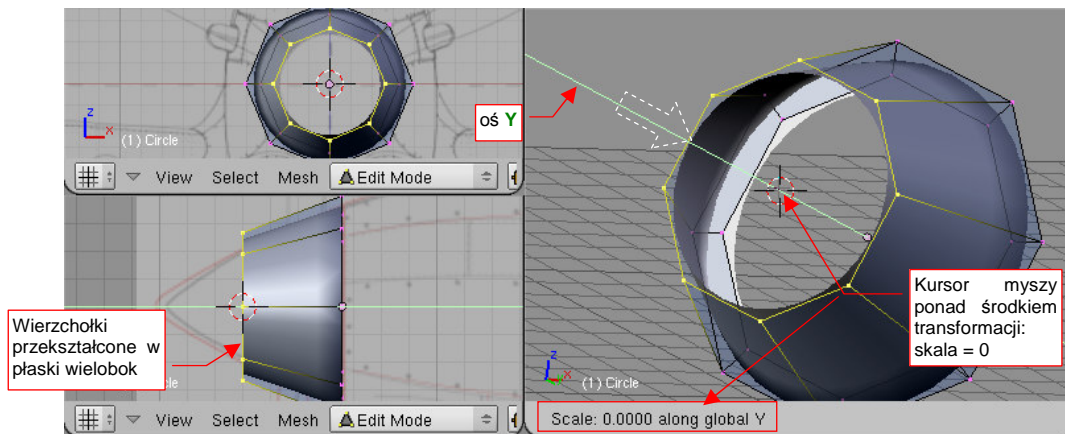
Jeżeli w trakcie przesuwania naciśniesz **X**, **Y**, lub **Z** — ograniczysz zmianę skali do jednej z tych osi. W pozostałych kierunkach żaden rozmiar nie ulegnie zmianie. Po pierwszym naciśnięciu klawisza jest wybrana oś globalna. Jeżeli jednak jeszcze raz naciśniesz ten sam klawisz — przełączysz się na lokalną oś obiektu. (Odpowiada to poleceniom *Mesh→Transform→Scale on Axis*). To rozróżnienie pomiędzy osią globalną i lokalną ma swój sens wtedy, gdy obiekt jest w jakiś sposób obrócony.

Rysunek 15.5.3 pokazuje przykład skalowania wzdłuż jednej osi. Wybrałem tu także inny sposób określenia środka — tym razem jest to *3D Cursor*. Umieściłem go na płaszczyźnie, gdzie chcę umieścić wszystkie wybrane wierzchołki. Przełączyłem typ punktu odniesienia na *3D Cursor* (na klawiaturze: **.**):



Rysunek 15.5.3 Zmiana skali względem kursora 3D — przygotowanie

Nacisnąłem **S**, **Y**. Trzymając wciśnięty klawisz **Ctrl** (aby uzyskać dokładną zmianę skali) zmniejszyłem odległość wierzchołków od środka skalowania do zera (Rysunek 15.5.4):

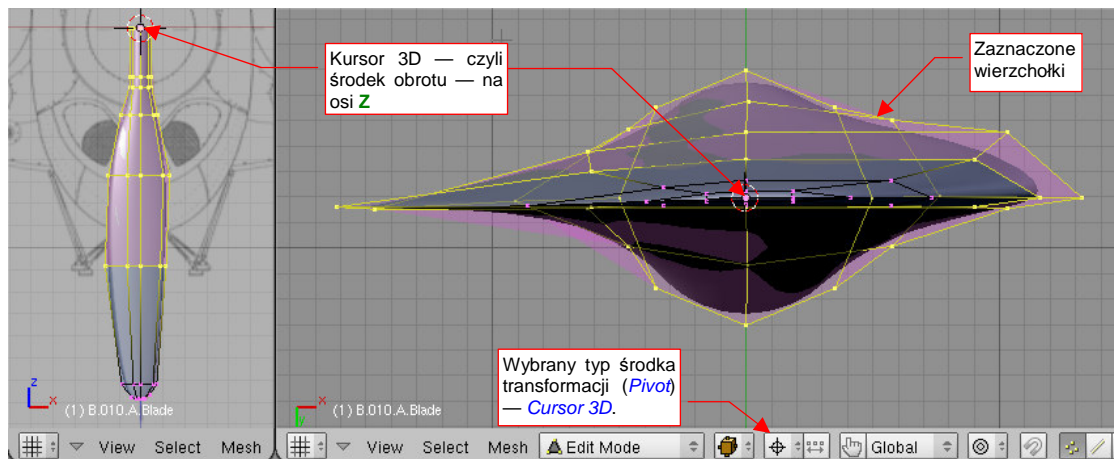


Rysunek 15.5.4 Zmiana skali względem kursora 3D (i wzdłuż jednej osi)

Możesz stosować pokazaną powyżej zmianę skali wzdłuż jednej osi, do zera, jako sposób na dokładne i szybkie "spłaszczenie" grupy wierzchołków i umieszczenie ich w pożądanym miejscu. Taka operacja dość często się przydaje w czasie modelowania samolotu.

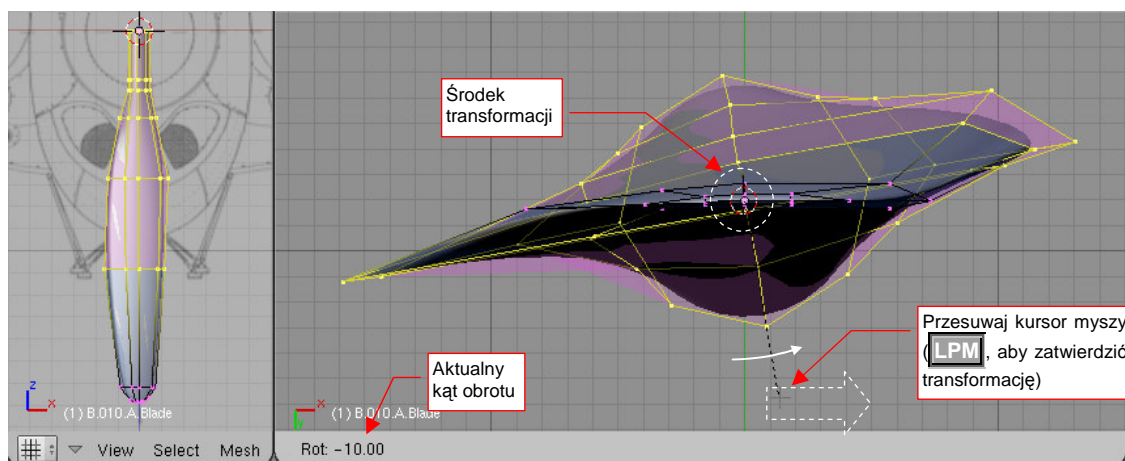
15.6 Obrót (*Rotate*)

Zaznacz grupę wierzchołków, które chcesz obrócić, oraz ustal środek transformacji (*Pivot*) (Rysunek 15.6.1). Praktycznie stosujemy jeden z dwóch rodzajów środka: *Bounding Box Center* albo *Cursor 3D*. W tym przypadku, środkiem obrotu będzie *Cursor 3D*. Możesz go ustawić z klawiatury — naciśnij **[C]** (kropka):



Rysunek 15.6.1 Zaznaczenie wierzchołków

Wybierz okno 3D, którego płaszczyzna widoku ma być płaszczyzną obrotu. (Kursor powinien być widoczny gdzieś w środku tego okna.) Naciśnij **[R]** (*Mesh* → *Transform* → *Rotate*). Przełączysz się w ten sposób w tryb obrotu. Na rysunku pojawi się kreskowana linia od kursora myszy do środka transformacji. Każde przesunięcie myszy będzie powodować obrót zaznaczonych wierzchołków — proporcjonalnie do zmiany kąta kreskowanej linii. Dokonywane zmiany w kształcie siatki są pokazywane na bieżąco we wszystkich oknach Blendera (Rysunek 15.6.2):



Rysunek 15.6.2 Obrót grupy wierzchołków

W nagłówku aktywnego okna jest wyświetlany aktualny kąt obrotu. (Domyślnie obrót następuje w płaszczyźnie aktywnego widoku). Aby zatwierdzić nowe położenie wierzchołków (zakończyć transformację), należy kliknąć **[LPM]**. Aby zrezygnować z operacji, możesz w każdej chwili nacisnąć **[Esc]**.

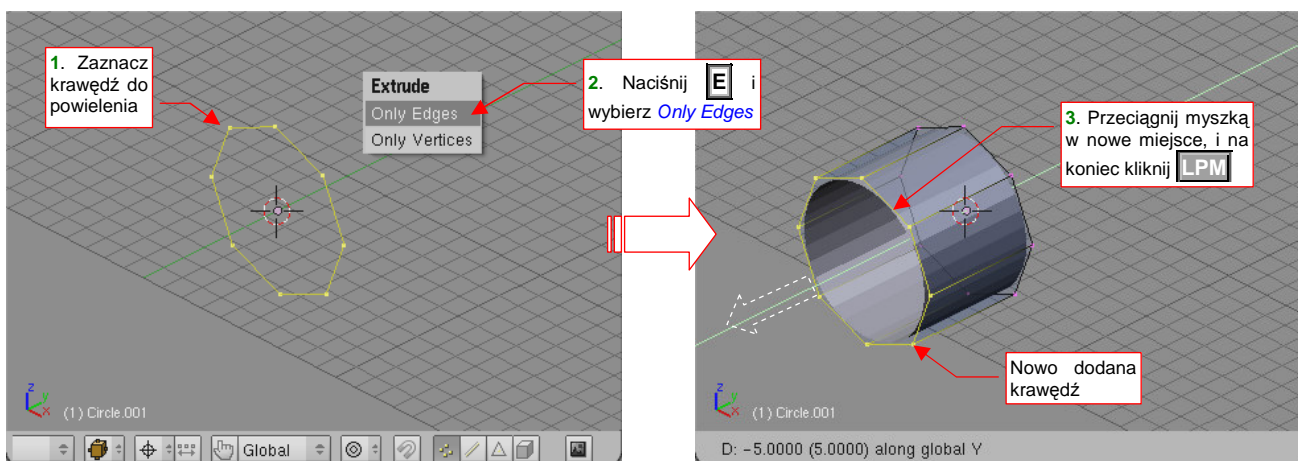
Podczas obrotu możesz dodatkowo trzymać wciśnięty klawisz **[Ctrl]**. Spowoduje to skokową zmianę proporcji (co 5°). Jeżeli będziesz trzymał wciśnięty klawisz **[Shift]** — zmiana obrót na ekranie "wyhamuje", pozwalając na dokładniejsze ustalenie. Wreszcie kombinacja obydwu — **[Shift-Ctrl]** — umożliwia skokową zmianę obrotu o niewielkie kąty (np. co 0.5°).

Jeżeli w trakcie obrotu naciśniesz **X**, **Y**, lub **Z** — przełączysz się na obrót wokół jednej z tych osi. Po pierwszym naciśnięciu klawisza jest wybrana oś globalna. Jeżeli jednak jeszcze raz naciśniesz ten sam klawisz — przełączysz się na lokalną oś obiektu. (Odpowiada to poleceniom *Mesh* → *Transform* → *Rotate on Axis*). To rozróżnienie pomiędzy osią globalną i lokalną ma swój sens wtedy, gdy obiekt jest w jakiś sposób obrócony.

15.7 Wyłaczanie (*Extrude*)

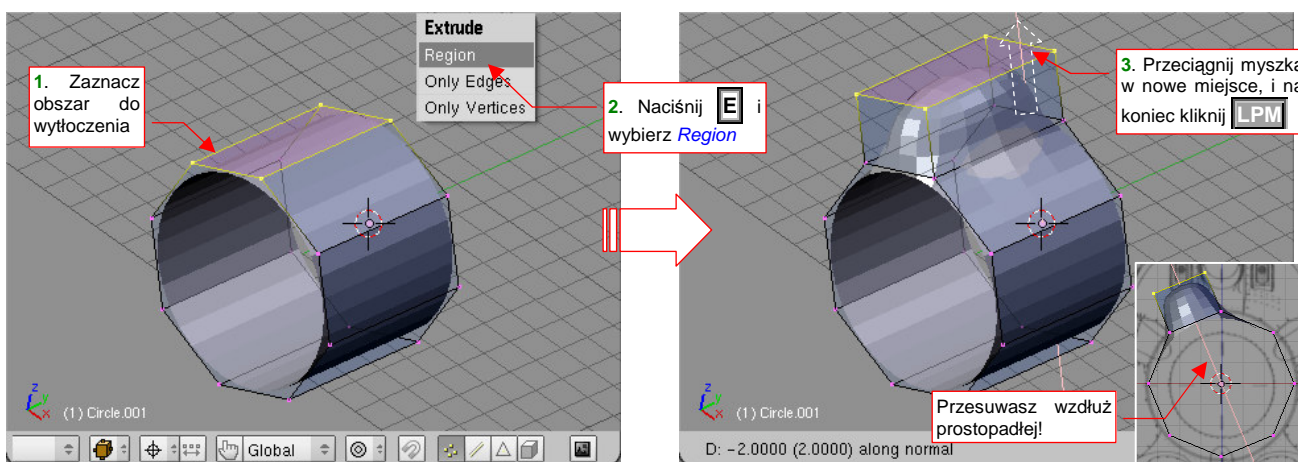
Przełącz się w tryb edycji (**Tab**).

Aby **wyłoczyć wybraną krawędź**, zaznacz ją. Następnie naciśnij **E** (*Mesh*→*Extrude*). Z menu rozwijalnego, które się pojawi, wybierz **Only Edges**. W tym momencie została utworzona kopia zaznaczonej krawędzi. Kopia ta będzie podążać za ruchem myszki, zupełnie jak podczas przesuwania. "Ciągnąć" za sobą będzie krawędzie poprzeczne, łączące jej wierzchołki z wierzchołkami oryginału. Kliknij **LPM** w miejscu, gdzie nowa krawędź ma zostać ostatecznie umieszczona (Rysunek 15.7.1):



Rysunek 15.7.1 Wyłaczanie krawędzi

Aby **wyłoczyć wybrany obszar**, zaznacz go. Następnie naciśnij **E** (*Mesh*→*Extrude*). Z menu rozwijalnego, które się pojawi, wybierz **Region**. W tym momencie wzdłuż krawędzi zaznaczonego obszaru zostały dodane dodatkowa ściany. Sam obszar będzie podążać za ruchem myszki, zupełnie jak podczas przesuwania. Kliknij **LPM** w miejscu, gdzie ma zostać ostatecznie umieszczony (Rysunek 15.7.2):



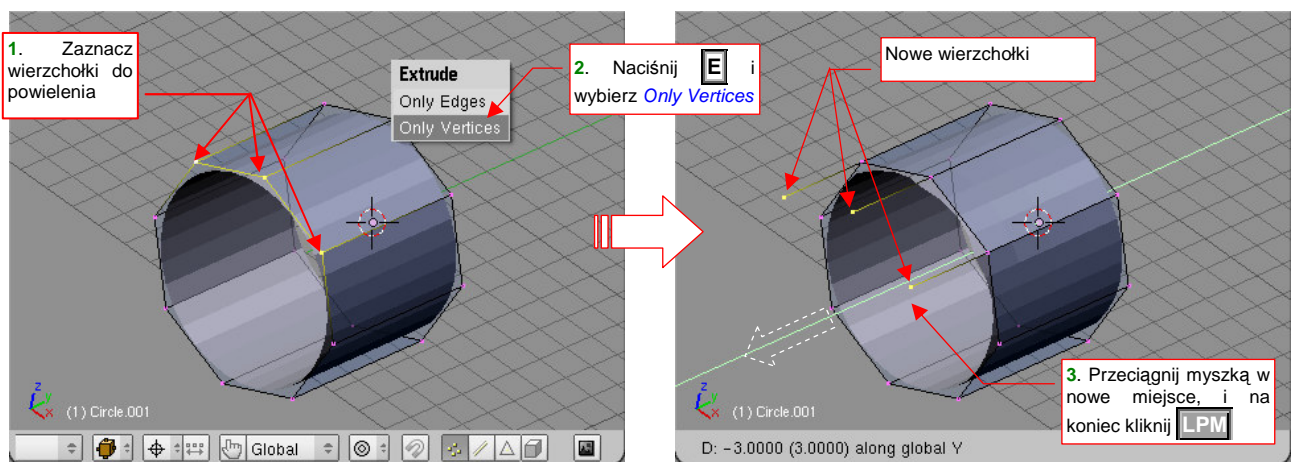
Rysunek 15.7.2 Wyłaczanie obszaru

Domyślnie kierunek przesunięcia jest ograniczony do linii prostopadłej do płaszczyzny przesuwanego obszaru.

- Jeżeli zaraz po wywołaniu polecenia *Extrude* naciśniesz **Esc** — przerwana zostanie (tylko!) druga część operacji. Mimo, że tego zazwyczaj nie widać, odpowiednie krawędzie/ ściany/ wierzchołki uległy już powieleniu, i są zaznaczone.

Gdy np. zaraz po wywołaniu *Extrude* naciśniesz **Esc** i wybierzesz polecenie skalowania — skalowane będą nowo utworzone krawędzie/ściany/wierzchołki. Ten efekt możesz świadomie wykorzystywać, gdy chciałbyś nie przesunąć, a obrócić lub zmienić skalę powielonym elementom. W taki przypadku uważaj na punkt odniesienia! Lepiej jest używać w tym charakterze kursor (tryb *3D Cursor*) — bo to punkt, nad którym w pełni panujesz. Zdanie się na wypadkową środków (*Median Point*, *Bounding Box Center*) może prowadzić do niespodziewanych rezultatów! (Chyba że masz dokładnie symetryczny rozkład zaznaczonych wierzchołków.)

Czasami zachodzi także konieczność **wyłoczenia wybranych wierzchołków** (np. po to, by utworzyć "ręcznie" nową ścianę). Operację zaczynasz tak, jak zazwyczaj — zaznacz wierzchołki do powielenia. Następnie naciśnij **E** (*Mesh→Extrude*). Z menu rozwijalnego, które się pojawi, wybierz **Only Vertices**. W tym momencie została utworzone kopie zaznaczonych wierzchołków. Kopie te będzie podążać za ruchem myszki, zupełnie jak podczas przesuwania. "Ciągnąć" za sobą będą krawędzie, łączące je z oryginałami. Kliknij **LPM** w miejscu, gdzie nowe wierzchołki mają zostać ostatecznie umieszczone (Rysunek 15.7.3):

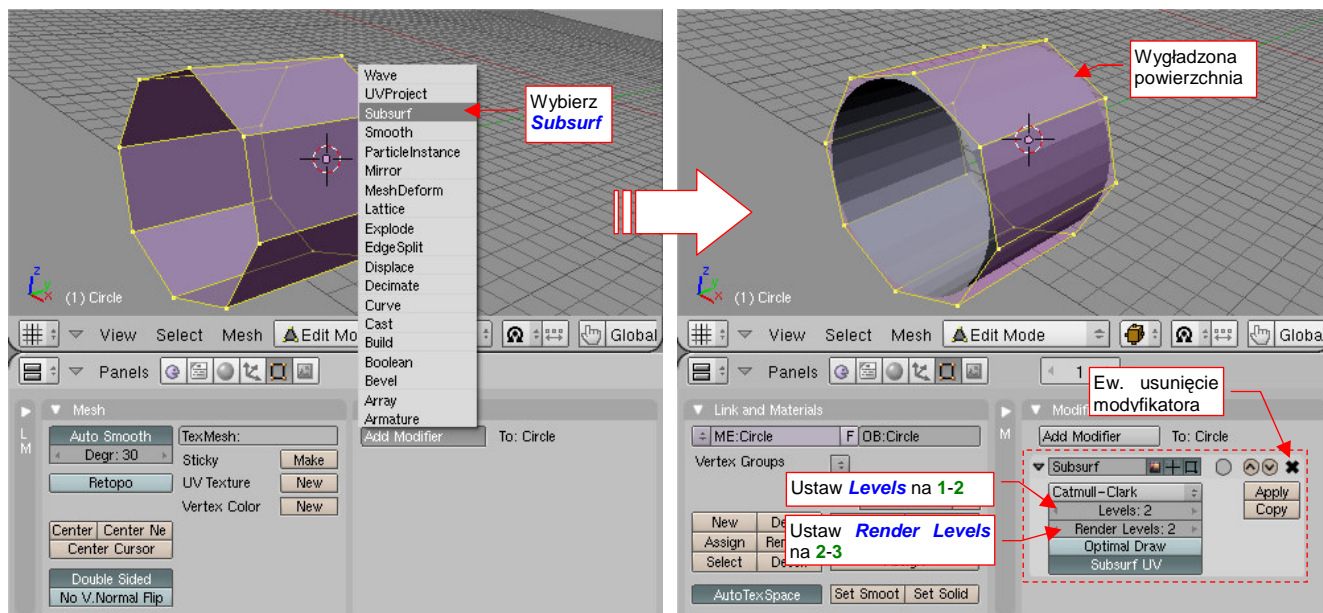


Rysunek 15.7.3 Wyłoczenie wierzchołków

15.8 Wygładzanie siatki (**Subsurf**)

Aby pokazać, jak wygładzić siatkę, przekształcę prosty, ośmiokątny graniastosłup w gładki walec.

W panelu **Modifiers** naciśnij przycisk **Add Modifier**. Spowoduje to rozwinięcie listy dostępnych "modyfikatorów" siatki. Wybierz spośród nich **Subsurf** (Rysunek 15.8.1):




Rysunek 15.8.1 Dodanie modyfikatora wygładzenia siatki (**Subsurf**)

Dodanie modyfikatora spowodowało wygładzenie powierzchni - wszystkie narożniki uległy zaokrągleniu "do środka". Faktura powierzchni zdradza, że teraz zamiast 8 ścian siatka składa się z 32 ścian. Jest to tak zwana "powierzchnia podziałowa" (ang. *subdivison surface*). Wierzchołki oryginalnej siatki pełnią rolę tzw. punktów sterujących (*control points*). Możesz zmienić kształt powłoki, zmieniając ich położenie.

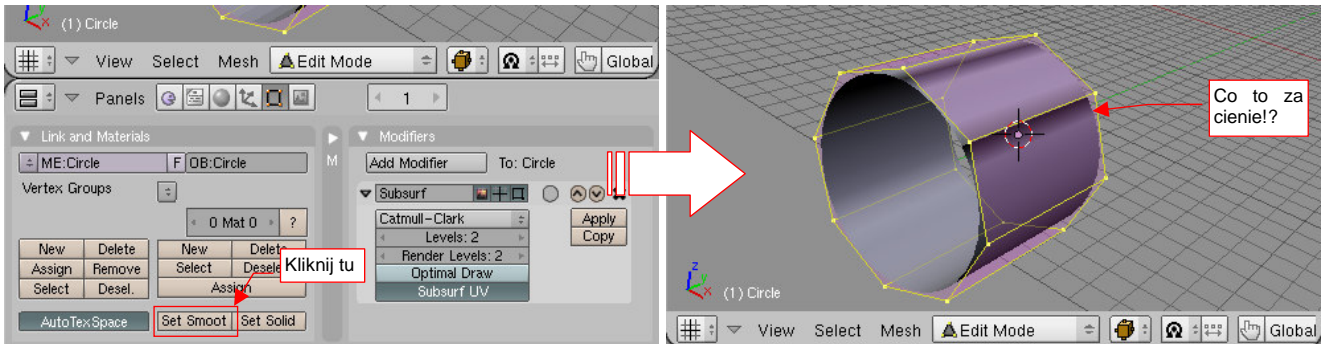
Proponuję ustalać liczbę podziałów powierzchni (**Levels**) na 1-2, w wyjątkowych przypadkach 3. To zapewni odpowiednio małą liczbę ścian przy wystarczającej "gładkości". Poziom podziału do renderowania (**Render Levels**) można ustawić o 1 większy. (Choć na pewno w jakimś stopniu wydłuży to czas obliczania finalnego obrazu). Jeżeli chcesz dowiedzieć się więcej o powierzchniach podziałowych — p. "Powierzchnie podziałowe", str. 571.

Zwróć uwagę, że możesz w każdej chwili zmienić liczbę podziałów. To może mieć znaczenie, gdy dla potrzeb gry chcesz mieć model o jak najmniejszej liczbie ścian. Dzięki **Subsurf** możesz używać tego samego modelu raz w roli "dokładnego" (większa liczba ścian). Innym razem — możesz go użyć w roli modelu "uproszczonego" (zmniejszając liczbę ścian poprzez zmniejszenie liczby podziałów).

Subsurf jest modyfikatorem siatki — oznacza to, że zmiany, jakie wprowadził, są odwracalne. Wystarczy że usuniesz ten modyfikator, klikając w mały przycisk  w jego lewym, górnym narożniku (Rysunek 15.8.1) . Wygładzona siatka zniknie, i pozostaniesz z powrotem z podstawowym zestawem ścian.

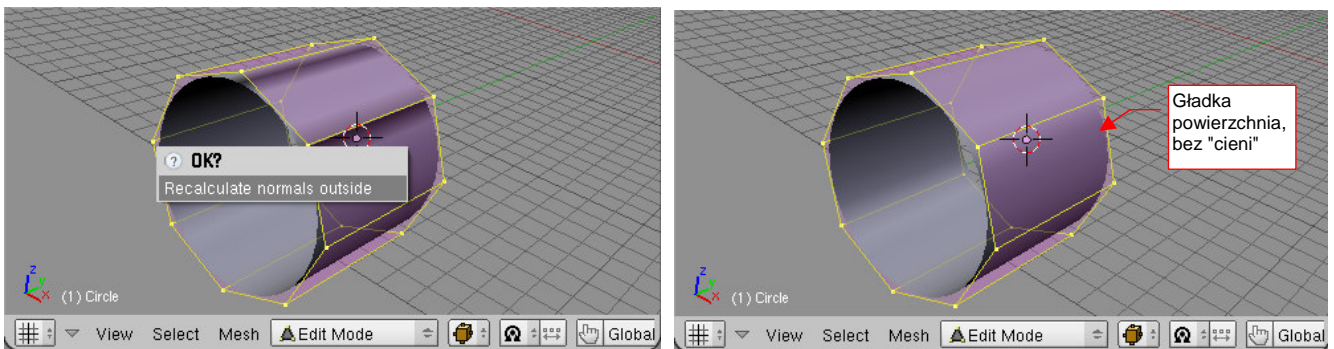
Możesz "utrwalić" podział siatki — naciskając przycisk **Apply** (Rysunek 15.8.1). Spowoduje to zniknięcie modyfikatora i trwałe przyjęcie przez siatkę kształtu powierzchni podziałowej.

Początkowo cienie na wygładzonej siatce zdradzają przebieg krawędzi elementarnych ścian (Rysunek 15.8.1). Aby temu zaradzić, zmienimy model cieniowania. Zaznacz wszystkie wierzchołki siatki (naciskając **A** — *Select*→*Select All*). Następnie na panelu *Link and Materials* kliknij w przycisk *Set Smooth* (Rysunek 15.8.2):



Rysunek 15.8.2 Zmiana modelu cieniowania na "gładki"

Efekt nie całkiem odpowiada naszym oczekiwaniom: co to za ciemne cienie, biegnące wzdłuż oryginalnych krawędzi? To rezultat niewłaściwego skierowania wektorów normalnych w niektórych wierzchołkach! Na szczęście poprawienie tego nie jest kłopotliwe. Naciśnij **Ctrl-N** (*Mesh*→*Normals*→*Recalculate Outside*). W oknie dialogowym, które się pojawi, potwierdź *Recalculate normals outside* (Rysunek 15.8.3):



Rysunek 15.8.3 Uporządkowanie kierunku normalnych

I tym razem rezultat jest już tym, o co nam chodziło - gładką powierzchnią, bez załamań i cieni w miejscu podziału ścian.

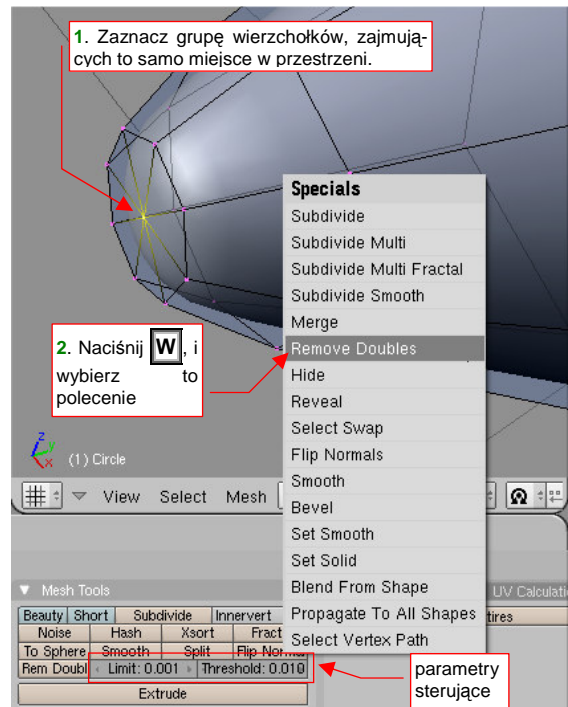
- Aby uzyskać z powrotem powierzchnię z pocieniowanymi ścianami — naciśnij przycisk *Link and Materials: Set Solid*.
- Zmianę metody cieniowania możesz ustalać oddzielnie dla różnych fragmentów siatki. W zależności od tego, jakie wierzchołki zaznaczysz, część powierzchni może być "smooth", a część — "solid".

15.9 Scalanie wierzchołków (*Remove Doubles*)

Zaznacz wierzchołki, które powinny zostać scalone w jeden. Ich współrzędne powinny być identyczne, lub niemal identyczne. Czasami, aby ten warunek zapewnić, zmniejszałem ich skalę do zera (z włączonym środkiem typu *Median Point* - p. str. 862).

Następnie naciśnij **W**, by przywołać menu *Specials*. Wybierz z niego polecenie *Remove Doubles* (Rysunek 15.9.1). (Alternatywna droga wyłącznie poprzez menu — *Mesh* → *Vertices* → *Remove Doubles*).

W efekcie tych kilka wierzchołków zostanie zastąpionych jednym. W razie potrzeby parametry tolerancji są ustawiane w panelu *Mesh Tools* — pola *Limit* i *Threshold* (Rysunek 15.9.1).

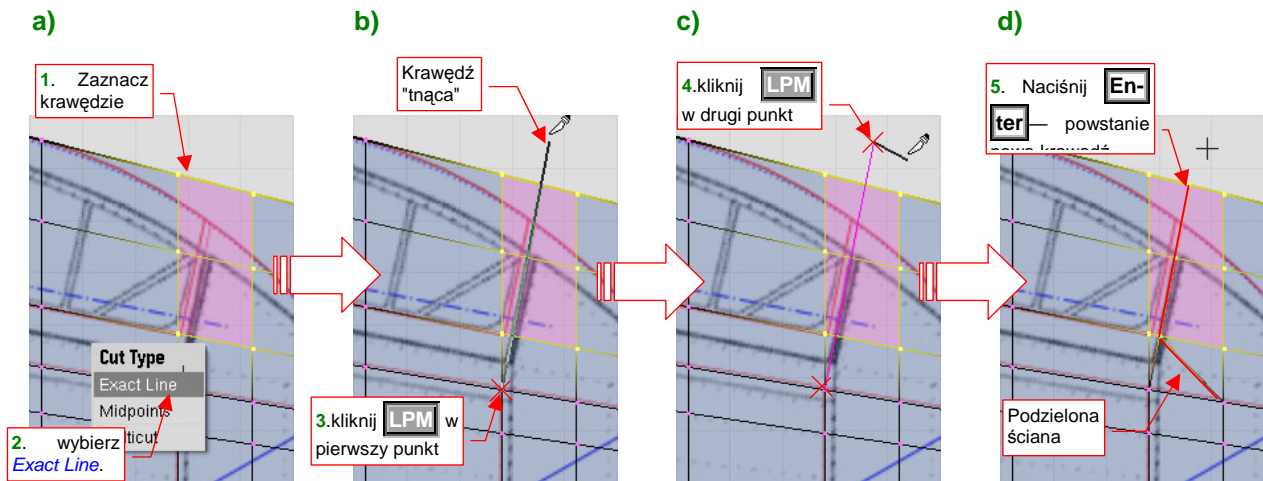


Rysunek 15.9.1 Scalanie wierzchołków

15.10 Nacinanie siatki (*Knife*)

Ta funkcja pozwala stworzyć w siatce nowe krawędzie. Powstają one poprzez przecięcie zaznaczonych ścian, w kierunku prostopadłym do ekranu (tzn. aktualnego widoku). Rolę "krawędzi tnącej" pełni narysowana na ekranie chwilowa linia łamana. Nacięcie często wykonujemy, by odwzorować jaką istotną linię konstrukcyjną — stąd przykład poniżej pokazuje nanoszenie

Zaznacz na siatce krawędzie, które mają zostać przecięte. Następnie naciśnij **Shift-K** (**Mesh** → **Edges** → **Knife Subdivide**). Wybierz z menu, które się pojawi, opcję **Exact Line** (Rysunek 15.10.1a):



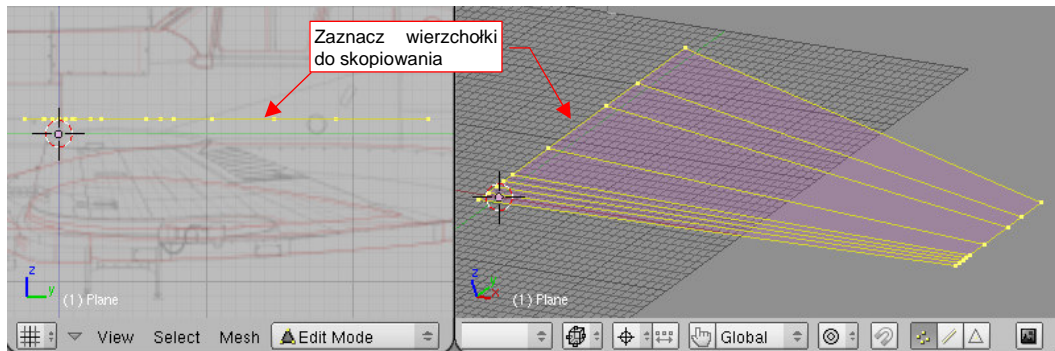
Rysunek 15.10.1 proces nacinania siatki (*Knife*)

Zwróć uwagę, że kursor myszy zmienił się teraz na mały "skalpel". Narysuj teraz "krawędź tnącą". Kliknij **LPM** w miejsce ekranu, gdzie ma się znaleźć pierwszy punkt tej krawędzi (Rysunek 15.10.1b). Za kursorem myszy zaczęła teraz podążać linia, pokazująca przebieg pierwszego segmentu linii. Kliknij **LPM** w drugi punkt — w ten sposób zakończysz rysowanie pierwszego segmentu. Blender automatycznie ciągnie od ostatnio narysowanego punktu do kursora myszy kolejną linię (Rysunek 15.10.1c). Jeżeli nie chcesz dodawać do "krawędzi tnącej" dodatkowego segmentu — naciśnij **Enter**. To zakończy rysowanie linii i spowoduje wykonanie przez Blender "nacięcia" siatki (Rysunek 15.10.1d). (Mógłbyś, oczywiście, narysować linię cięcia złożoną z kilku segmentów, ale nie jest w tym przypadku potrzebna.)

Zwróć uwagę na ścianę, w której zaczyna się krawędź nacięcia (Rysunek 15.10.1d). W wyniku nacięcia czworokątna ściana została zmieniona w trzy ściany trójkątne. To częsty efekt w otoczeniu nacinanych ścian.

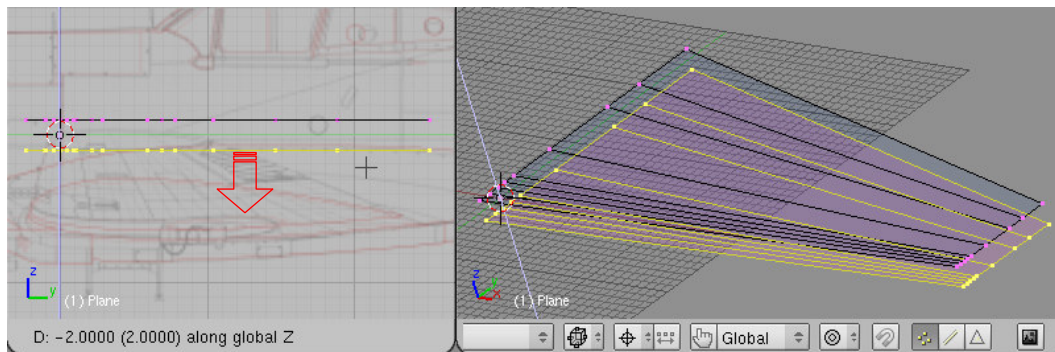
15.11 Kopiowanie wierzchołków

Zaznacz wierzchołki, które chcesz skopiować (Rysunek 15.11.1):



Rysunek 15.11.1 Zaznaczenie wierzchołków do powielenia

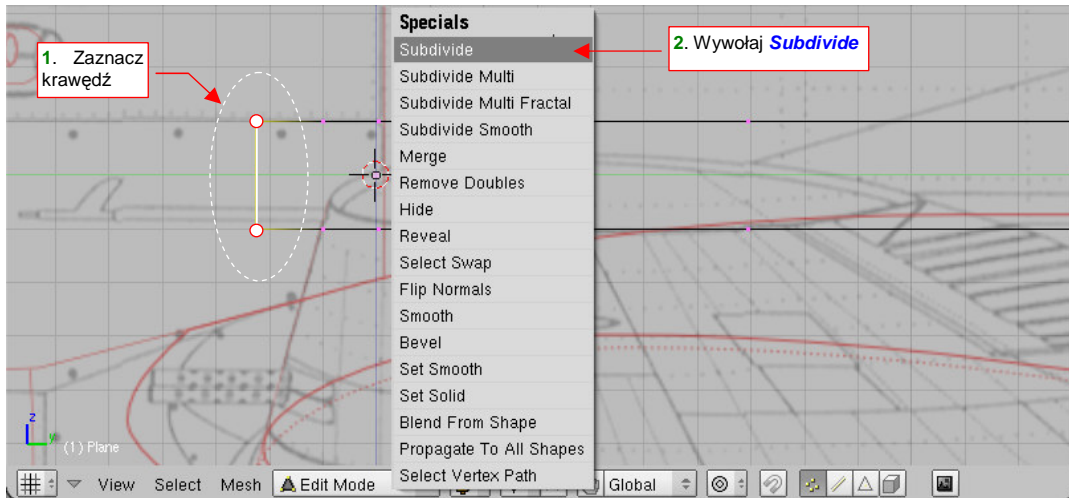
Następnie naciśnij **Shift** → **D** (**Mesh** → **Duplicate**). Powielony fragment siatki możesz od razu odsunąć od oryginału (Rysunek 15.11.2):



Rysunek 15.11.2 Odsunięcie powielonych wierzchołków

15.12 Podział krawędzi (*Subdivide*)

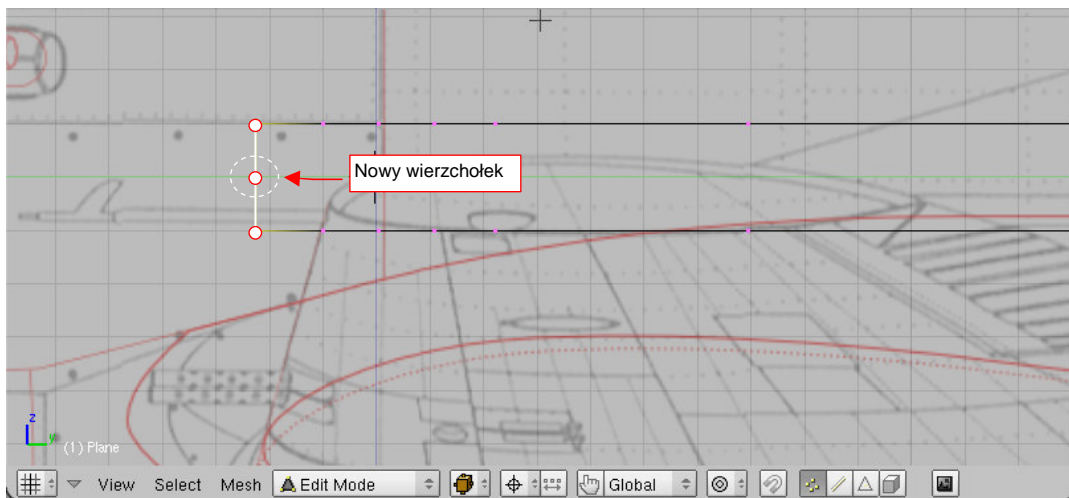
Zaznacz krawędź, które chcesz podzielić, a następnie naciśnij **W**, by otworzyć menu *Specials*. Wybierz stamtąd polecenie *Subdivide* (Rysunek 15.12.1):



Rysunek 15.12.1 Wywołanie z menu *Specials* polecenia *Subdivide*

(To samo polecenie znajduje się także w "zwykłym" menu: *Mesh*→*Edge*→*Subdivide*.)

Rysunek 15.12.2 pokazuje rezultat — w środku krawędzi pojawił się nowy wierzchołek:

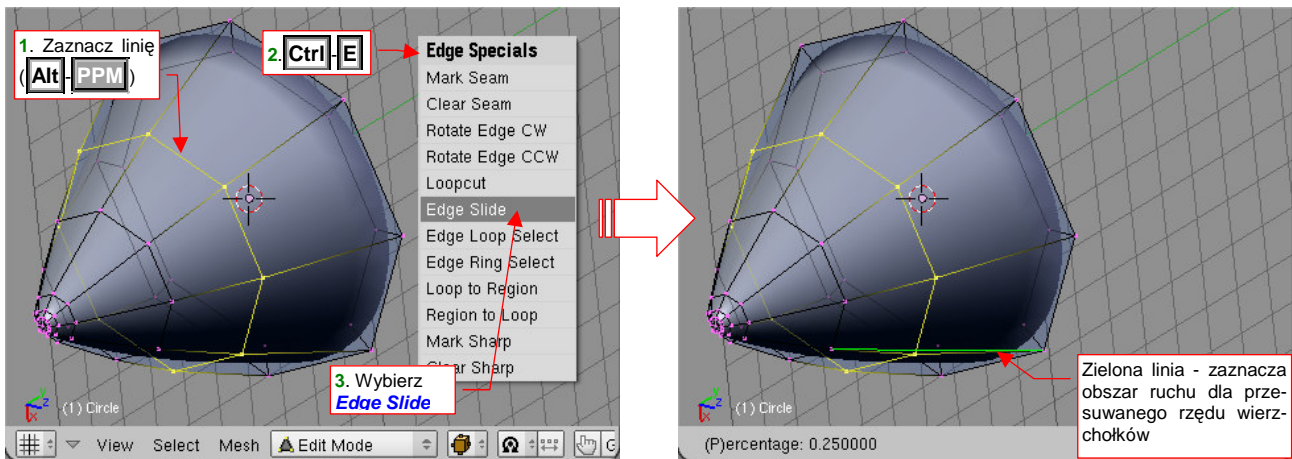


Rysunek 15.12.2 Krawędź po podziale

15.13 Przesuwanie linii wierzchołków (*Edge Slide*)

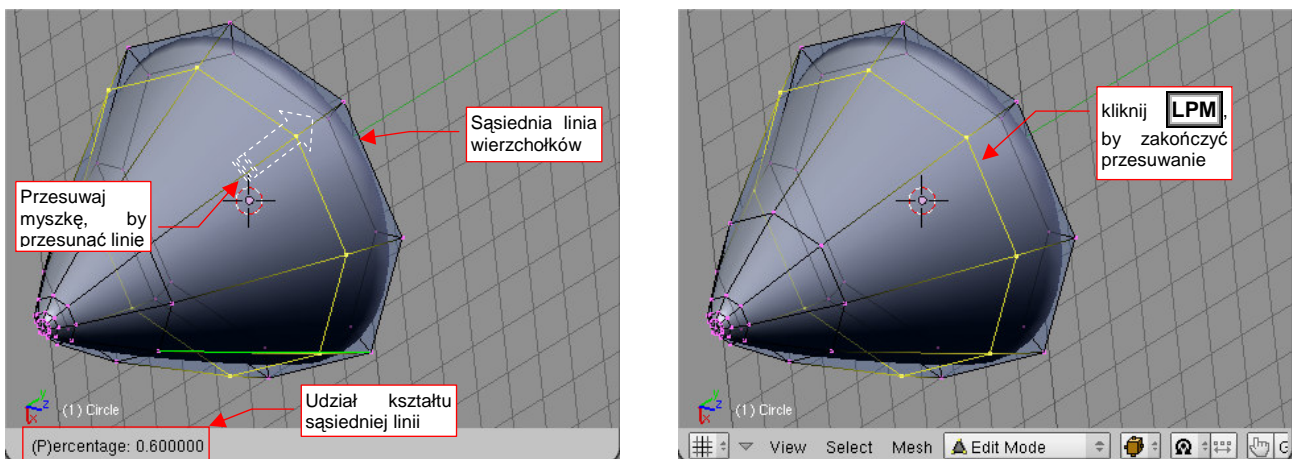
Operacja polega na przesunięciu "po dotychczasowym kształcie siatki" jednej z linii (*edgeloop*), które ją tworzą, w nowe położenie.

Zaznacz rząd wierzchołków, który ma być przesunięty. (Musi to być konkretny rząd, a nie dowolny zbiór wierzchołków — zaznaczaj więc używając **Alt** - **PPM**). Następnie naciśnij **Ctrl** - **E**, i z menu *Edge Specials* wybierz polecenie *Edge Slide* (Rysunek 15.13.1). (Możesz je także wywołać jako *Mesh*→*Edges*→*Slide Edge*):



Rysunek 15.13.1 Rozpoczęcie przesuwania linii wierzchołków

Blender połączył następny i poprzedni rząd wierzchołków zieloną linią (Rysunek 15.13.1). W ten sposób zazna- czył granice obszaru, w obrębie którego możemy przesuwać zaznaczone wierzchołki. Od tej chwili każde prze- sunięcie myszy powoduje wybranej linii wierzchołków. Gdy zbliżymy ją do linii z prawej (Rysunek 15.13.2), kształt będzie rósł, aż — na końcu — stanie się identyczny z poprzednim rzędem. Gdy zbliżymy zaznaczo- ną linię do linii z lewej — kształt będzie proporcjonalnie się zmniejszał. Kończymy całą operację, naciskając **LPM** (Rysunek 15.13.2).

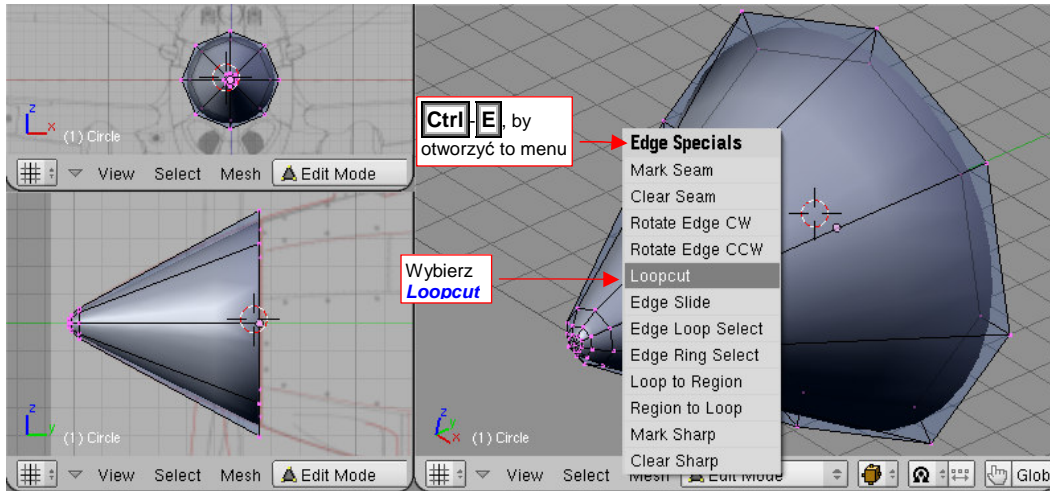


Rysunek 15.13.2 Przesunięcie linii wierzchołków

15.14 Wstawienie nowej linii wierzchołków (*Loopcut*)

Operacja polega na wstawieniu nowej linii wierzchołków (*egdeloop*), pomiędzy istniejące linie siatki.

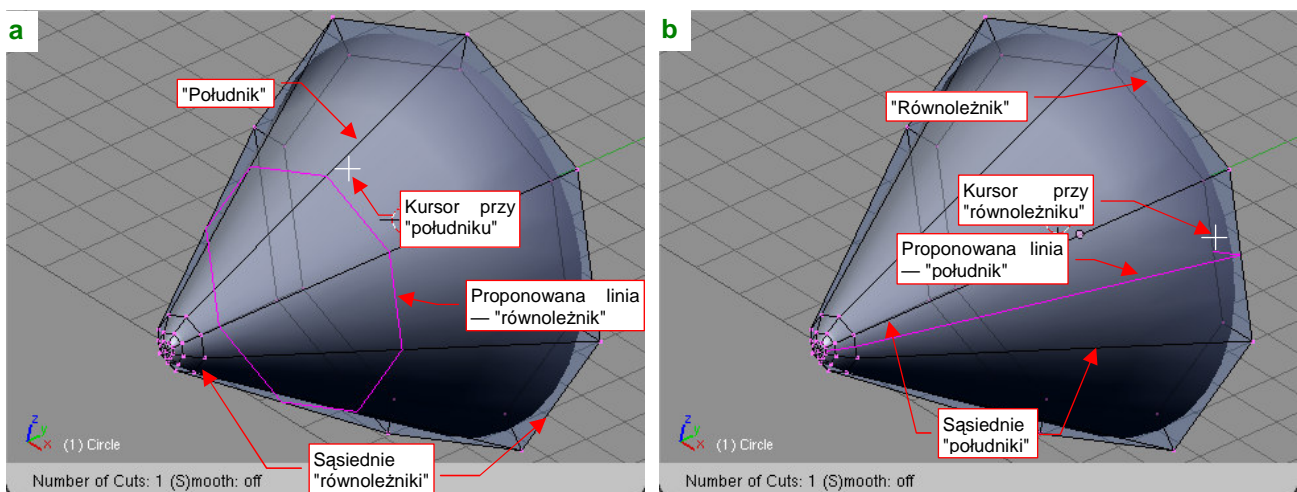
Naciśnij **Ctrl** - **E**, i z menu *Edge Specials* wybierz polecenie *Loopcut* (Rysunek 15.14.1):



Rysunek 15.14.1 Wywołanie polecenia *Loopcut*

Teraz masz wybrać, w jakim kierunku, i pomiędzy którymi liniami chcesz wstawić nowy rząd wierzchołków. W zależności od tego, gdzie ustawisz mysz, Blender będzie pokazywał aktualną możliwość za pomocą fioletowej linii.

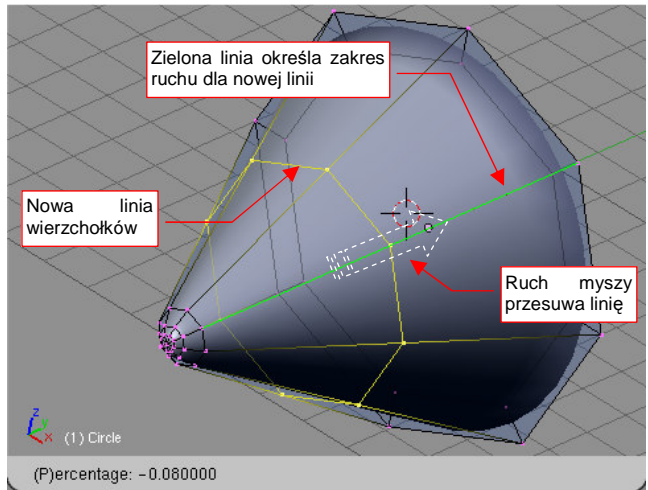
Dla uproszczenia tłumaczenia, nazwijmy jedne linie siatki "równoleżnikami", a drugie "południkami". Rysunek 15.14.2a pokazuje, że gdy zbliżysz kursor do jakiegoś "południka" siatki, Blender zaproponuje utworzenie nowego równoleżnika. Ta nowa linia zostanie utworzona pomiędzy dwoma sąsiednimi równoleżnikami, które są najbliższe kursora. Jeżeli z kolei zbliżysz kursor w pobliże "równoleżnika" — Blender zaproponuje nowy "południk" (Rysunek 15.14.2b).



Rysunek 15.14.2 Wybór kierunku i pary sąsiednich linii

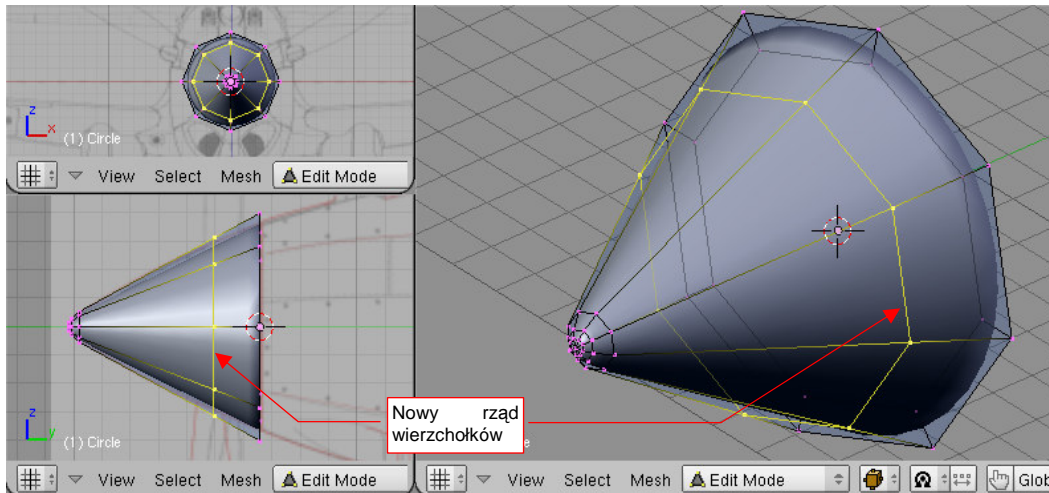
Wybór kierunku i sąsiednich linii siatki zatwierdzasz kliknięciem **LPM**.

Powiedzmy, że wybraliśmy opcję stworzenia nowego "równoleżnika" (Rysunek 15.14.2a). Po kliknięciu **LPM** Blender połączył sąsiednie "równoleżniki" zieloną linią, i wstawił pomiędzy nie nowy (Rysunek 15.14.3). W ten sposób zaznaczył granice obszaru, w obrębie którego możemy przesuwać nowy rząd wierzchołków. Od tej chwili każde przesunięcie myszy powoduje przesuwanie wybranej linii wierzchołków po powierzchni siatki. Kończymy całą operację, naciskając jeszcze **LPM** (Rysunek 15.13.2).



Rysunek 15.14.3 Przesuwanie nowej linii wierzchołków po powierzchni siatki

Ostateczny efekt pokazuje Rysunek 15.14.4 — możesz porównać ze stanem początkowym (Rysunek 15.14.1):

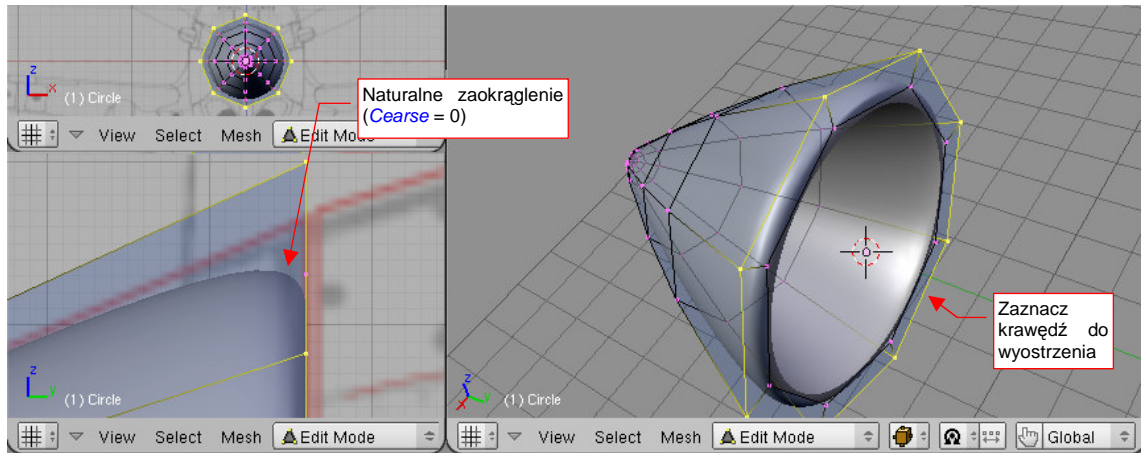


Rysunek 15.14.4 Wstawiony nowy rząd wierzchołków

15.15 Zmiana ostrości krawędzi (*Cease*)

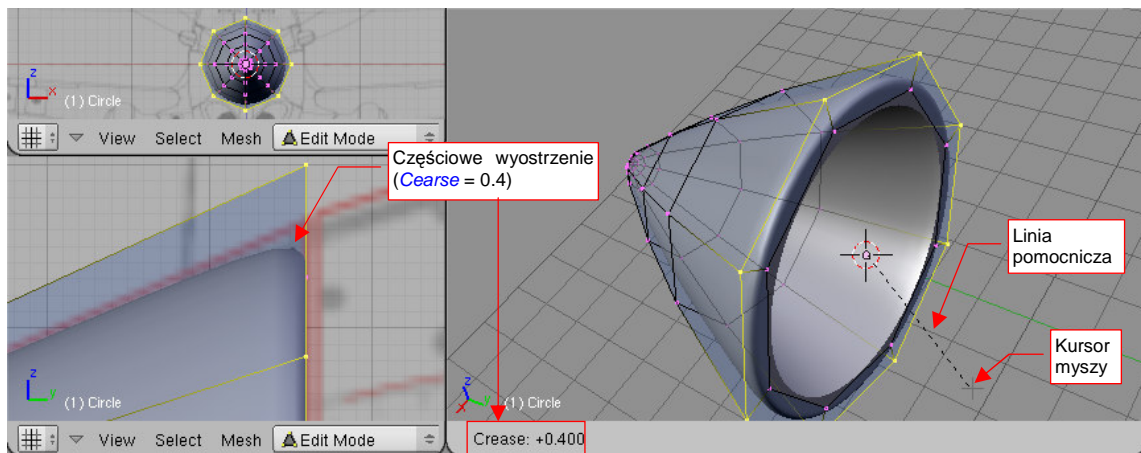
Blender pozwala na zmianę ostrości (*cearse*) wybranych krawędzi "wygładzonej" siatki. Możesz zwiększać ich ostrość, możesz także z powrotem czynić je gładkimi. Sprowadza się to do tego, że każda krawędź ma dodatkowy parametr — *cearse* — który może przyjmować wartości od 0 (zupełnie gładka) do 1 (zupełnie ostra). Zmianę ostrości pokażę na przykładzie zwiększenia ostrości krawędzi podstawy stożka. (Analogicznie możesz ostrą krawędź wygładzić).

Zaznacz krawędzie, które chcesz wyostrzyć. Następnie naciśnij **Shift-E** (*Mesh→Edges→Cearse Subsurf*) (Rysunek 15.15.1):



Rysunek 15.15.1 Stan początkowy: krawędź "naturalnie zaokrąglona"

Spowoduje to pojawienie się pomocniczej, przerywanej linii, biegnącej od środka zaznaczonych wierzchołków do aktualnej pozycji kursora myszy (Rysunek 15.15.2):

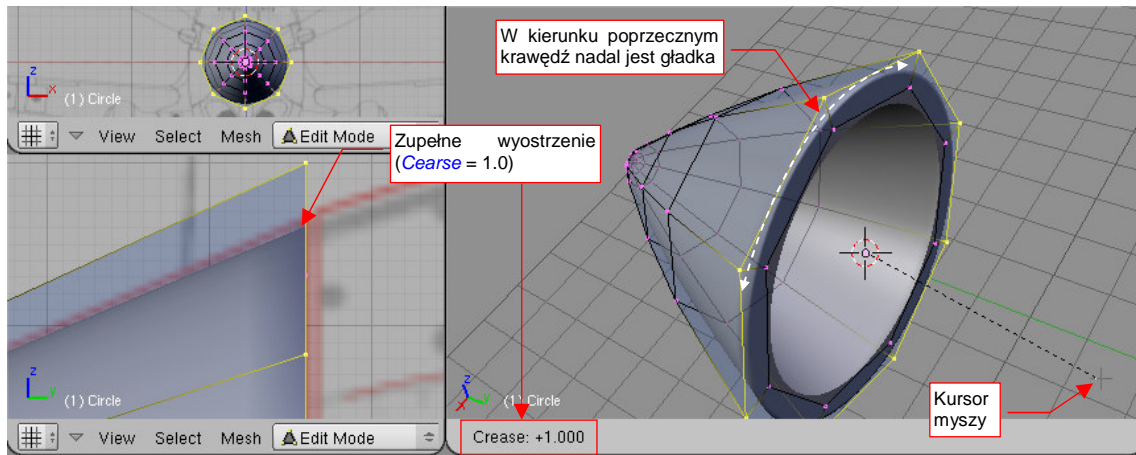


Rysunek 15.15.2 Wyostrzenie krawędzi

Od tej chwili położenie kursora określa poziom wyostrzenia (*cearse*). Jego wartość jest pokazywana na bieżąco w nagłówku aktywnego okna (Rysunek 15.15.2). Do dyspozycji masz zakres wartości od *cearse* = 0 (naturalna powierzchnia, bez żadnego zaokrąglenia), do *cearse* = 1.0 (ostra krawędź). Rysunek 15.15.2 pokazuje, na ile wyostrzyła się krawędź przy *cearse* = 0.4.

- Podczas zmiany ostrości krawędzi możesz także wykorzystywać klawisz **Ctrl** — powoduje skokową zmianę parametru *cearse* o 0.1.

Gdy zwiększysz ostrość (*cearse*) do poziomu = 1.0, uzyskasz absolutnie ostrą krawędź (Rysunek 15.15.3):

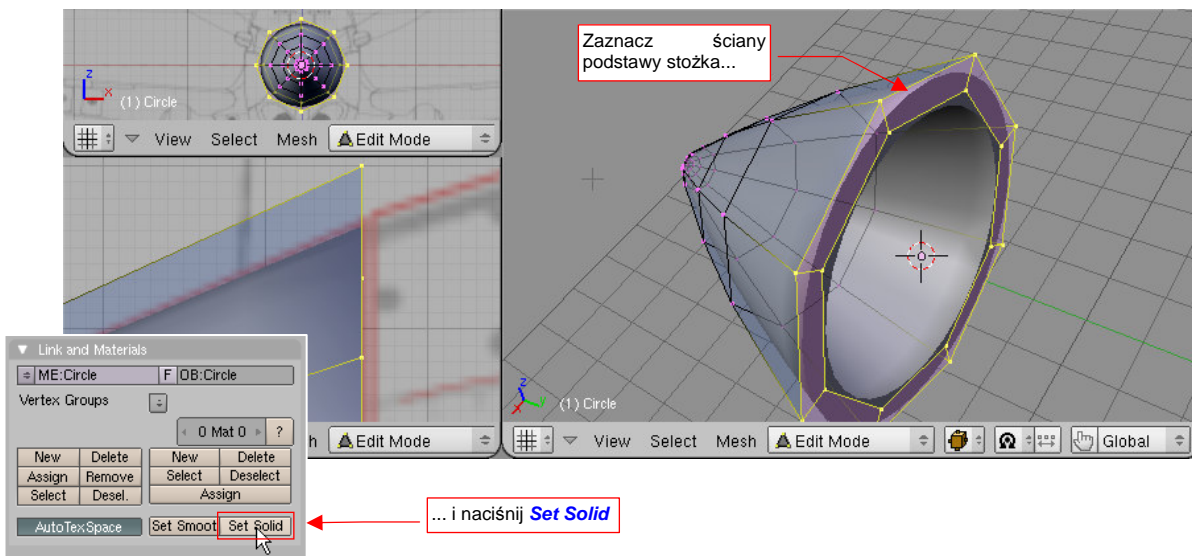


Rysunek 15.15.3 Krawędź ostra (*cearse* = 1.0)

Zwróć uwagę, że krawędź jest ostra tylko w jednym kierunku. W kierunku poprzecznym nadal jest gładka, dzięki czemu zachowała kształt okręgu. (Gdybyś chciał uzyskać kształt ośmiokąta — powinieneś wyostrzyć także krawędzie "południków" tego stożka)

Zatwierdzamy zmianę ostrości krawędzi kliknięciem **LPM**.

Na koniec czynność kosmetyczna — krawędź będzie lepiej wyglądać, gdy zmienimy sposób cieniowania ścian podstawy z "gładkiego" na "płaski". Zaznacz ściany podstawy i w panelu **Link and Materials** naciśnij przycisk **Set Solid**. J (Rysunek 15.15.4):

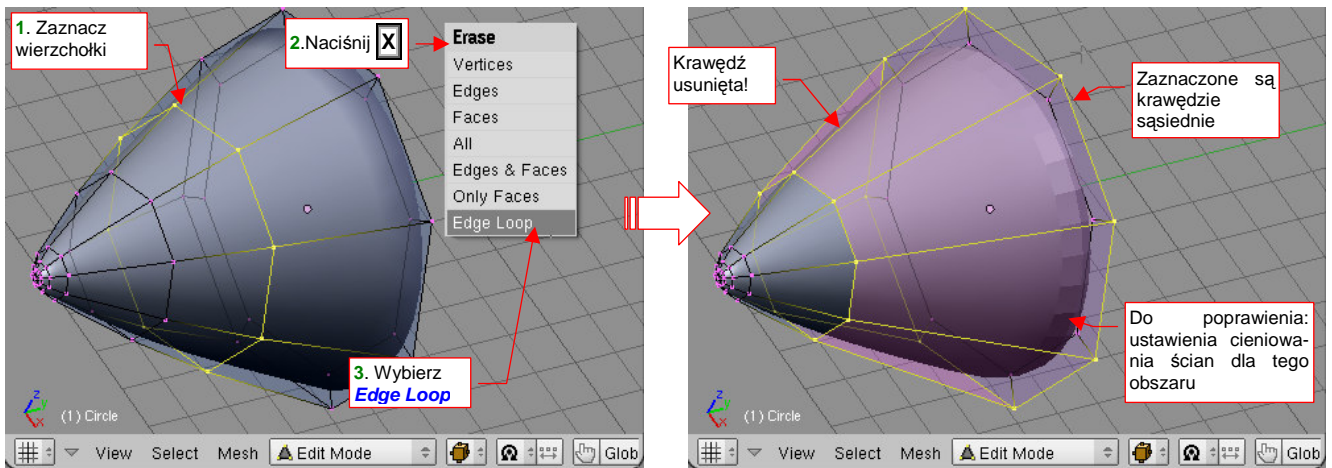


Rysunek 15.15.4 Zmiana sposobu cieniowania ścian podstawy na "płaski" (**solid**)

15.16 Usuwanie linii wierzchołków

Usuwanie linii wierzchołków różni się od wszystkich innych "usunięć" w trybie edycji. Różnica polega na tym, że zachowuje ciągłość ścian siatki. Jest to operacja odwrotna do wstawienia nowej linii wierzchołków poleceniem *Loopcut* (str. 875).

Zaznacz linię wierzchołków, którą zamierzasz usunąć (najlepiej — użyj **PPM** i klawisza **Alt**, aby na pewno zaznaczyć całą linię). Następnie naciśnij **X** (lub *Mesh→Delete...*) i z menu *Erase*, które się pojawi, wybierz opcję *Edge Loop* (Rysunek 15.16.1):



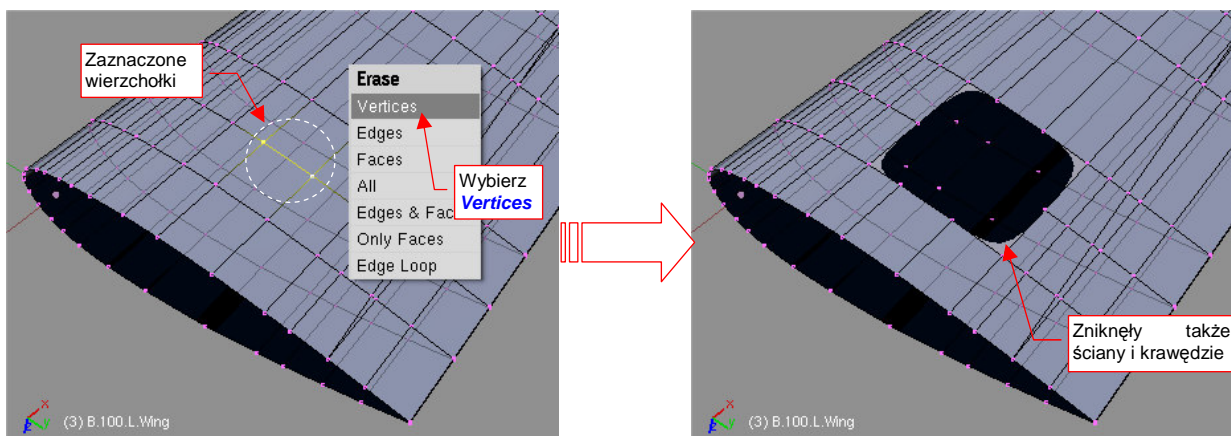
Rysunek 15.16.1 Usuwanie linii wierzchołków

W efekcie rząd wierzchołków, który chcieliśmy usunąć, znikł z siatki. Ciągłość powierzchni została jednak zachowana — sąsiadujące wzdłuż usuwanej krawędzi ściany uległy scaleniu.

Przy usuwaniu krawędzi warto zdawać sobie sprawę, że sąsiednie linie wierzchołków mogą ulec "uproszczeniu": np. stracą przypisaną ostrość (*cearse*). Rysunek 15.16.1 pokazuje właśnie taką sytuację. Należy tu повторно zaznaczyć krawędź podstawy stożka jako ostrą (p. str. 877).

15.17 Usuwanie wierzchołków

Zaznacz wierzchołki, które chcesz usunąć, a następnie naciśnij **Delete** (lub **X**). Z menu *Erase* wybierz *Vertices* (Rysunek 15.17.1):

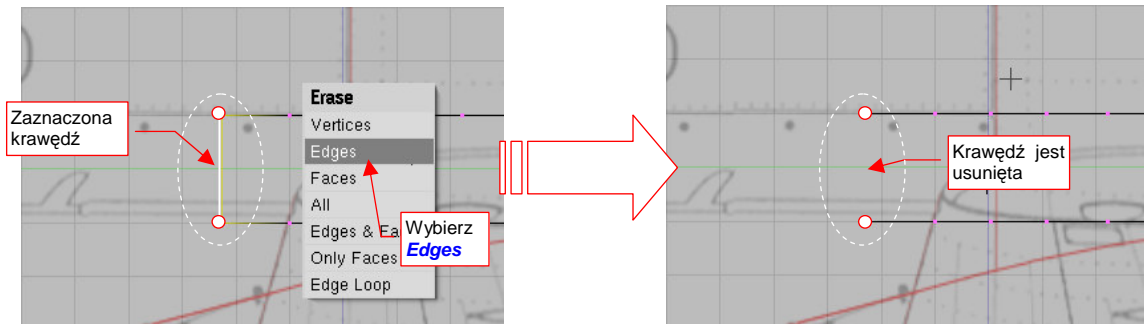


Rysunek 15.17.1 Usuwanie wierzchołków z siatki

W rezultacie usunięcia wierzchołki znikną także wszystkie krawędzie i ściany, do których należał. Stąd zazwyczaj rezultatem takiej operacji jest "dziura" w siatce. (Oczywiście, to "twórcza destrukcja". Zazwyczaj w ten sposób przygotowuje się miejsce na nowe ściany, otaczające np. otwór.)

15.18 Usuwanie krawędzi

Zaznacz krawędź/krawędzie, które chcesz usunąć (wygodniej to zrobić w trybie wyboru krawędzi, a nie wierzchołków — p. str. 859). Następnie naciśnij **Delete** (lub **X**). Z menu *Erase* wybierz **Edges** (Rysunek 15.18.1):

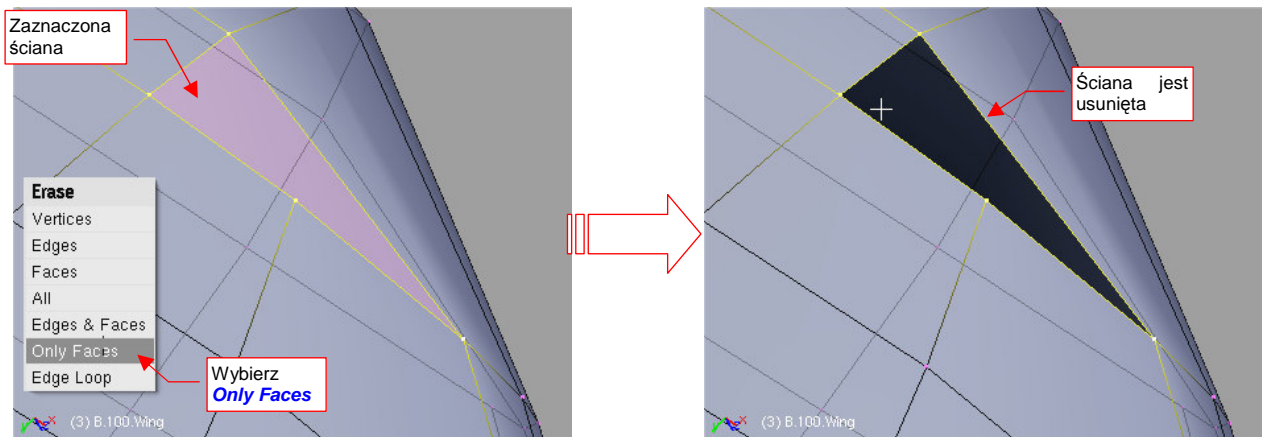


Rysunek 15.18.1 Usuwanie wierzchołków z siatki

W rezultacie usunięcia krawędzi znikną także wszystkie ściany, które dzieliła.

15.19 Usuwanie ścian

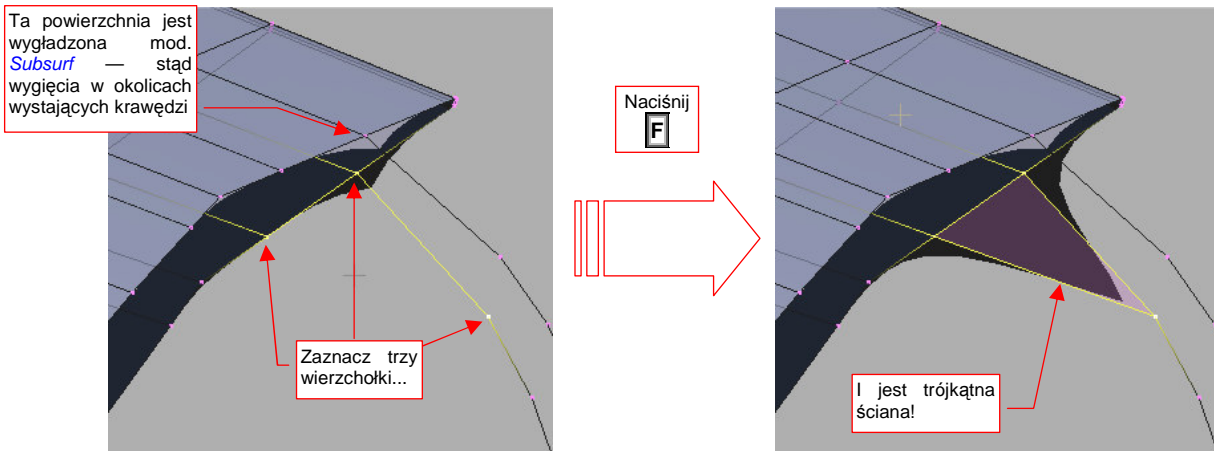
Zaznacz ścianę/ściany, które chcesz usunąć (wygodniej to zrobić w trybie wyboru ścian, a nie wierzchołków — p. str. 859). Następnie naciśnij **Delete** (lub **X**). Z menu *Erase* wybierz **Only Faces** (Rysunek 15.19.1):



Rysunek 15.19.1 Usuwanie ścian z siatki

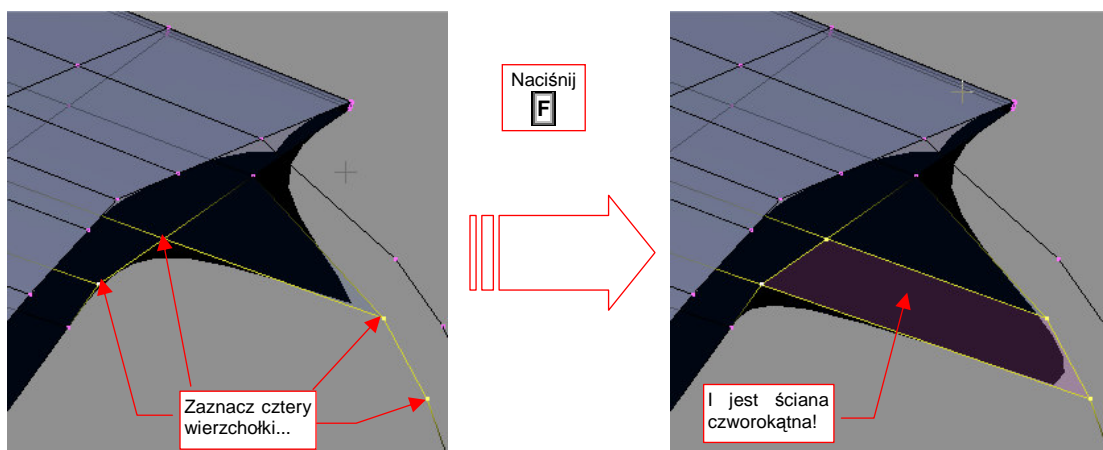
15.20 Tworzenie nowej ściany

Zaznacz wierzchołki (lub krawędzie), na których chcesz "rozpiąć" nową ścianę. Następnie naciśnij **F** — **Mesh → Make Edge/Face** (Rysunek 15.20.1):



Rysunek 15.20.1 Dodanie trójkątnej ściany do siatki

Gdy zaznaczysz trzy wierzchołki i naciśniesz **F**, powstanie ściana trójkątna. Gdy zaznaczysz cztery wierzchołki i naciśniesz **F** — powstanie ściana czworokątna (Rysunek 15.20.2):

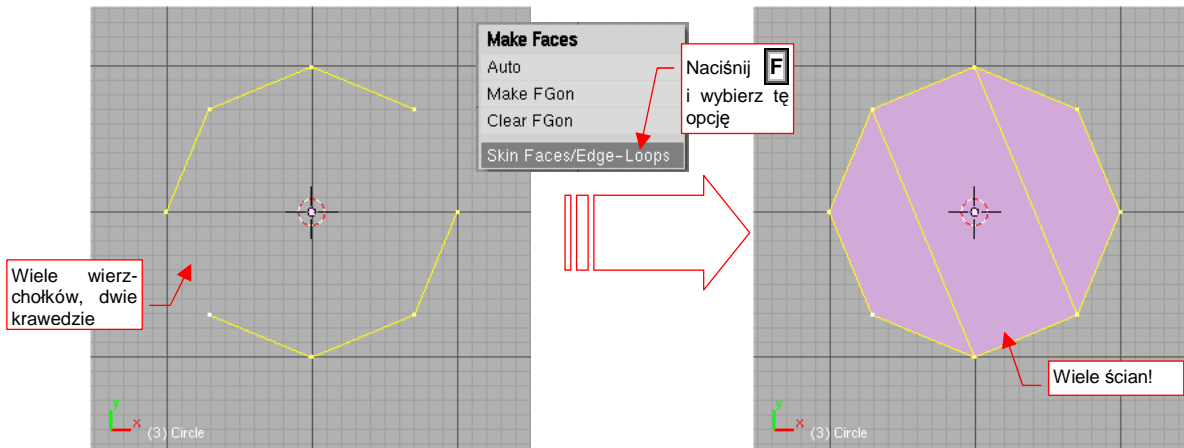


Rysunek 15.20.2 Dodanie czworokątnej ściany do siatki

Wraz z nową ścianą utworzone są nowe krawędzie (tam, gdzie ich brakowało). Siatki w Blenderze mogą się składać ze ścian trójkątnych i czworokątnych.

- W przypadku ścian czworokątnych można tylko tworzyć tzw. czworokąty wypukłe. Gdy kąt rozwarcia krawędzi w jakimkolwiek z wierzchołków jest $> 180^\circ$ — Blender odmawia utworzenia ściany czworokątnej. Mimo to nadal możesz z tych wierzchołków utworzyć dwie ściany trójkątne.

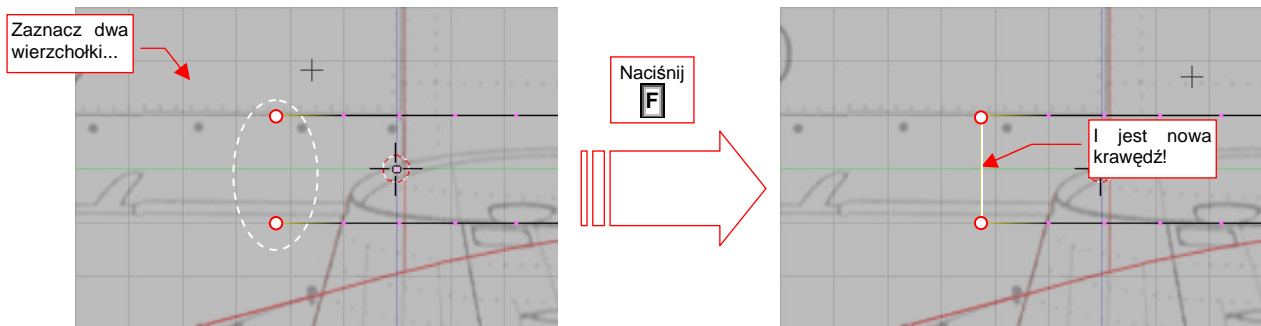
Jeżeli zaznaczysz więcej wierzchołków niż cztery i naciśniesz **F**, Blender zaproponuje automatyczne stworzenie całego ciągu ścian. Wyświetli wówczas na ekranie kilka opcji do wyboru w menu *Make Faces*. Wybierz stamtąd *Skin Faces/Edge Loops*. Jeżeli wskazałeś dwie krawędzie, zostanie automatycznie wygenerowany ciąg ścian, łączących ze sobą poszczególne wierzchołki (Rysunek 15.20.3). (Sądzę, że krawędzie tych nowych ścian są biegać pomiędzy najbliższymi wierzchołkami każdej ze wskazanych linii). Ta metoda pozwala szybko utworzyć siatkę dla stosunkowo regularnych ciągów wierzchołków. Jak każdy "automat" potrafi dać nieoczekiwany rezultat — ale wtedy wystarczy nacisnąć **Ctrl-Z** (*Undo*), aby wycofać taki niepożądany efekt.



Rysunek 15.20.3 Utworzenie całego ciągu ścian

15.21 Tworzenie nowej krawędzi

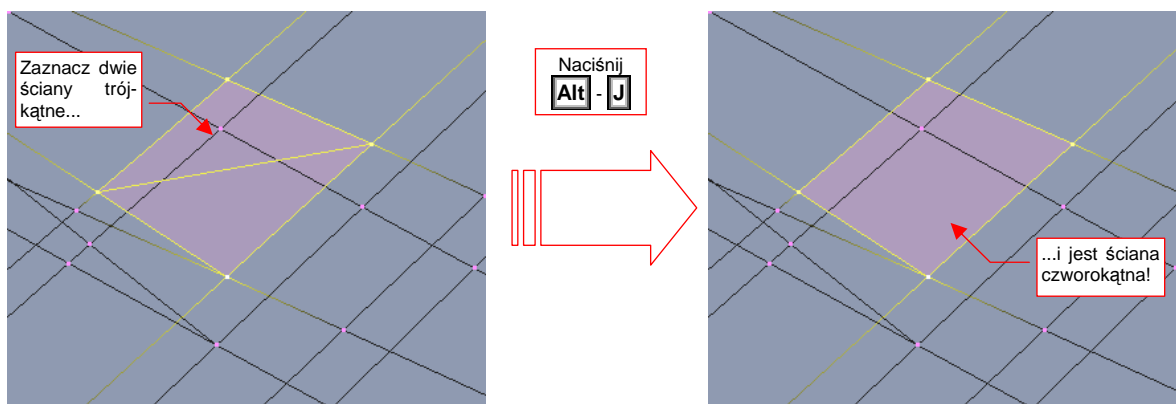
Zaznacz dwa wierzchołki, które chcesz połączyć krawędzią. Następnie naciśnij **F** — **Mesh → Make Edge/Face** (Rysunek 15.21.1):



Rysunek 15.21.1 Dodanie do siatki nowej krawędzi

15.22 Scalanie ścian trójkątnych

Zaznacz trójkątne ściany, które chcesz złączyć ze sobą (muszą mieć wspólną krawędź). Następnie naciśnij **Alt-J** — **Mesh** → **Faces** → **Convert Triangles to Quads** (Rysunek 15.22.1):

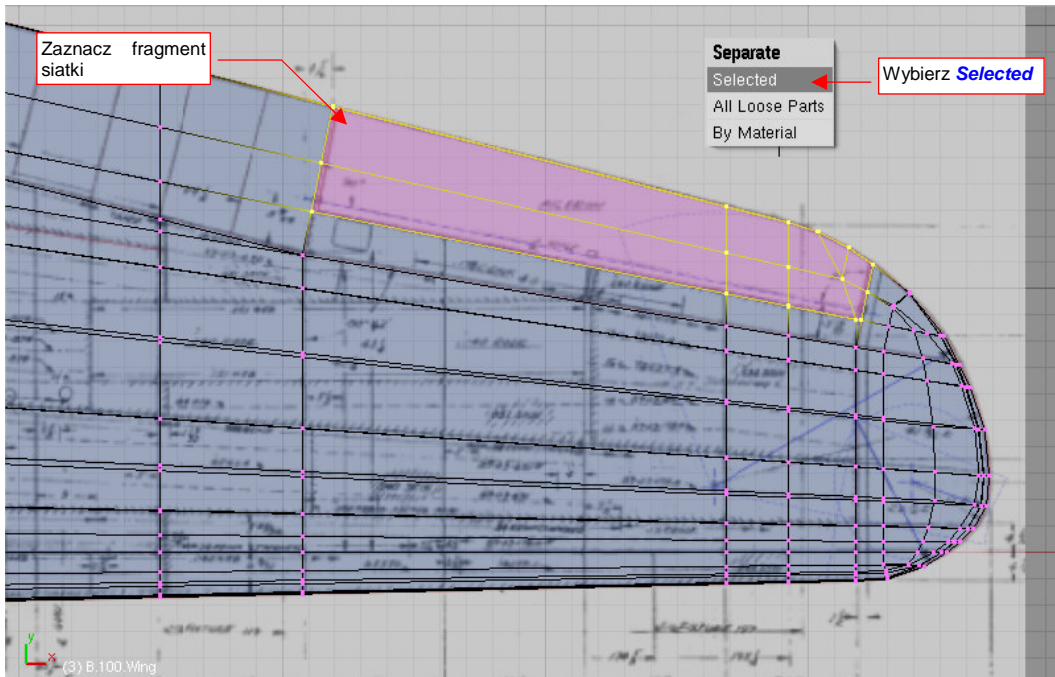


Rysunek 15.22.1 Scalanie dwóch ścian trójkątnych w jedną ścianę czworokątną

- Scalanie nie uda się, gdy trójkąty nie tworzą wypukłego czworokąta (Gdy kąt rozwarcia krawędzi w jakimkolwiek z wierzchołków byłby $> 180^\circ$)

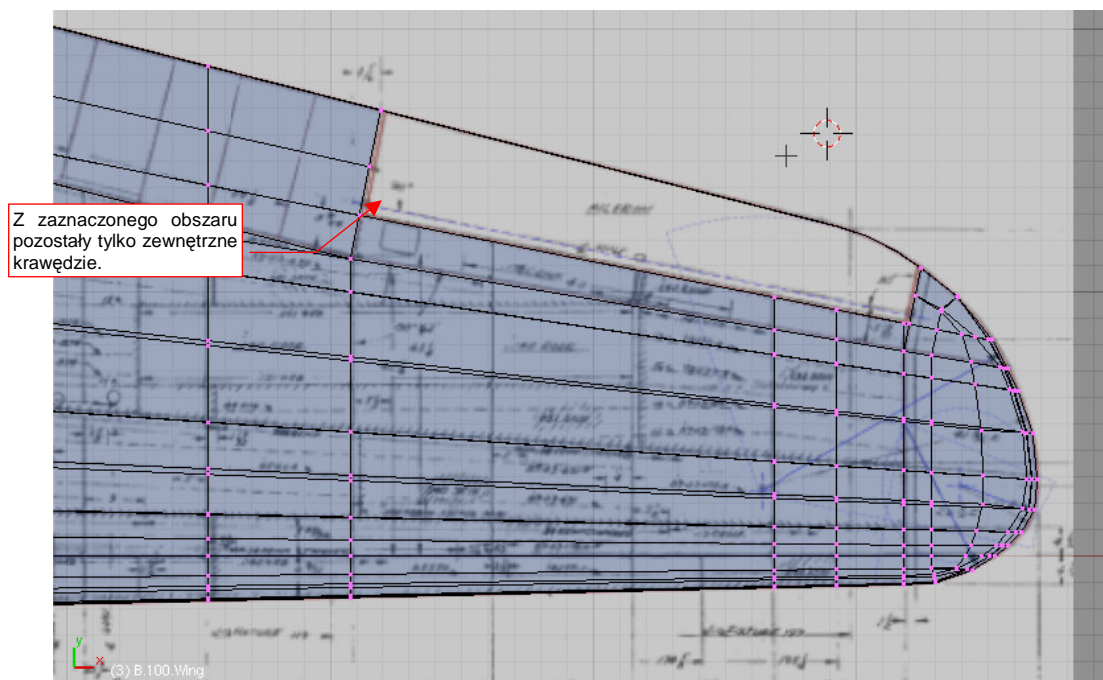
15.23 Wydzielenie fragmentu siatki w nowy obiekt (*Separate*)

Zaznacz fragment siatki, który chcesz przekształcić w nowy obiekt. Następnie naciśnij **P** — *Mesh* → *Vertices* → *Separate*. Z menu opcji, które się pojawiło, wybierz **Selected** (Rysunek 15.23.1):



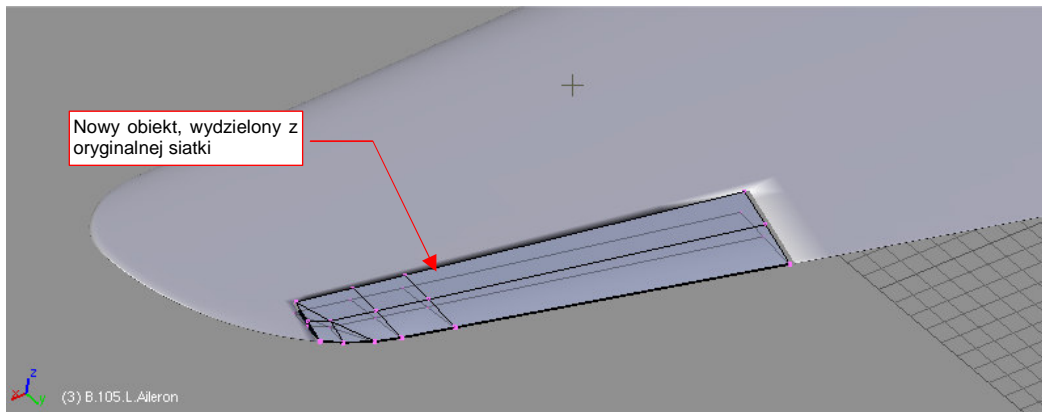
Rysunek 15.23.1 Wydzielenie zaznaczenia (lotki) w nową siatkę

W rezultacie zaznaczone ściany znikają z oryginalnej siatki. Nie są usuwane jednak tak samo, jak gdybyśmy wybrali polecenie *Erase* (str. 880). W tym przypadku krawędzie zaznaczonego obszaru pozostaną na swoim miejscu (Rysunek 15.23.2):



Rysunek 15.23.2 Siatka po wydzieleniu lotki

Gdzie się podział zaznaczony fragment siatki? Został umieszczony w nowym obiekcie (Rysunek 15.23.3):



Rysunek 15.23.3 Lotka, wydzielona z siatki

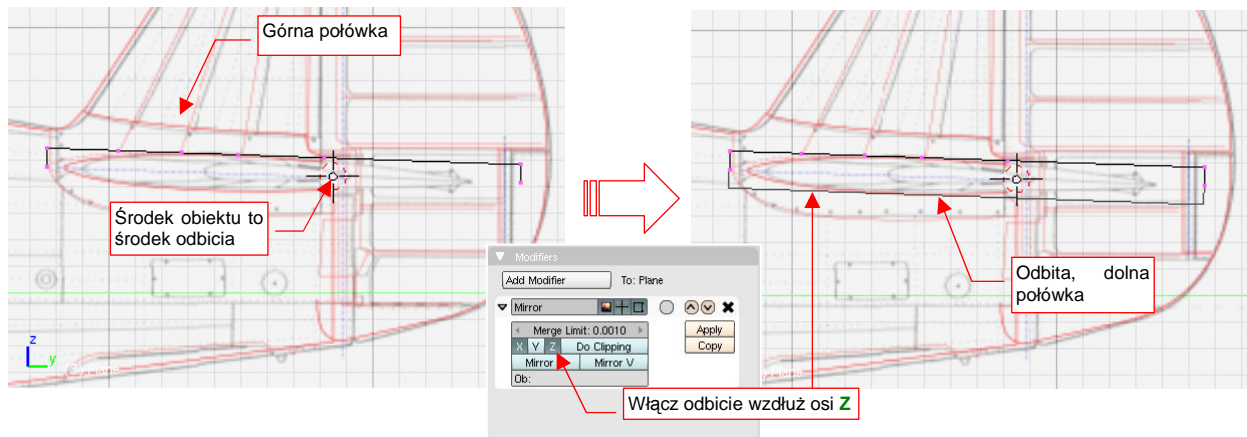
Nowy obiekt otrzymał taką samą nazwę jak ten, z którego powstał, zakończoną jednak dodatkową końcówką typu ".001". Na przykład lotka, wydzielona w ten sposób z płata **B.100.Wing**, otrzymuje początkową nazwę **B.100.Wing.001**. Oczywiście, zmień ją zaraz na prawidłową (str. 798).

- Operacją odwrotną do wydzielenia siatki jest scalenie siatek obiektów ([Join Objects](#), str. 806).

15.24 Lustrzane odbicie siatki (*Mirror*)

Siatki obiektów symetrycznych można tworzyć w uproszczony sposób — modelując tylko jedną połowę. Kształt drugiej strony można uzyskać, stosując modyfikator *Mirror*.

Aby stworzyć lustrzane odbicie siatki, w zestawie *Editing*, panelu *Modifiers*, naciśnij przycisk *Add Modifier*. Spowoduje to rozwinięcie listy dostępnych "modyfikatorów" siatki. Wybierz spośród nich *Mirror*.



Rysunek 15.24.1 Zastosowanie modyfikatora *Mirror*

Środkiem lustrzanego odbicia, wykonywanego przez modyfikator *Mirror*, jest zawsze środek obiektu (tzn. jego punkt odniesienia — nie mylić ze środkiem geometrycznym). Stąd też zawczasu należy zadbać, aby znajdował się we właściwym miejscu (por. str. 808). Odbicie dokonuje się wzdłuż lokalnych osi **X**, **Y**, **Z** obiektu. Oś odbicia ustalasz przełącznikami w panelu modyfikatora (**X**, **Y**, **Z** — Rysunek 15.24.1). Możliwe jest jednoczesne włączenie odbicia wzdłuż dwóch, a nawet wszystkich trzech osi.

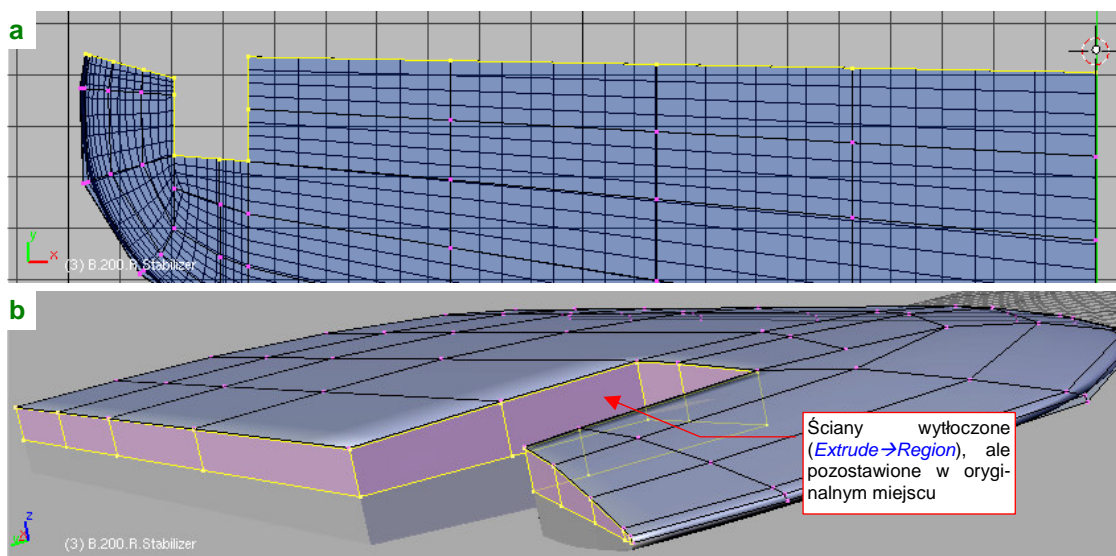
Naciśnięcie przycisku *Apply* spowoduje "utrwalenie" siatki tworzonej przez modyfikator, tzn. stworzenie nowych wierzchołków i usunięcie modyfikatora *Mirror*.

Gdy stosujesz ten modyfikator dla powierzchni podziałowych — razem z modyfikatorem *Subsurf* — pamiętaj, aby *Mirror* był zawsze powyżej *Subsurf*. Szczególnie ważne w przypadku powierzchni podziałowych jest także ustawienie wartości *Merge Limit*. Dwa wierzchołki — oryginał i jego odbity obraz — leżące w odległości mniejszej niż *Merge Limit*, są traktowane jakby były jednym wierzchołkiem. Dzięki tej właściwości wygładzone modyfikatorem *Subsurf* powierzchnie nie mają w płaszczyźnie symetrii ostrej krawędzi, ani przerwy. Zazwyczaj nie musisz zmieniać wartości *Merge Limit*.

15.25 Odsunięcie (*Shrink/Fatten*)

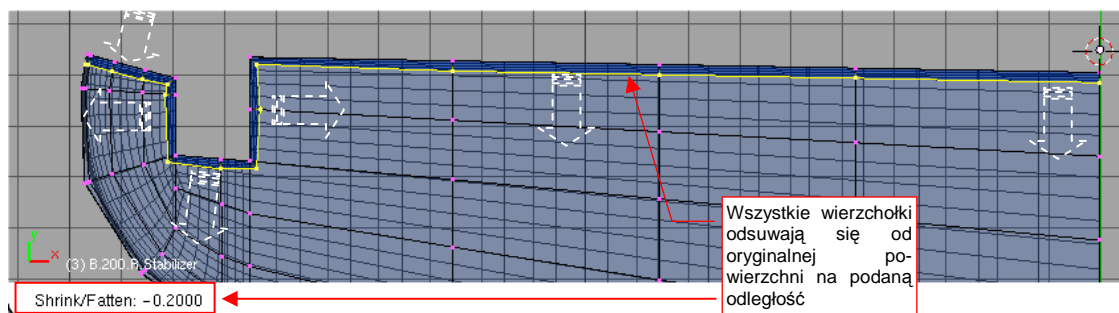
Transformacja polega na odsunięciu¹ wybranych wierzchołków od aktualnej powierzchni. Każdy wierzchołek jest przemieszczony wzdłuż normalnych do powierzchni, na określoną odległość. Za pomocą odsunięcia można łatwo uzyskać takie efekty, jak nadanie powłoce grubości, czy wytłoczenie regularnych wgłębień (np. zeber lub podłużnic).

Operację zaczynamy od zaznaczenia wierzchołków, które mają być odsunięte. Często zaznaczone wierzchołki powstały w wyniku wcześniejszego wytłoczenia (**E,1** — *Extrude→Region*), podczas których zrezygnowaliśmy z domyślnie proponowanego przesunięcia (**Esc**). Rysunek 15.25.1a, b) pokazuje przykład zaznaczonych ścian po takiej właśnie operacji wytłoczenia "bez przesunięcia":



Rysunek 15.25.1 Przygotowanie do operacji — zaznaczenie ścian, które mają być odsunięte

Następnie naciskamy **Alt-S** (*Mesh→Transform→Shrink/Fatten Along Normals*) (Rysunek 15.25.3):

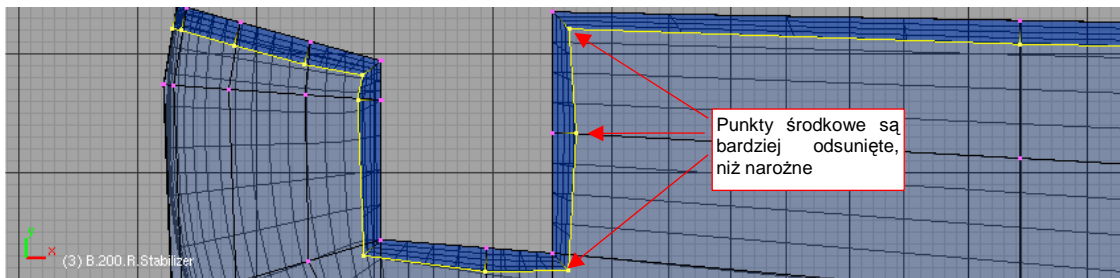


Rysunek 15.25.2 Przebieg operacji — wierzchołki przesuwać się wzdłuż normalnych do oryginalnej powłoki

Sterujesz dystansem odsunięcia przesuując myszkę. W trakcie ustalania tej odległości możesz się także posłużyć standardowymi klawiszami "modyfikującymi" Blendera. Trzymanie wciśniętego **Ctrl** powoduje, że dystans zmienia się skokowo, a **Shift** — zwiększa dokładność ruchu.

¹ Kalką językową z angielskiej nazwy — *Shrink/Fatten* — byłoby "skurczenie/pogrubienie", ale taką nazwą po polsku jest jakoś nieporęcznie operować. Przeszukałem dotychczasowe publikacje na temat Blendera, i znalazłem tylko w jednym miejscu "*Blender-Kompedium*" ([2]) inne określenie: "ekspansja". Wydało mi się jednak zbyt jednostronne.

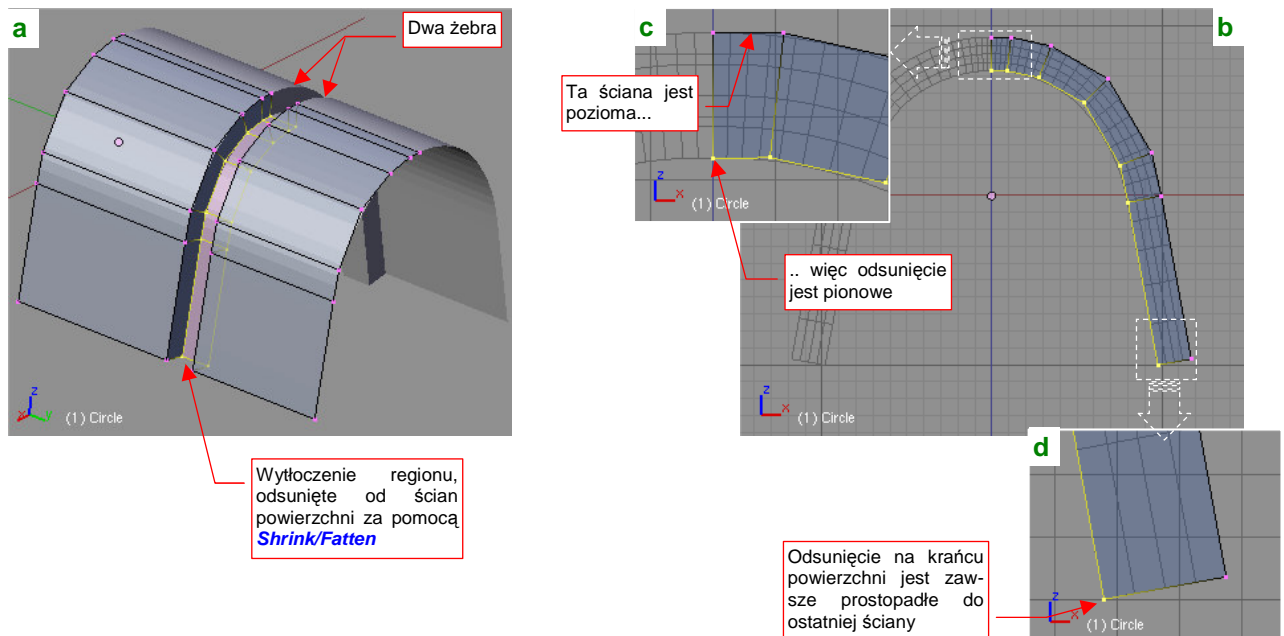
Rysunek 15.25.3 pokazuje efekt, który może wystąpić w przypadku odsuwania wierzchołków od płaskich ścian z ostrymi narożnikami:



Rysunek 15.25.3 Deformacja płaskich ścian w wyniku odsunięcia

Wierzchołki, które leżały w środku płaskich ścian, zostały odsunięte "głębiej" od oryginalnej powierzchni niż wierzchołki narożne. Mimo pozorów, wszystkie wierzchołki zostały przemieszczone na tę samą odległość (0.2 jedn. Blendera). Dysproporcja wynika z różnych kierunków normalnych. Wektory normalne w narożnikach są pochylone pod kątem 45° , więc wierzchołki narożne przesunęły się o 0.2 jedn. pod kątem 45° . Punkty środkowe miały normalne skierowane prostopadle, więc odsunęły się na pełen dystans. W efekcie powstała deformacja, którą trzeba skorygować, "spłaszczając" z powrotem ściany. (Poprzez zmianę skali wzdłuż jednej z osi: **X** lub **Y**).

O wiele lepiej transformacja *Shrink/Fatten* sprawdza się na obłych kształtach. Rysunek 15.25.4a) przedstawia typowy przypadek takiej powierzchni — symetryczną połowę jakiejś powłoki:



Rysunek 15.25.4 Odsunięcie ścian na typowej, obłej powierzchni

W środku powłoki znajdują się dwa żebra. Zazaczyłem ściany pomiędzy nimi i wytłoczyłem je (*Extrude*). Zrezygnowałem z przesuwania (**Esc**) i wywołałem odsunięcie (**Alt-S**).

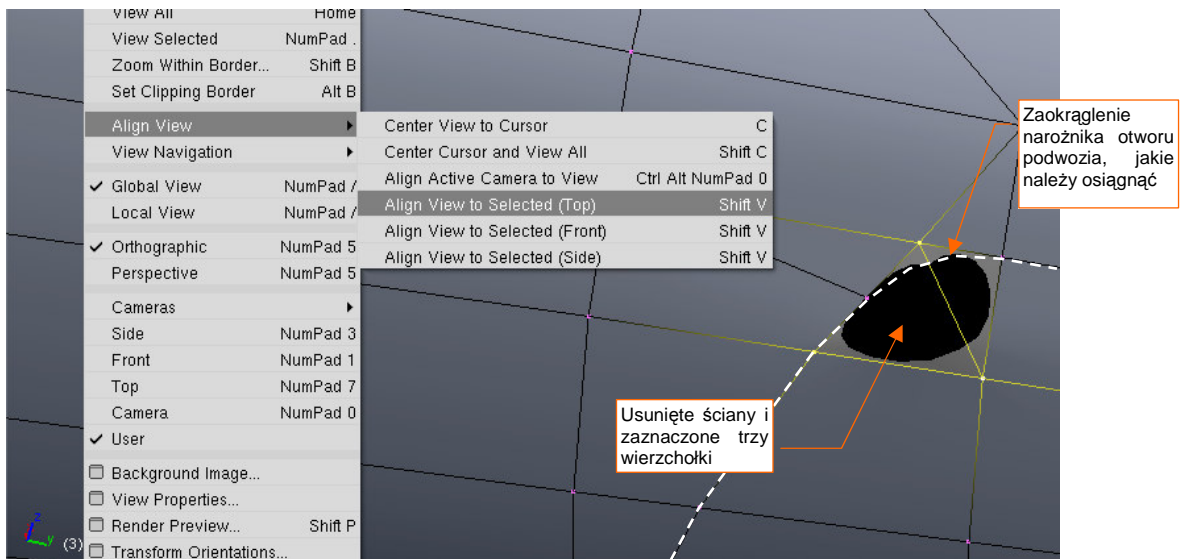
Rysunek 15.25.4b) pokazuje, jak należy przygotować krawędzie siatki, by odsunięcie przebiegło zgodnie z naszymi intencjami. Wykorzystałem tu fakt, że na krańcach powierzchni kierunek odsunięcia jest zawsze prostopadły do ostatniej ściany. Aby przebiegał wzdłuż płaszczyzny lustrzanego odbicia siatki, umieściłem na tym krańcu powierzchni niewielką, poziomą ścianę Rysunek 15.25.4c). Jej istnienie wymusza odsunięcie końcowego wierzchołka pionowo do dołu. Dla porównania przyjrzyj się drugiemu, "wolnemu" końcowi powierzchni (Rysunek 15.25.4d). Tu niczego specjalnego nie przygotowywałem, więc zaznaczony wierzchołek jest odsunięty pod pewnym kątem.

- Aby zobaczyć, jak skierowane są wektory normalne wierzchołków, przełącz w zestawie paneli **Edycji**, w panelu **Mesh Tools More**, przełącznik **Draw VNormals**.
- Operacja odsunięcia jest jednorazową. Po odsunięciu fragmentu powłoki zmienia się układ normalnych. Jeżeli chcesz wrócić do poprzedniego stanu — użyj polecenia **Undo** (**Ctrl-Z**).
- Alternatywną metodą odsunięcia powierzchni jest użycie skryptu **Solidify Selection** — p. str. 897.

15.26 Dopasowanie widoku do wybranej ściany (*Align View*) — zastosowanie

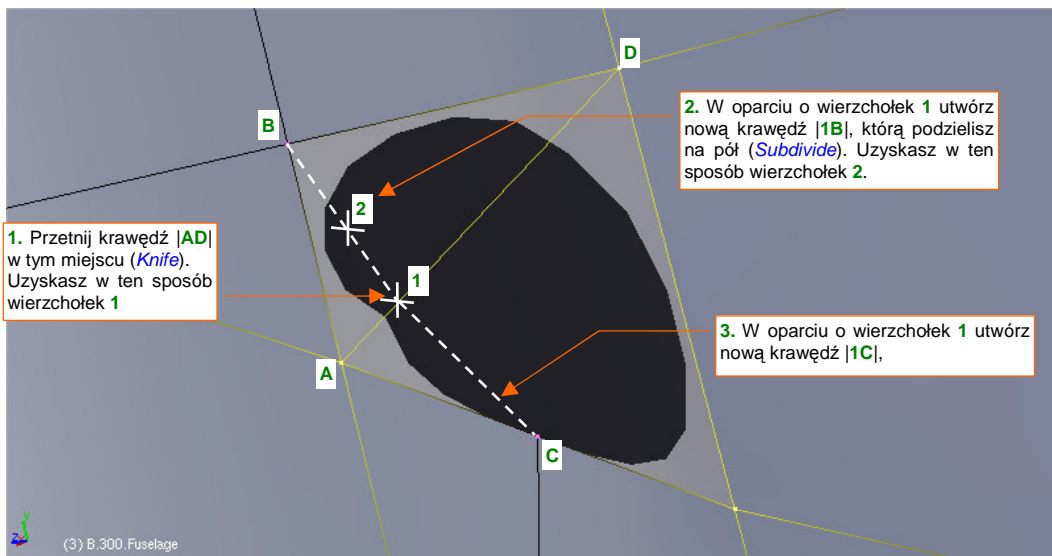
Czasami, podczas formowania otworu, pojawia się potrzeba "narysowania" linii łamanej, której wszystkie wierzchołki będą leżeć dokładnie na powierzchni wybranej ściany. Przykładem takiej sytuacji jest przygotowywanie otworu na kółko ogonowe w kadłubie. Otwór ten miał duży promień zaokrąglenia narożników (Rysunek 15.26.1). Musimy ten efekt odtworzyć.

Zacznij od przygotowania operacji: usuń (**X**, *Erase→Only Faces*) jedną — dwie ściany w obszarze, gdzie ma powstać zaokrąglenie (Rysunek 15.26.1). Następnie zaznacz trzy wierzchołki ściany, na której powierzchni chcesz rysować. (Trzy punkty jednoznacznie wyznaczają płaszczyznę). Potem wywołaj polecenie *View→Align View→Align View to Selected (Top)* (**Shift-V**):



Rysunek 15.26.1 Wywołanie dopasowania widoku do zaznaczonych wierzchołków

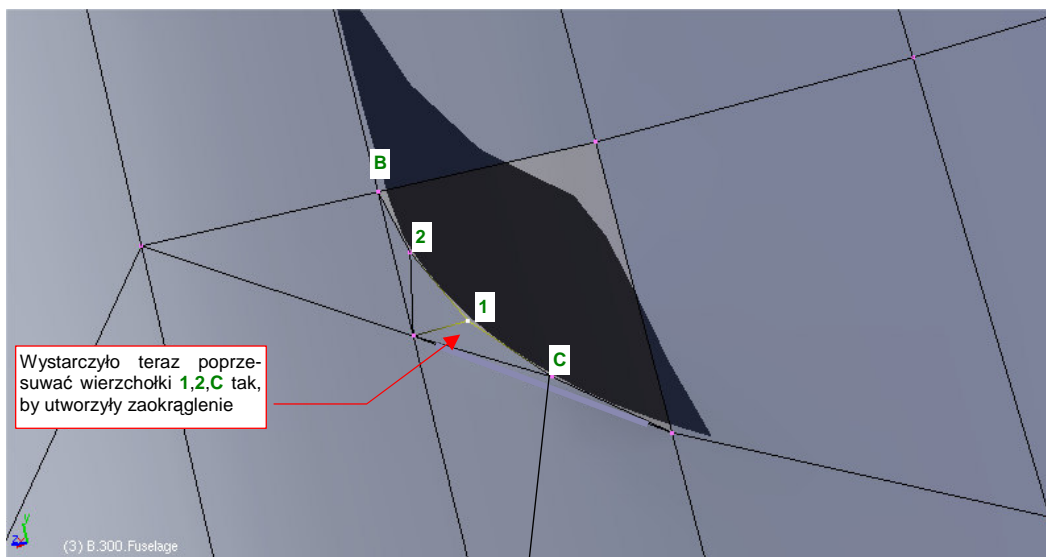
W rezultacie płaszczyzna aktualnego widoku została ustawiona równoległe do ściany (Rysunek 15.26.2):



Rysunek 15.26.2 Widok dopasowany do zaznaczonej ściany (a właściwie — wierzchołków)

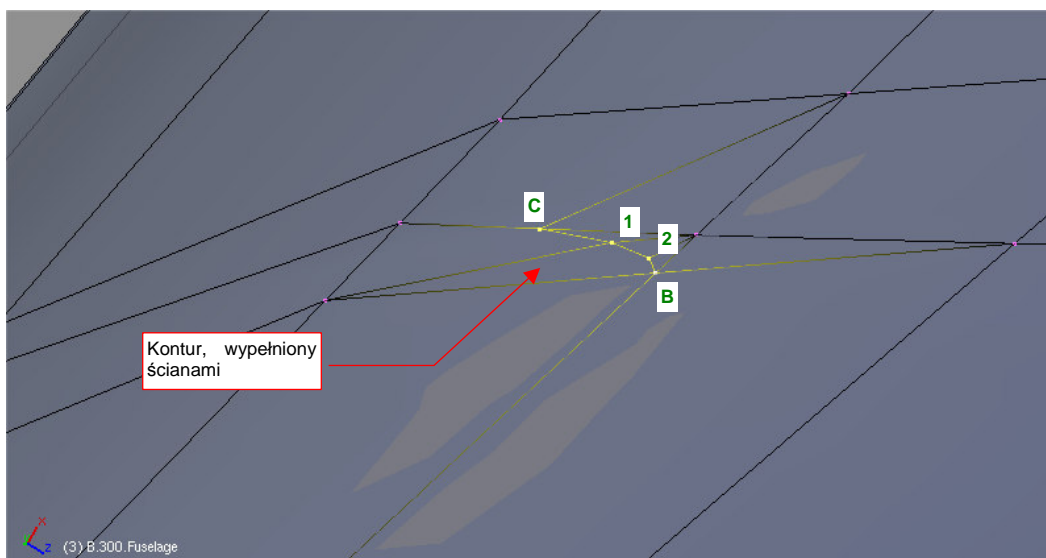
Przetnij krawędź |AD| (*Knife*), aby uzyskać wierzchołek 1. W oparciu o wierzchołek 1 możesz utworzyć dwie kolejne krawędzie — |1B| i |1C|. Z podziału krawędzi |1B| uzyskasz kolejny wierzchołek, potrzebny do stworzenia zaokrąglenia: 2.

Utworzone w ten sposób wierzchołki na pewno leżą na płaszczyźnie ściany. Korzystając z tego, że znajdujemy się na płaszczyźnie równoległej, wystarczy je zwyczajnie poprzesuwać (*Grab*) tak, by uformowały zaokrąglenie (Rysunek 15.26.3):



Rysunek 15.26.3 Wierzchołki C,1,2, ukształtowane w zaokrągloną krawędź

Jednoczesny widok z innej projekcji potwierdza, że narysowany kontur leży w płaszczyźnie ściany, tak jak chcieliśmy (Rysunek 15.26.4):



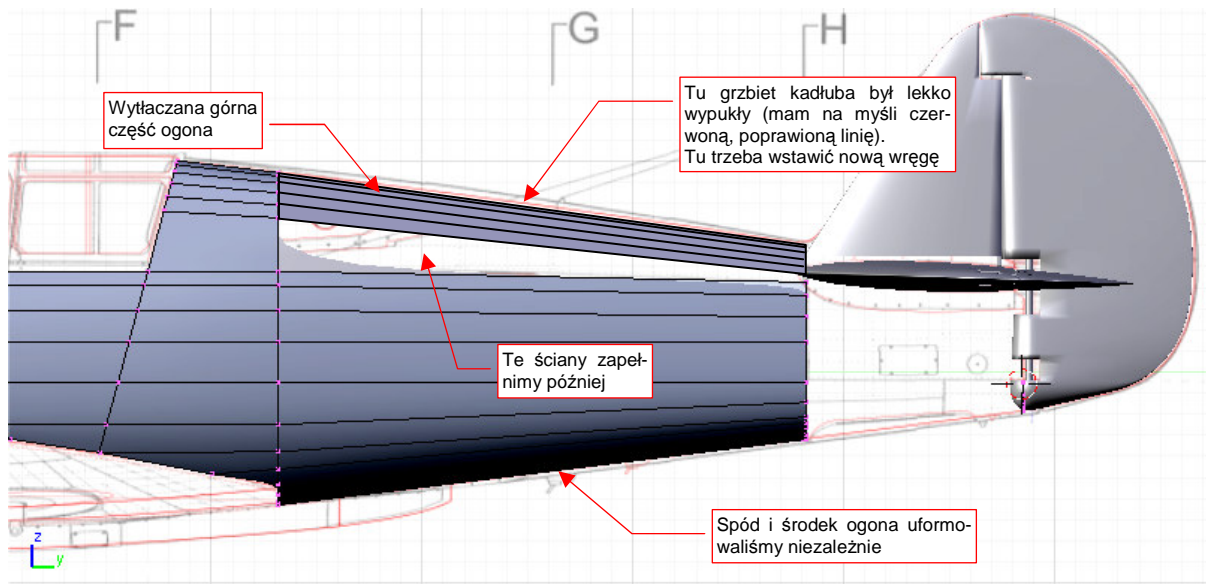
Rysunek 15.26.4 Utworzony kształt, widziany z innej projekcji

15.27 Dopasowanie wręgi do kadłuba

Kadłub proponuję formować metodą "ewolucyjną". Polega to na:

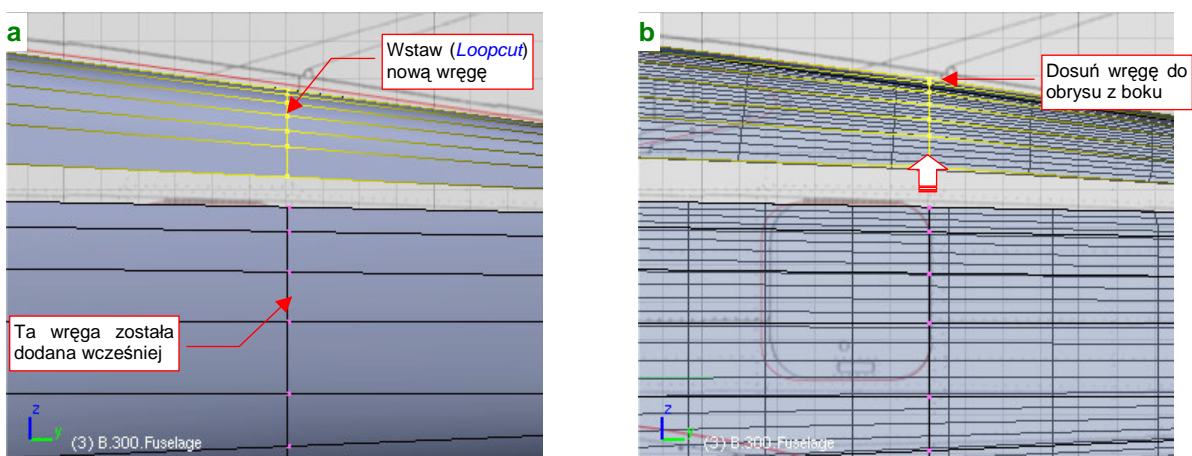
- wytłoczeniu na całą długość jakiegoś dużego, jednorodnego elementu;
- stopniowej modyfikacji takiej "rury" poprzez wstawianie w środek kolejnych wręg. Nadaję za ich pomocą niezbędną wypukłość, tam, gdzie powinna wystąpić.

W tej sekcji chcę pokazać typowe postępowanie podczas dodawania kolejnej wręgi. Przedstawię to na przykładzie formowania górnej części ogona P-40. Kształtujemy ją niezależnie od części środkowej i dolnej. Zaczniemy od wytłoczenia całej długości, do statecznika pionowego (Rysunek 15.27.1):



Rysunek 15.27.1 Wytłoczony zarys ogona P-40

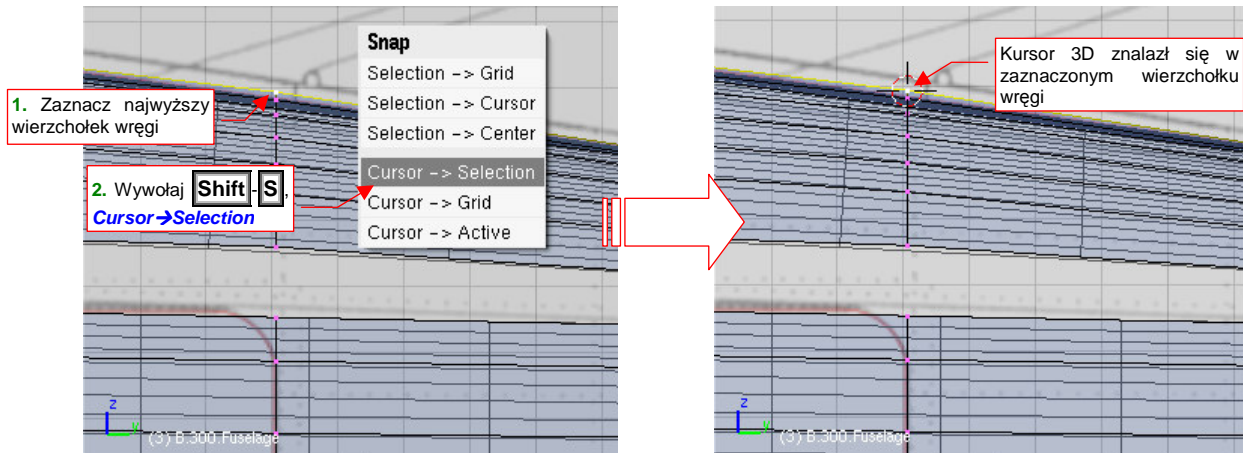
Uzyskany obrys "odstaje" od rzutu z boku w środku długości (mam na myśli czerwoną, poprawioną linię — p. Rysunek 15.27.1). Wstaw więc w tym miejscu nową wręgę (Rysunek 15.27.2a):



Rysunek 15.27.2 Dodanie i dosunięcie do obrysu nowej wręgi

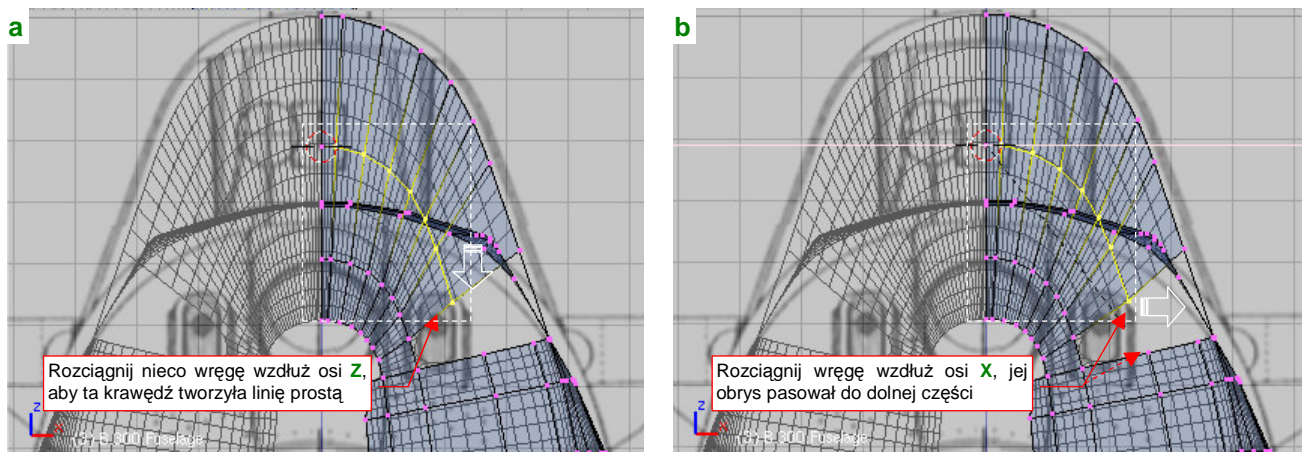
Po wstawieniu przesuń ją do góry (wzdłuż osi **Z**) tak, by obrys kadłuba z boku pokrył się z obrysem na planach (Rysunek 15.27.2b).

Teraz umieść kursor 3D w najwyższym wierzchołku wręgi. Najszybciej i najdokładniej jest to zrobić zaznaczając ten wierzchołek, i wywołując polecenie **Shift-S** (**Snap**), **Cursor→Selection** (Rysunek 15.27.3):



Rysunek 15.27.3 Umieszczenie kursora 3D w najwyższym wierzchołku wręgi

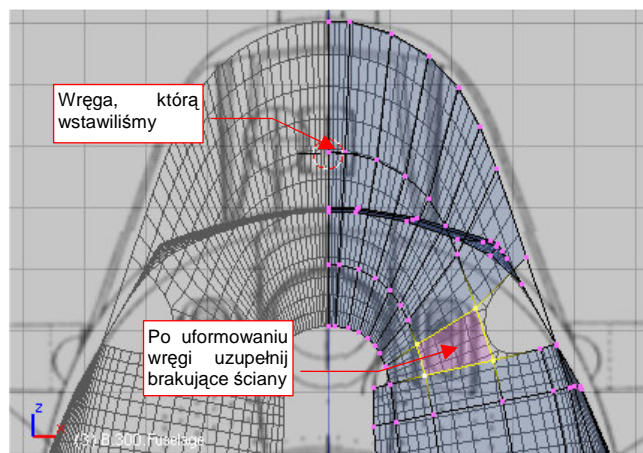
(Zakładam, że najwyższy punkt wręgi leży jednocześnie w osi kadłuba samolotu. Zazwyczaj tak jest). Teraz możesz wygodnie zmienić, poprzez zmianę skali względem kursora 3D, wysokość i szerokość wręgi (Rysunek 15.27.4a,b):



Rysunek 15.27.4 Dopasowanie wysokości i szerokości wręgi do reszty kadłuba

Środek transformacji leży na górnej krawędzi kadłuba. Dzięki temu przeskalowywanie wzdłuż osi **Z** nie zagraża zgodności z obrysem z boku.

Poprawność uformowania wręgi warto jest także ocenić w rzucie z przodu (Rysunek 15.27.5). Podłużnice, które na nim widzimy, powinny wykazywać stopniowe przekształcanie, bez żadnych nagłych załamań.

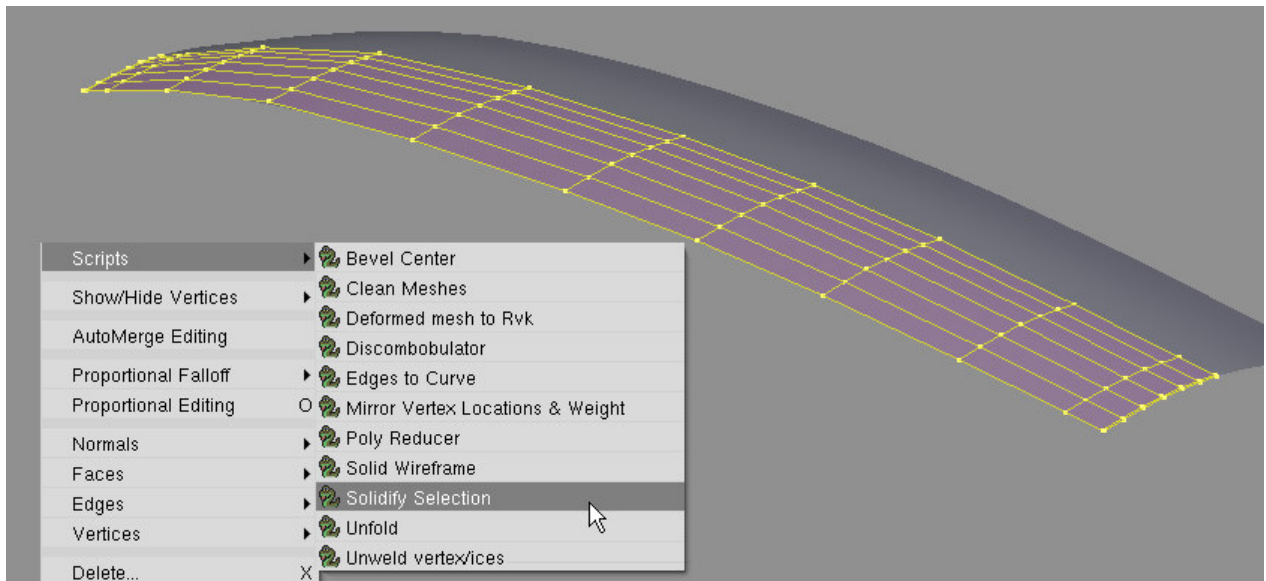


Rysunek 15.27.5 Uzupełnianie brakujących ścian kadłuba

15.28 Pogrubienie (*Solidify Selection*)

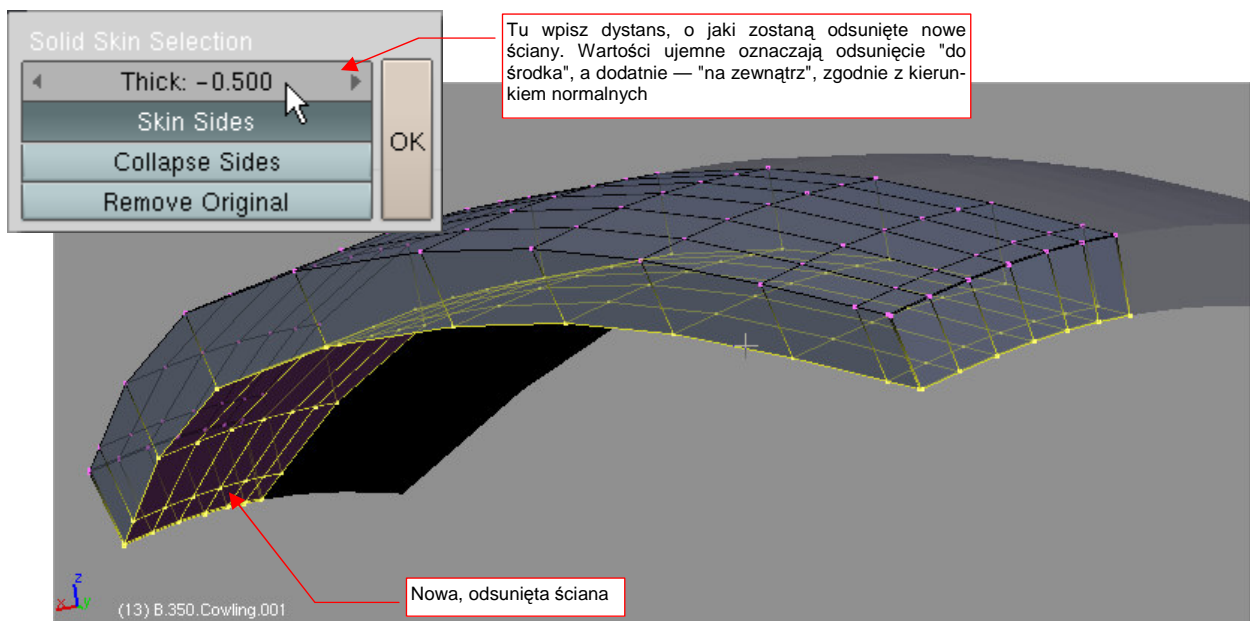
Czasami pogrubienie obiektu za pomocą polecenia *Shrink/Fatten* (str. 890) może nie być poprawne. Dzieje się tak np. dla długiej, wąskiej, wygiętej, powłoki — na jednym z krańców powstaną silnie pochylone ściany (Rysunek 15.28.1).

Na szczęście możemy ją pogubić w inny sposób — za pomocą skryptu *Solidify Selection*. Zaznacz najpierw obszar, dla którego chcesz uzyskać ten efekt (to nie musi być cała powłoka). Następnie wywołaj polecenie *Mesh → Scripts → Solidify Selection* (Rysunek 15.28.1):



Rysunek 15.28.1 Wywołanie skryptu *Solidify Selection*

Na ekranie pojawi się okno dialogowe skryptu (Rysunek 15.28.2). Ustaw odpowiednio dystans odsunięcia (w polu *Thick*). Wartości ujemne oznaczają odsunięcie "w głąb" powłoki. Po naciśnięciu *OK* zobaczysz rezultat — nową, odsuniętą powłokę.

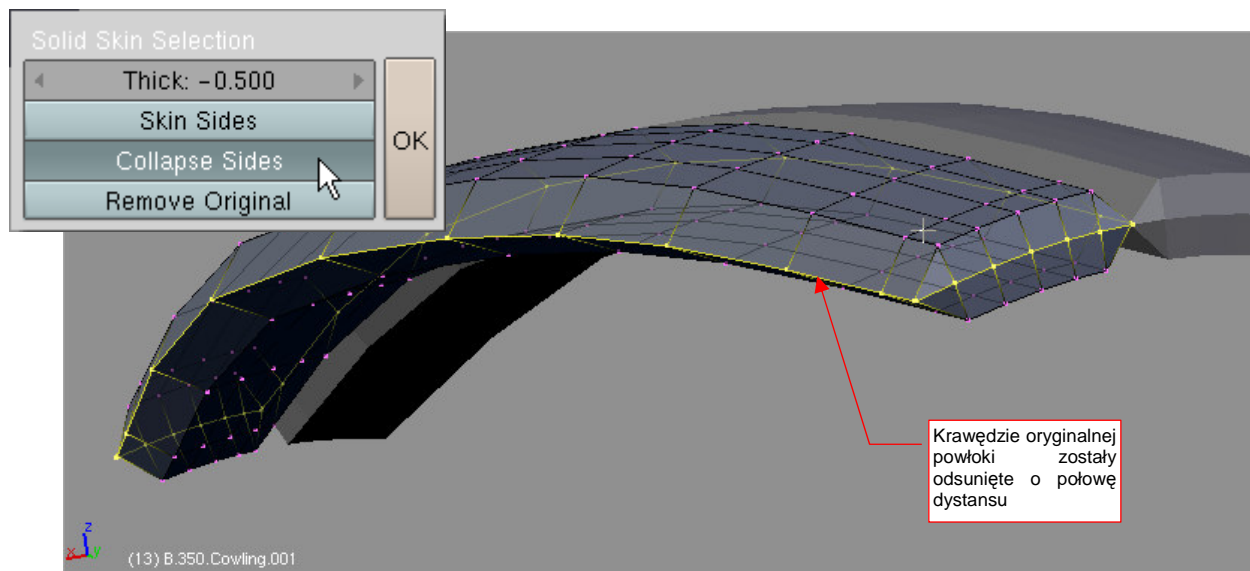


Rysunek 15.28.2 Ustawienie parametrów i efekt działania skryptu *Solidify Selection*

Zwróć uwagę, że nowa powłoka jest połączona z oryginalną ścianami bocznymi. To efekt pozostawienia domyślnego ustawienia przełącznika *Skin Sides* (włączone).

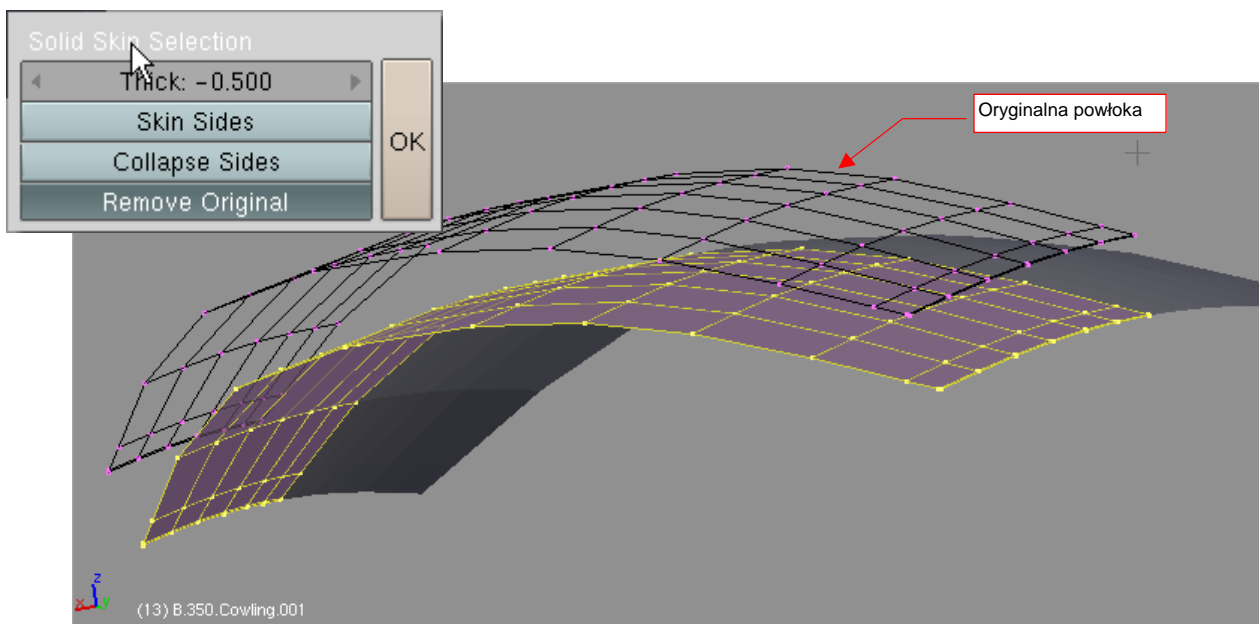
Domyślne ustawienia przełączników skryptu zapewniają uzyskanie typowego efektu — zamiany pojedynczej otwartej powłoki, na powłokę zamkniętą. Dla porządku pokażę jednak działanie pozostałych dwóch przełączników skryptu *Solidify Selection*. Być może kiedyś Ci się przydadzą.

Rysunek 15.28.3 pokazuje efekt włączenia *Collapse Sides*. Zamiast tworzyć wzdłuż krawędzi oryginalnej powłoki nowe ściany, skrypt odsunął ich wierzchołkom o połowę dystansu *Thick*. Uzyskany efekt nazwałbym "efektem poduszki".



Rysunek 15.28.3 Efekt włączenia *Collapse Sides* — krawędzie siatki odsunięte o połowę dystansu

Rysunek 15.28.4 pokazuje efekt włączenia *Remove Original*. Ściany (ale nie krawędzie!) oryginalnej powłoki zostały usunięte. Mówiąc szczerze, nie jest to zbyt praktyczne, gdyż użytkownik nadal musi usunąć pozostałe krawędzie w tradycyjny sposób.

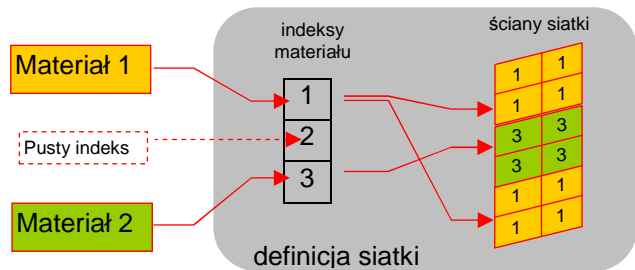


Rysunek 15.28.4 Efekt włączenia *RemoveOriginal* — usunięte oryginalne ściany

15.29 Przepisanie do siatki dodatkowego materiału

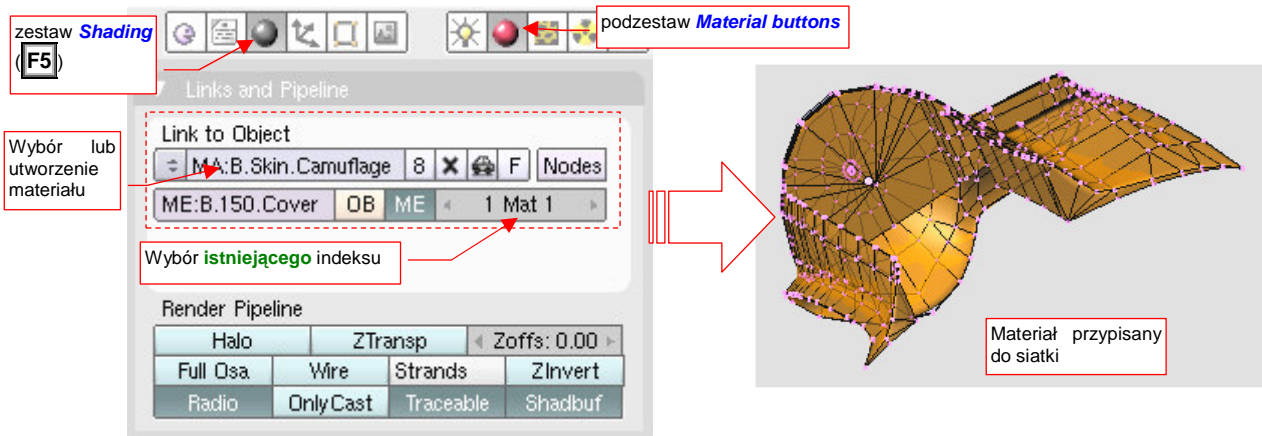
Materiały są przypisane do ścian siatki za pośrednictwem tzw. indeksów (*material index* — Rysunek 15.29.1).

W każdej siatce możesz stworzyć do 16 indeksów materiałów. Każda ściana jest przypisana tylko do jednego indeksu. Niektóre z indeksów mogą nie mieć przypisanych ścian. Mogą także nie mieć przypisanych materiałów. (Takie "zdegenerowane" indeksy nie mają wpływu na rendering).



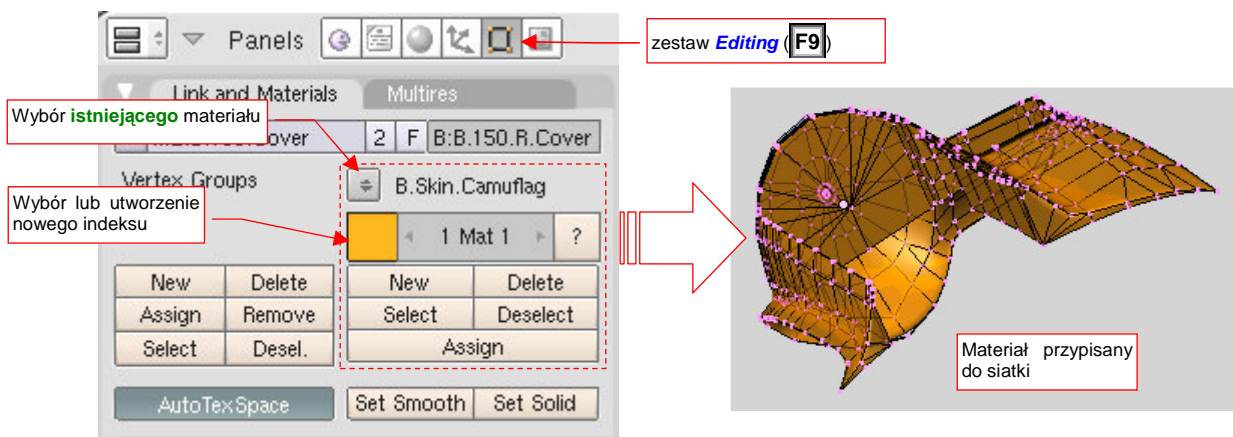
Rysunek 15.29.1 Powiązanie ścian siatki z materiałem (*indeksy*)

W Blenderze istnieją dwie alternatywne metody przypisywania materiału do siatki. Pierwsza (Rysunek 15.29.2) jest dostępna w zestawie *Shading:Material buttons* (por. str. 958, Rysunek 17.1.5). Panel *Links and Pipeline* umożliwia tam utworzenie nowego materiału, lub wybranie jednego z istniejących, i przypisanie go do istniejącego już na siatce indeksu. Wyjątkowo, gdy siatka nie ma jeszcze w ogóle żadnego indeksu materiału, Blender automatycznie tworzy dla niego nowy — nr 1. Przepisuje także do niego wszystkie ściany siatki. To najwygodniejszy sposób na przypisanie do siatki pierwszego materiału (Rysunek 15.29.2):



Rysunek 15.29.2 Pierwsza metoda: przypisanie materiału do istniejącego indeksu siatki — w zestawie *Shading*.

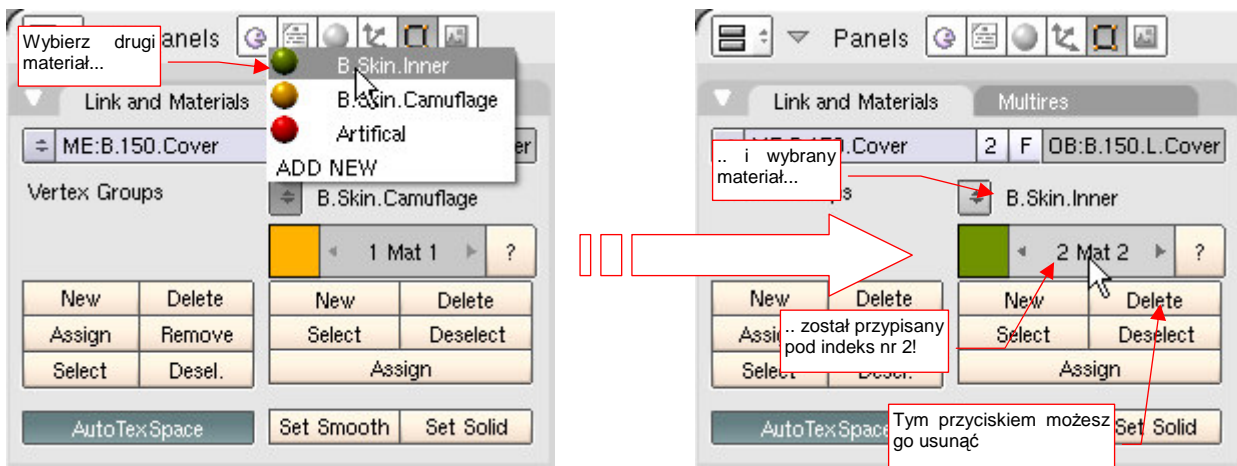
Alternatywna metoda jest dostępna w panelu *Link and Materials*, zestawu *Editing* (Rysunek 15.29.3):



Rysunek 15.29.3 Druga metoda: przypisanie indeksu siatki do istniejącego materiału w zestawie *Editing*.

Panel *Link and Materials* umożliwia pełną kontrolę nad indeksami materiału siatki. Można tu je tworzyć, usuwać, oraz zmieniać przypisanie ścian. Za to z materiałem nie można tu nic więcej zrobić, jak tylko wybrać z listy istniejących. Ten panel najlepiej się nadaje do przypisywania do siatki dodatkowych indeksów materiału.

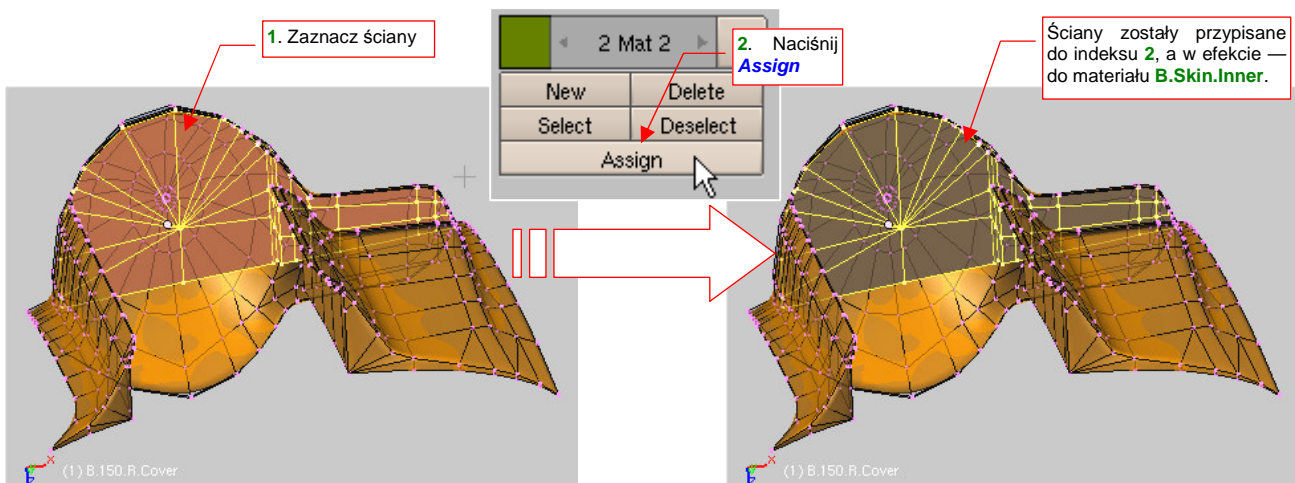
Aby dodać dodatkowy materiał do siatki, trzeba przygotować mu "miejsce" — nowy indeks. Stwórz go w panelu **Link and Materials**, wybierając z listy¹ nowy materiał (Rysunek 15.29.4):



Rysunek 15.29.4 Stworzenie nowego indeksu dla materiału.

- Listę materiałów, którą pokazuje Rysunek 15.29.4, można rozwinąć tylko wtedy, gdy siatka jest w trybie edycji (**Edit Mode**)!
- Przyciskiem **Delete**, który pokazuje Rysunek 15.29.4, można usunąć aktualny indeks materiału. Można to zrobić jednak jedynie w trybie obiektu (**Object Mode**)!

Do nowego indeksu należy teraz przypisać ściany. Zaznaczasz je w oknie **View 3D**, a następnie naciskasz w panelu **Link and Materials** przycisk **Assign** (Rysunek 15.29.5):



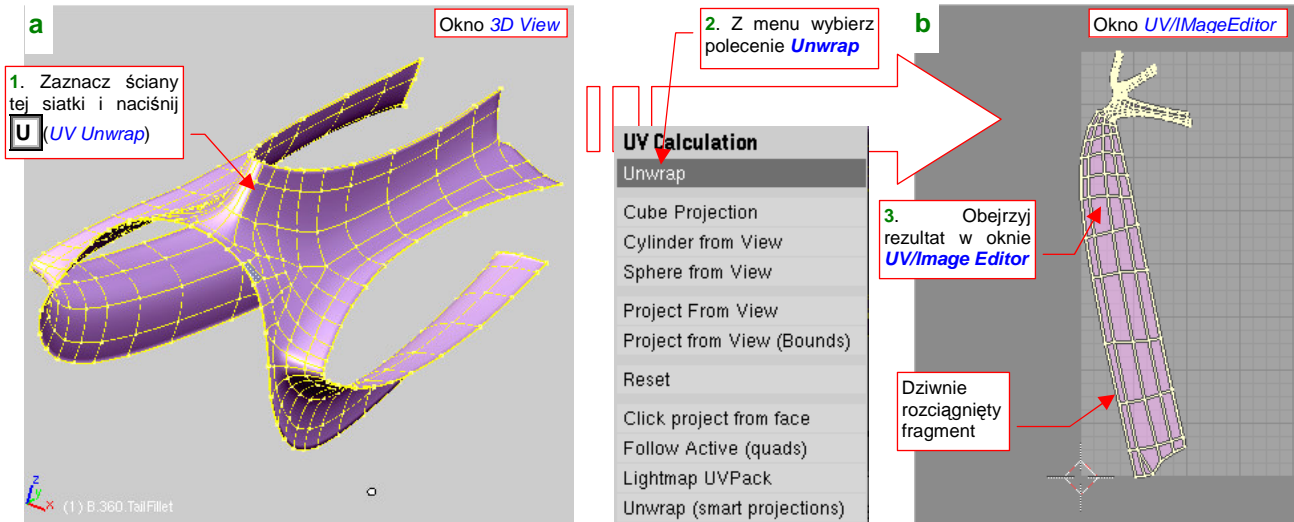
Rysunek 15.29.5 Przypisanie ścian do nowego indeksu materiału.

Nie musisz zaznaczać od razu wszystkich ścian. Możesz tę operację powtórzyć kilka razy (w przypadku bardzo złożonych kształtów tak jest prościej)

¹ Naciśnięcie przycisku **New** także doda do siatki nowy indeks materiału. Tak się to robiło w starszych wersjach Blendera. Nie polecam tego jednak, gdyż domyślnie z nowym indeksem stworzonym przyciskiem **New** kojarzona jest kopia aktualnego materiału. Potem trzeba to zmienić

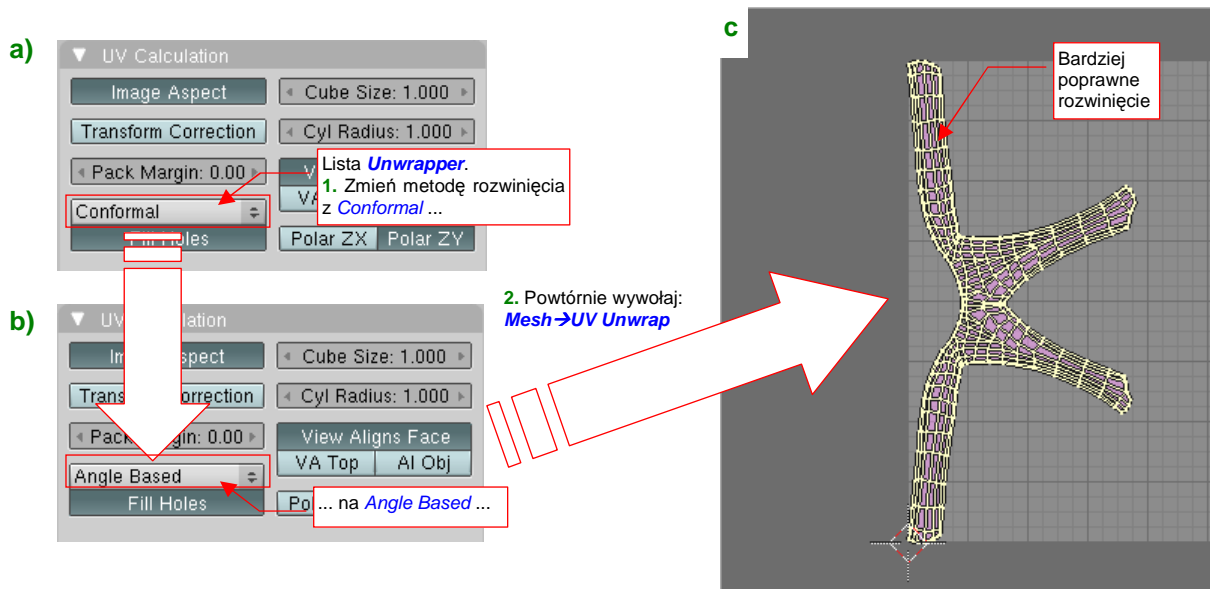
15.30 “Zwykłe” rozwijanie siatki (*Unwrap*)

Zaznacz ściany siatki, które chcesz rozwinąć (to może być nawet mały fragment, niekoniecznie cała siatka). Następnie naciśnij **U** (*Mesh→UV Unwrap*). Na ekranie pojawi się menu *UV Calculation*, z alternatywnymi metodami rozwijania siatek. Wybierz z niego opcję *Unwrap* (Rysunek 15.30.1):



Rysunek 15.30.1 Zwykłe rozwinięcie (*Unwrap*) siatki w przestrzeni UV

Rezultat rozwinięcia zobaczysz w oknie *UV/Image Editor* (p.str. 932). Jeżeli wyszło zdeformowane, np. tak jak pokazuje Rysunek 15.30.1b, możesz spróbować użyć alternatywną metodę rozwijania. Ustawiasz ją w zestawie *Editing* (**F9**), panel *UV Calculation*. Przyjrzyj się, co masz wybrane na liście *Unwrapper* (Rysunek 15.30.2a):

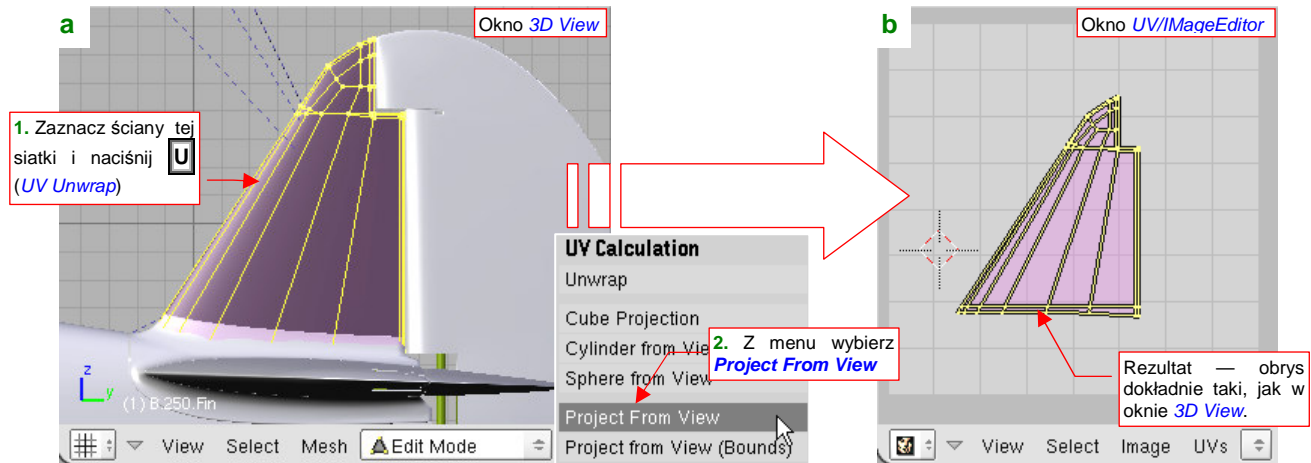


Rysunek 15.30.2 Sterowanie metodą rozwinięcia

Jak do tej pory, Blender oferuje dwie metody rozwijania „ogólnego”: *Conformal* i *Angle Based*. Jeżeli nie jesteś zadowolony z aktualnej, przestaw ją na alternatywę (Rysunek 15.30.2a). Następnie jeszcze raz wywołaj *Mesh→UV Unwrap:Unwrap*. Rezultat może ulec poprawie (porównaj Rysunek 15.30.2c i Rysunek 15.30.1b). Jeżeli nadal nie jesteś zadowolony — pozostaje poprawić rozwinięcie „ręcznie”, tzn. poprzez zmianę położenia ścian w edytorze UV.

15.31 Rozwijanie poprzez rzutowanie siatki (*Project from View*)

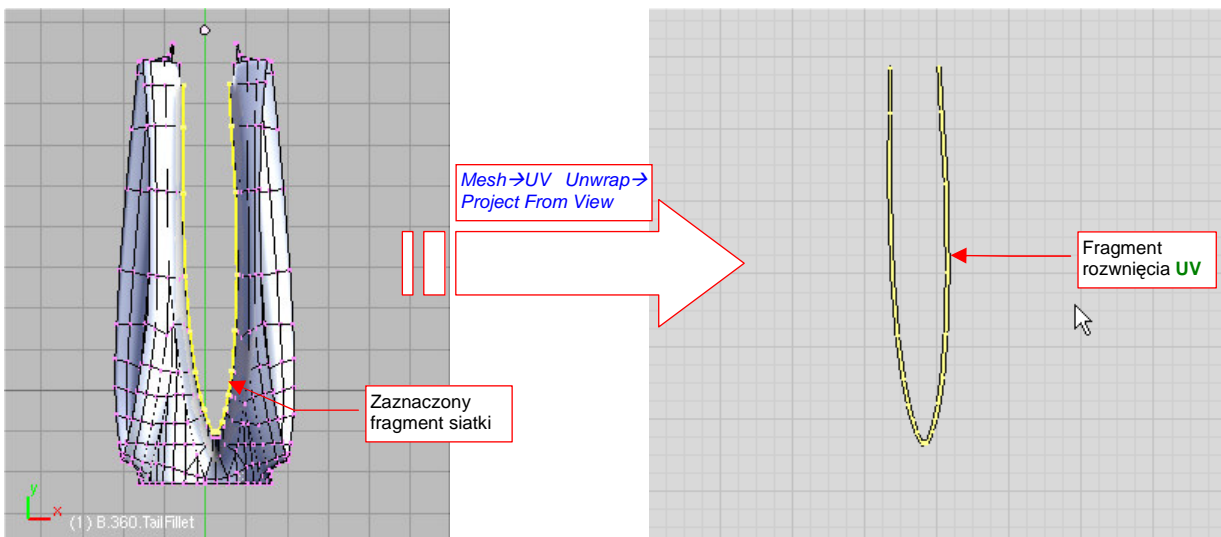
Zaznacz ściany siatki, które chcesz rozwinąć w przestrzeni **UV**. Następnie naciśnij **U** (*Mesh→UV Unwrap*). Na ekranie pojawi się menu *UV Calculation*, z alternatywnymi metodami rozwijania siatek. Wybierz z niego opcję *Project From View* (Rysunek 15.31.1):



Rysunek 15.31.1 Rozwinięcie UV poprzez rzutowanie na płaszczyznę widoku (*Project From View*)

Rezultat rozwinięcia zobaczysz w oknie *UV/Image Editor* (opis — str. 932). Siatka wygląda dokładnie tak samo, jak w oknie *3D View*, ale jest płaskim obrazem, umieszczonym na płaszczyźnie **UV**.

Za pomocą polecenia *Project From View* można rozwijać nie tylko całe siatki, ale także ich pojedyncze fragmenty. Przykład takiej operacji pokazuje Rysunek 15.31.2:



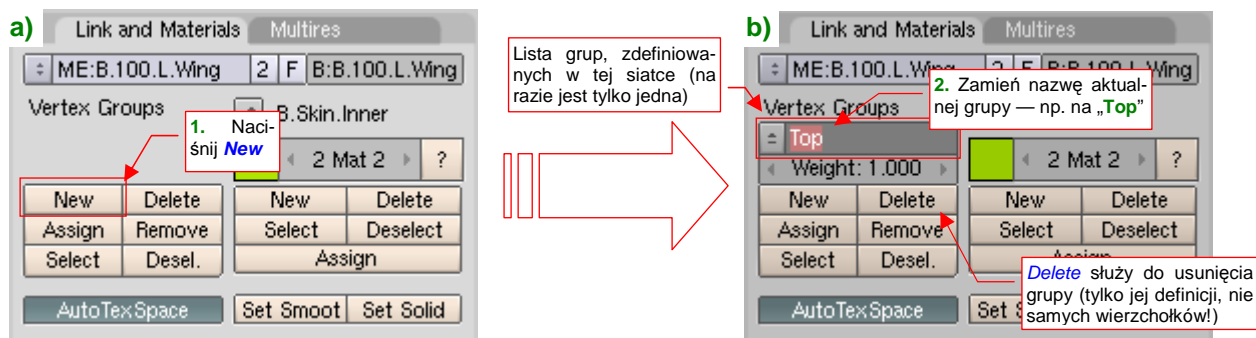
Rysunek 15.31.2 Rzutowanie na płaszczyznę UV fragmentu siatki

Gdy zaznaczysz na siatce w oknie *3D View* tylko jej fragment (jak np. krawędź owiewki, którą pokazuje Rysunek 15.31.2), i wywołasz polecenie *UV Unwrap→Project From View*, w oknie *UV/Image Editor* zobaczysz rzut tego fragmentu. Możesz go np. zaraz po rzutowaniu „przyszpilić” (*UVs→Pin* — patrz str. 943).

15.32 Definiowanie grupy wierzchołków (*Vertex Group*)

W Blenderze istnieje możliwość podzielenia wierzchołków siatki na grupy. Została stworzona z myślą o powiązaniu powierzchni modelu z animowanym szkieletem. Może się jednak przydać do „zapamiętania” jakiegoś skomplikowanego obszaru siatki, który będziesz często zaznaczał.

Nową grupę wierzchołków tworzysz w zestawie *Editing*, panelu *Link and Materials* (Rysunek 15.32.1):

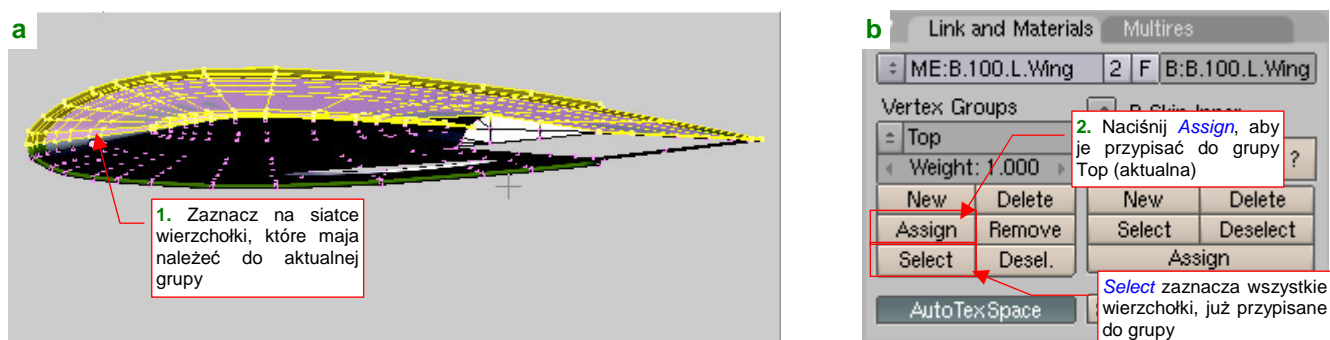


Rysunek 15.32.1 Stworzenie i zmiana nazwy nowej grupy wierzchołków

Po naciśnięciu przycisku **New** (mam na myśli ten z obszaru *Vertex Groups* — Rysunek 15.32.1a) Blender utworzy nową grupę. Poniżej etykiety *Vertex Groups* pojawi się dodatkowa kontrolka — lista rozwijalna wyboru aktualnej grupy (Rysunek 15.32.1b). Zobaczysz w niej napis **Group** (taka nazwa jest domyślnie nadawana każdej nowo utworzonej grupie) . Zmień ją na jakąś inną — na przykład **Top**.

- Do usunięcia aktualnej grupy, której nie potrzebujesz, służy przycisk **Delete**. Usunięciu ulegają tylko przypisania wierzchołków do grupy i sama nazwa grupy. Same wierzchołki siatki nie są, rzecz jasna, usuwane.

Na razie z grupą nie jest związany żaden wierzchołek. Zaznacz odpowiedni obszar siatki w oknie *3D View*, a następnie naciśnij w panelu *Link and Materials* przycisk **Assign**. To przypisze wybrane wierzchołki do aktualnej grupy (Rysunek 15.32.2):

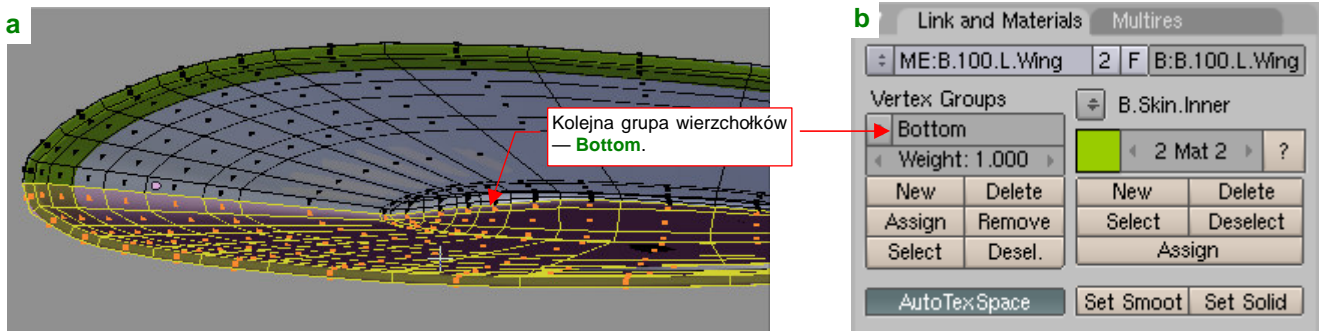


Rysunek 15.32.2 Stworzenie i zmiana nazwy nowej grupy wierzchołków

Samo przypisanie wierzchołków do grupy nie powoduje żadnej widocznej zmiany w oknie *3D View*. Aby się przekonać, czy są naprawdę w aktualnej grupie, wyłącz aktualne zaznaczenie (**A**). Następnie naciśnij przycisk **Select** (Rysunek 15.32.2b). Sprawdź, czy wszystkie wierzchołki, które powinny należeć do grupy, są zaznaczone. Jeżeli nie — zaznacz brakujące, i jeszcze raz naciśnij przycisk **Assign**.

- Przycisk **Select** dodaje do aktualnego zaznaczenia wszystkie wierzchołki grupy, a **Desel.** — wyłącza ich zaznaczenie.

Całą powierzchnię możesz podzielić na kilka grup. Skoro zdefiniowaliśmy już górną powierzchnię płata jako grupę, stwórzmy w ten sam sposób drugą, zawierającą „całą resztę”. Stwórz ją w ten sam sposób, co poprzednią. Nadaj jej nazwę **Bottom** (Rysunek 15.32.3b):



Rysunek 15.32.3 Przepisanie reszty siatki do drugiej grupy wierzchołków (**Bottom**)

Najprościej wybrać „całą resztę” siatki zaznaczając (przycisk **Select**) grupę **Top**. Potem należy przełączyć tryb wyświetlania z wierzchołków na ściany (por. str. 859), i z menu okna **3D View** wywołać polecenie **Selection→Inverse** (**Ctrl** - **I**). Powinieneś uzyskać takie zaznaczenie, jakie pokazuje Rysunek 15.32.3a). Na koniec pozostaje nacisnąć przycisk **Assign**, aby przypisać ten fragment siatki do grupy **Bottom**. Uważaj, aby nie zaznaczyć za dużo, gdyż Blender nie stawia tu przeszkód, i ten sam wierzchołek może należeć do wielu grup.

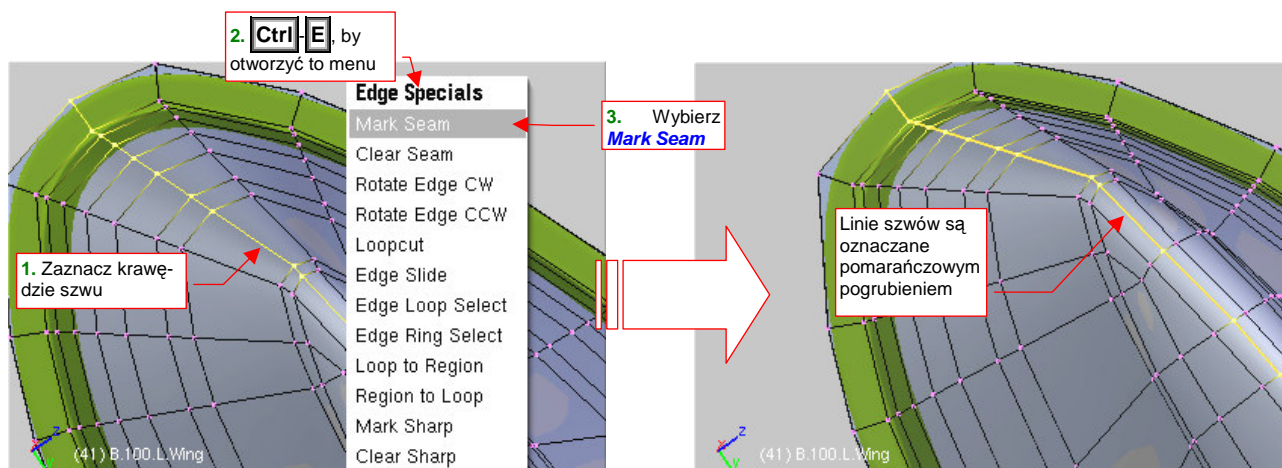
Na koniec kilka dodatkowych wskazówek:

- Wierzchołki siatki mogą należeć równocześnie do wielu grup wierzchołków.
- Do usunięcia wierzchołka z grupy służy przycisk **Remove**.
- Nazwy grup muszą być unikalne tylko w obrębie tej samej siatki. Oznacza to, że wiele różnych siatek w Twoim modelu może mieć grupy o takich samych nazwach — **Top**, **Bottom**, czy podobnych. To pewne ułatwienie, gdyż nie trzeba tworzyć jakiegoś kolejnego systemu nazw.

15.33 Zaznaczenie krawędzi jako szwu (*Seam*)

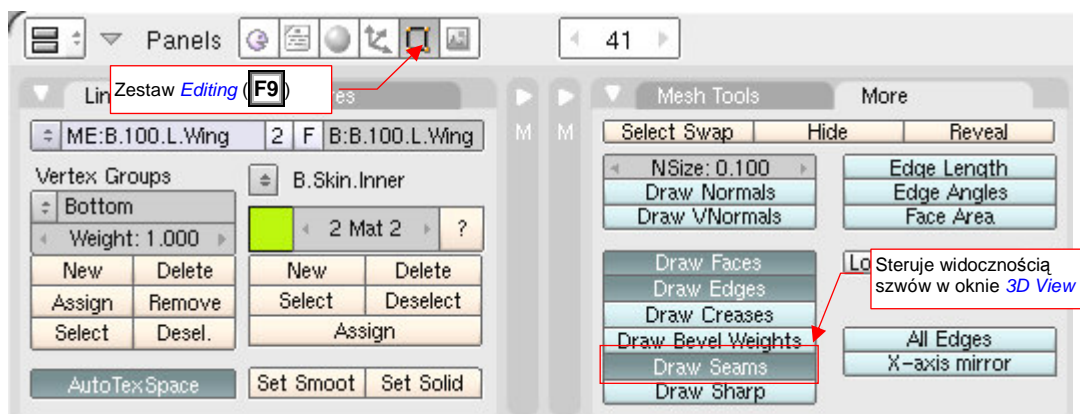
Podczas rozwinięcia UV, Blender automatycznie dzieli siatkę wzdłuż krawędzi, które oznaczysz jako „szwy” (*seams*).

Nanoszenie „szwów” jest bardzo proste: zaznacz krawędzie, wzdłuż których ma być podzielone rozwinięcie UV. Następnie naciśnij **Ctrl-E**, i z menu *Edge Specials* wybierz polecenie **Mark Seam** (Rysunek 15.33.1):



Rysunek 15.33.1 Oznaczenie krawędzi siatki jako linii szwów (*Mark Seam*)

(To samo polecenie znajdziesz także w menu: *Mesh*→*Edges*→*Mark Seam*). Szwy są na siatce oznaczane pomarańczowym pogrubieniem. Jeżeli nie widzisz żadnego dodatkowego oznaczenia linii szwów, które stworzyłeś — zajrzyj do zestawu *Editing* (**F9**), panelu *Mesh Tools - More* (Rysunek 15.33.2):



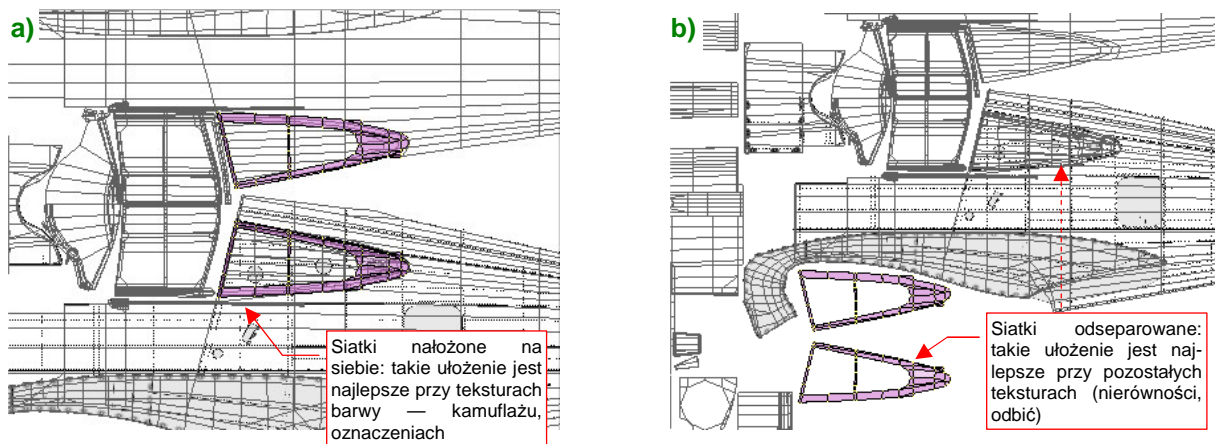
Rysunek 15.33.2 Sterowanie wyświetlaniem linii szwów

Sprawdź, czy w panelu *More* masz włączoną opcję **Draw Seams**. Jeżeli nie — możesz ją tu włączyć.

- Aby usunąć szwy z zaznaczonych krawędzi, użyj polecenia **Clear Seam**.

15.34 Dodanie alternatywnego rozwinięcia UV (*UV Layers*)

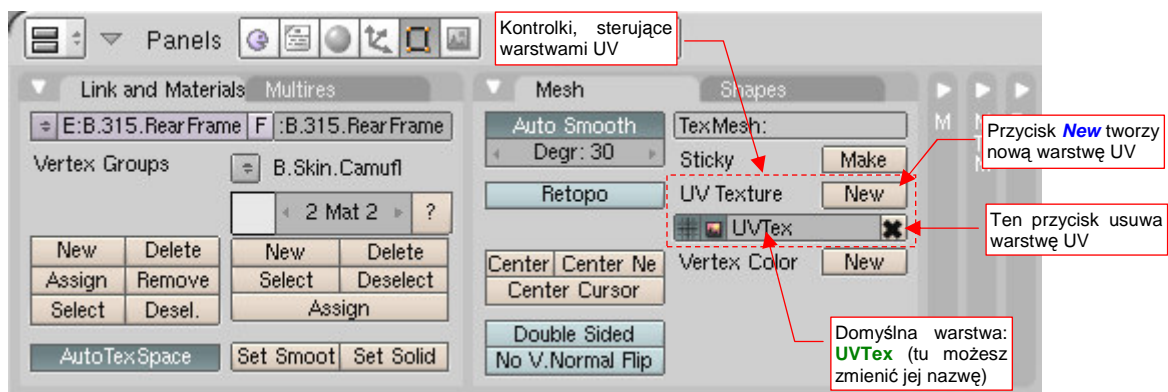
Czasami możesz natknąć się na sytuację, gdy dla dwóch różnych tekstur przydałyby się dwa różne rozwinięcia siatki w UV. Pokażę to na przykładzie ramki oszklenia za kabiną pilota P-40 (Rysunek 15.34.1):



Rysunek 15.34.1 Alternatywne rozwinięcia w UV siatki ramek oszklenia za głową pilota

Przy nakładaniu barw — plam kamuflażu, znaków rozpoznawczych, itp. — najlepiej, by siatka ramki znajdowała się tam, gdzie była przymocowana: na powierzchni kadłuba (Rysunek 15.34.1a). Przy takim ustawieniu jednym pociągnięciem w Gimpie namalujesz idealnie dopasowany kamuflaż na obydwu elementach. Z kolei dla tekstury nierówności (czy też odbić) lepiej byłoby, gdyby rozwinięcia kadłuba i ramki były odseparowane (Rysunek 15.34.1a). Każdy z tych elementów ma swój własny układ linii nitów i innych detali konstrukcyjnych. Gdy jedna siatka leży na drugiej — nie jesteś w stanie ich prawidłowo odwzorować (bo na każdej z powierzchni zobaczysz naraz wszystkie linie).

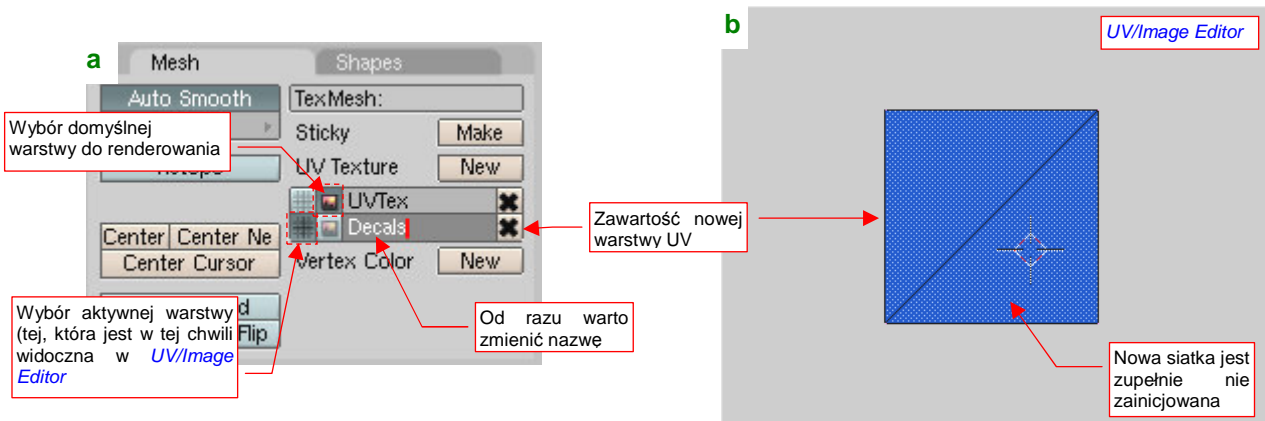
Na szczęście już kilka lat temu wprowadzono w Blenderze możliwość używania dla jednej siatki kilku alternatywnych rozwinięć UV. Są to tak zwane *warstwy UV (UV Layers)*. Każda warstwa zawiera kompletną informację o współrzędnych UV każdego wierzchołka siatki, a także o przypisanym do każdej ze ścian obrazie. Nie wiem, czy określenie „warstwy” nie jest trochę mylące: nie nakładają się w żaden sposób, żadna z nich nie zależy także od pozostałych. Do zarządzania warstwami UV służy kilka niewielkich kontrolerek, wciśniętych (tak, to dobre określenie) w panel *Mesh*, zestawu *Editing* (Rysunek 15.34.2):



Rysunek 15.34.2 Kontrolki, sterujące warstwami UV (*UV Layers*)

Gdy dokonałeś w edytorze UV rozwinięcia siatki, Blender automatycznie utworzył pierwszą warstwę, o nazwie **UVTex** (Rysunek 15.34.2). Gdy naciśniesz przycisk **New** (ten, który pokazuje objaśnienie — por. Rysunek 15.34.2), utworzysz nową. Możesz tu także zmienić nazwę warstwy (pole z nazwą to w istocie pole edycji). Nazwy warstw muszą być unikalne tylko w obrębie pojedynczej siatki. Mogą (a właściwie często muszą — zobacz dalej) powtarzać się w każdej siatce. Przycisk **X**, umieszczony z prawej, umożliwi usunięcie warstwy UV.

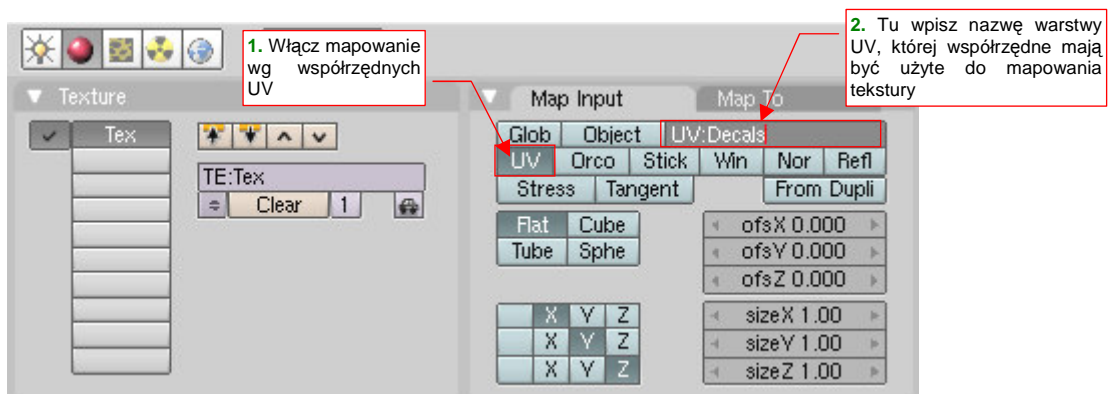
Gdy naciśniesz przycisk **New**, poniżej warstwy **UVTex** pojawi się nowa warstwa, **UVTex.001**. Zmień jej od razu nazwę — np. na **Decals** (Rysunek 15.34.3a):



Rysunek 15.34.3 Nowa warstwa UV

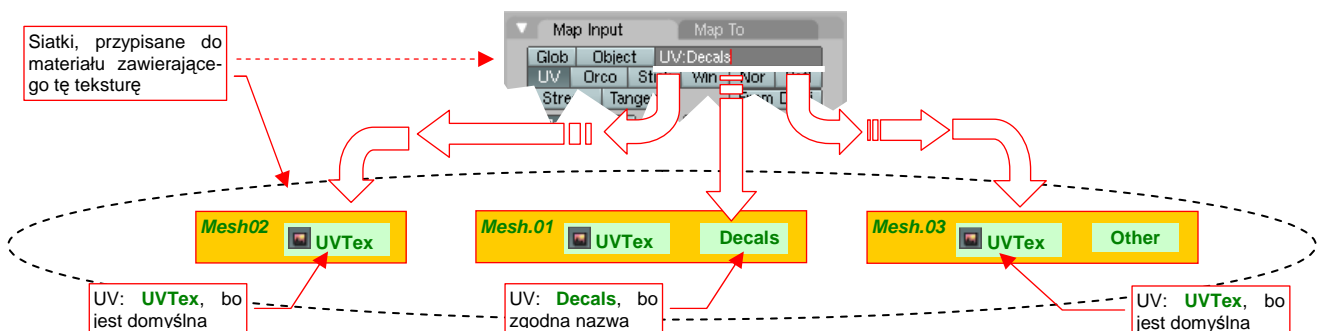
Zwróć także uwagę na przełączniki z lewej strony nazwy warstwy. Pierwszy (☐) służy do wskazania aktywnej warstwy UV — czyli tej, którą aktualnie widzisz w *UV/Image Editor* (a także — współrzędnych UV i obrazu, używanych w oknie *3D View*, w trybie *textured*).

Jak właściwie używać warstw UV? W parametrach mapowania tekstury (*Material buttons:Map Input*) możesz podać jej nazwę (Rysunek 15.34.4):



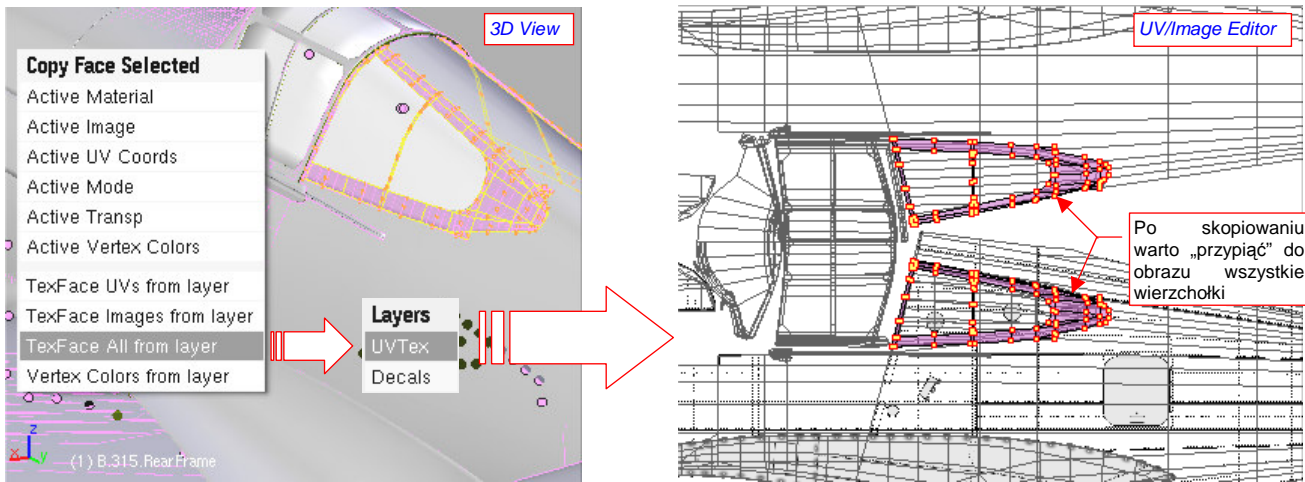
Rysunek 15.34.4 Przypisanie tekstury do konkretnej warstwy UV

Jeżeli nie wpiszesz nic — z każdej siatki zostanie użyta warstwa, która w panelu *Mesh* ma włączony przycisk (por. Rysunek 15.34.3a). Ta domyślna warstwa zostanie także użyta w sytuacji, gdy siatka nie posiada warstwy UV o nazwie podanej w parametrach mapowania (Rysunek 15.34.4). Dzięki mechanizmowi „zastępowania”, możesz umieścić warstwy UV o nazwie **Decals** tylko w kilku siatkach, spośród wszystkich, do których jest przypisany materiał zawierający tę teksturę. Dla pozostałych siatek Blender użyje współrzędnych z warstwy, którą oznaczyłeś jako domyślną (Rysunek 15.34.5):



Rysunek 15.34.5 Zasada wyboru współrzędnych UV do mapowania tekstury

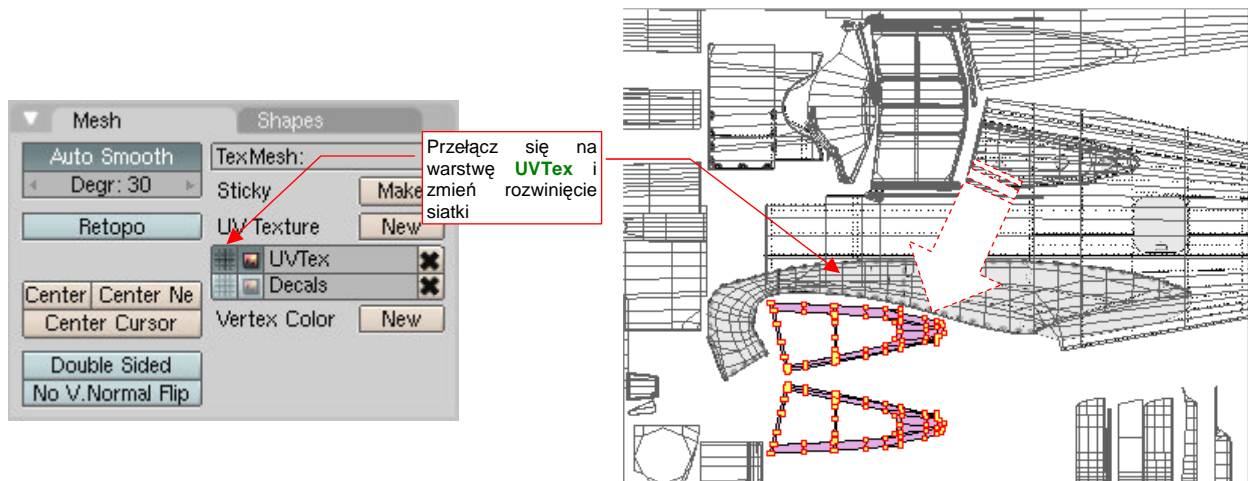
Każda nowa warstwa UV zawiera zupełnie nie zainicjalizowaną siatkę (Rysunek 15.34.3b). Na szczęście ustawienia z innych warstw można bardzo szybko powielić. W oknie **3D View** naciśnij **Ctrl-C**, aby otworzyć menu podręczne **Copy Face Selected** (nie jest dostępne w żaden inny sposób). Z tego menu wybierz polecenie **TexFace All from layer** (Rysunek 15.34.6):




Rysunek 15.34.6 Kopiowanie zawartości warstwy UV

Po wybraniu tego polecenia, pojawi się menu **Layers**, z którego należy wybrać warstwę — wzorec (tę, która już ma wszystkie poustawiane). W przykładzie na ilustracji wskazałem jako wzorec warstwę **UVTex**. W efekcie aktualna warstwa — **Decals** — stała się jej kopią, z jednym istotnym wyjątkiem: nie zostały przeniesione znaczniki „przypięcia” (*Pin*) do obrazu. Zaraz po skopiowaniu warstwy przypnij je wszystkie, aby Ci się nie „odwinęły” w jakąś niespodziewaną stronę, np. w wyniku omyłkowego naciśnięcia przycisku **E** (*Unwrap*).

W rezultacie użycia **Copy Face** → **TexFace All from layer** uzyskaliśmy dwie identyczne warstwy. Którą teraz przekształcić tak, jak pokazuje ją Rysunek 15.34.1b)? Rozwinięte w ten sposób siatki będą użyte przez większą liczbę tekstur, więc zmienmy warstwę domyślną (**UVTex** — Rysunek 15.34.7):

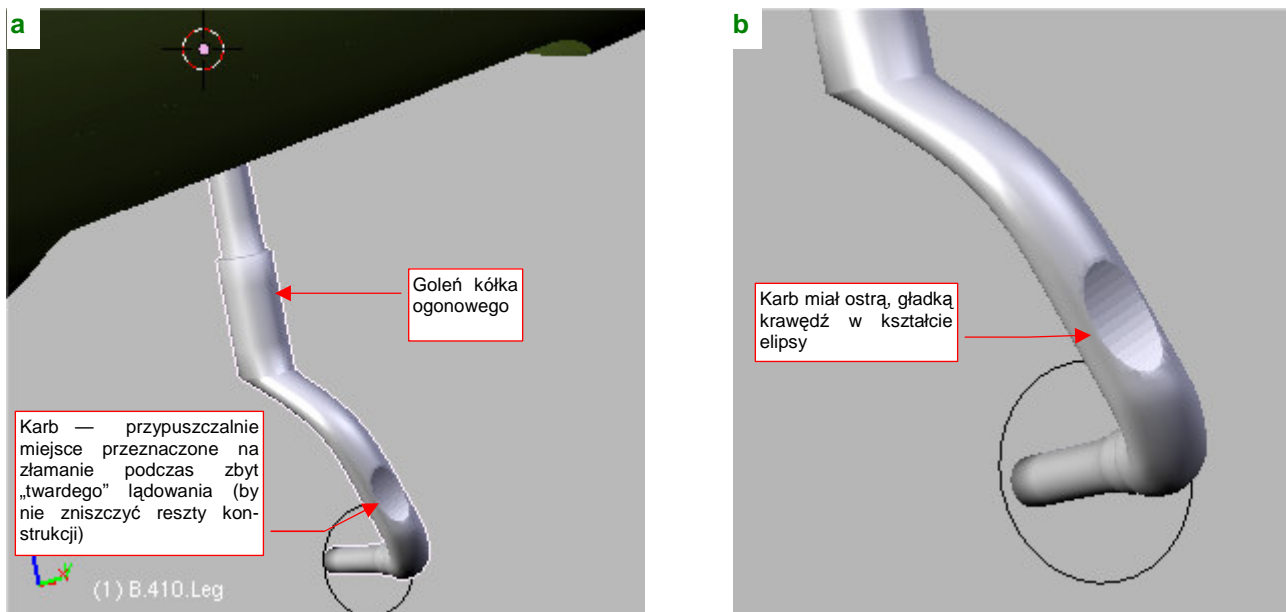


Rysunek 15.34.7 Nanieśienie zmian na jedną z warstw UV

Kliknij w przelącznik , by z powrotem uczynić **UVTex** warstwą aktywną, a potem przesuń rozwinięcie siatki ramek poza obszar kadłuba. W ten sposób uzyskałeś dla tej samej siatki dwa alternatywne rozwinięcia, które będziesz mógł użyć przy mapowaniu tekstur.

15.35 Operacje na bryłach (*Boolean*)

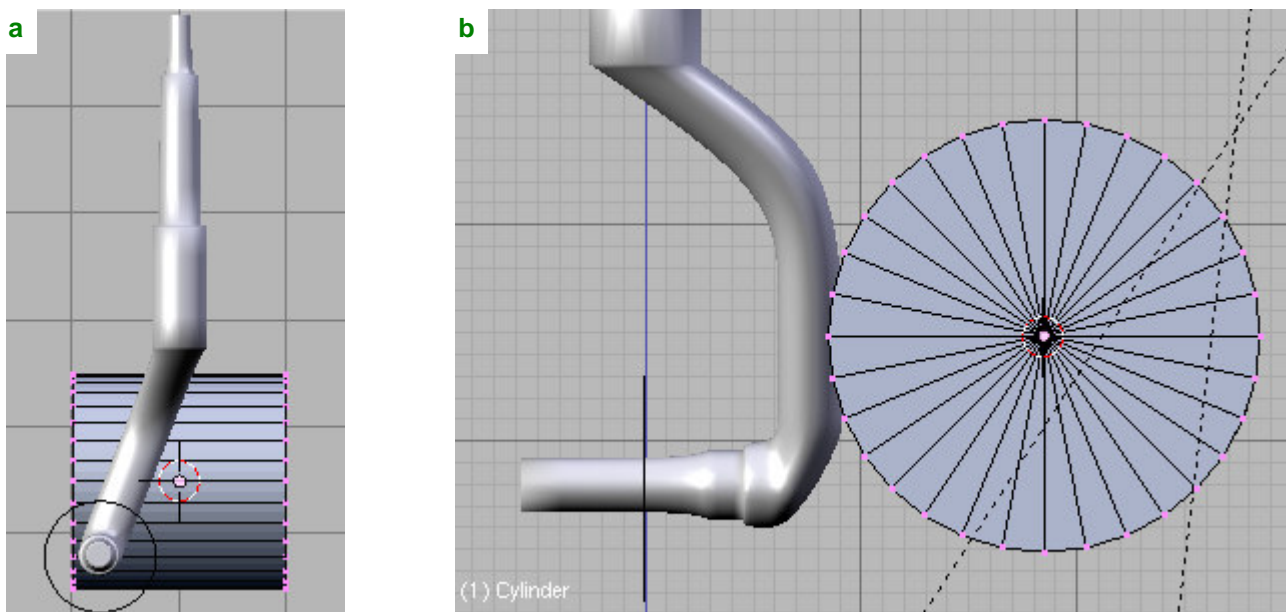
Podczas wykonywania detali, możesz się natknąć na takie drobiazgi, jak nacięcie (karb) na goleni kółka ogonowego (Rysunek 15.35.1a):



Rysunek 15.35.1 „Karb” na goleni kółka ogonowego

Jak szybko uzyskać taki szczegół? Gdybyśmy chcieli wykonać taki karb w ten sam sposób, co otwory w kadłubie — byłoby to bardzo pracochłonne. Trzeba by było z góry odpowiednio przygotować poprzeczny kontur goleni, z „ostrymi” (*crease* =1) krawędziami w pobliżu granicy zagłębienia (por. formowanie kadłuba, str. 169). W efekcie otrzymalibyśmy coś, co i tak nie miałoby takiej ładnej, ostrej krawędzi jak zagłębienie, które pokazuje Rysunek 15.35.1b). W takim przypadku bardziej opłaca się zastosować modyfikator *Boolean*.

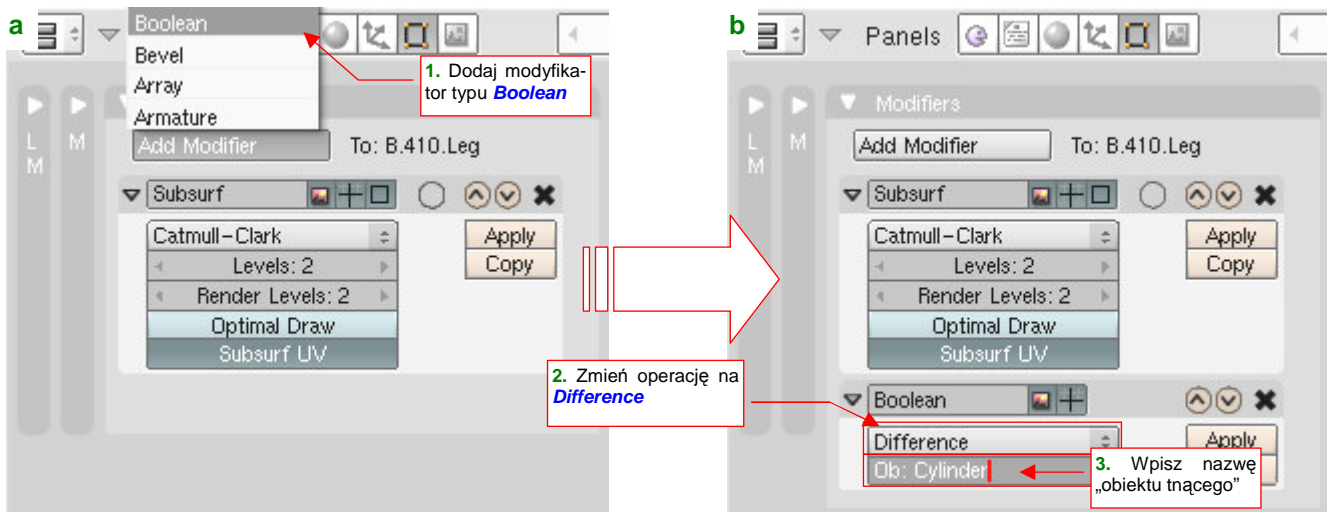
Karb wygląda jak wycięty fragmentem walca, więc wstawmy do rysunku cylinder o dość dużym promieniu (Rysunek 15.35.2a):



Rysunek 15.35.2 „Narzędzie tnące” — walec (*Cylinder*)

Rysunek 15.35.2b) pokazuje, że boczna ściana walca jest nieznacznie zagłębiona w goleni — na tyle, ile ma wynosić głębokość karbu. Na razie pozostawiłem temu obiektowi nazwę *Cylinder*. To nasze „narzędzie tnące”.

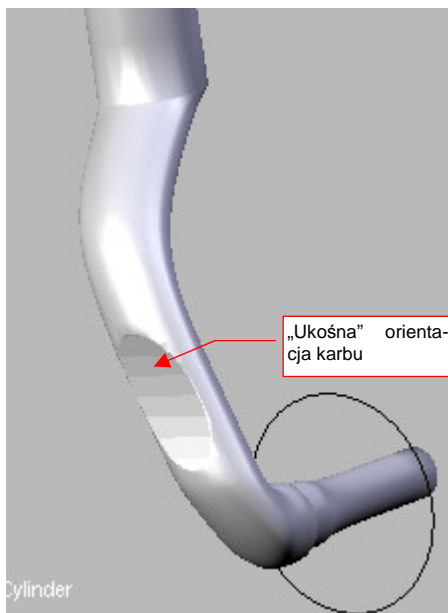
Teraz przełącz się w tryb edycji goleni (**B.410.Leg**), i dodaj nowy modyfikator, typu **Boolean** (Rysunek 15.35.3a):



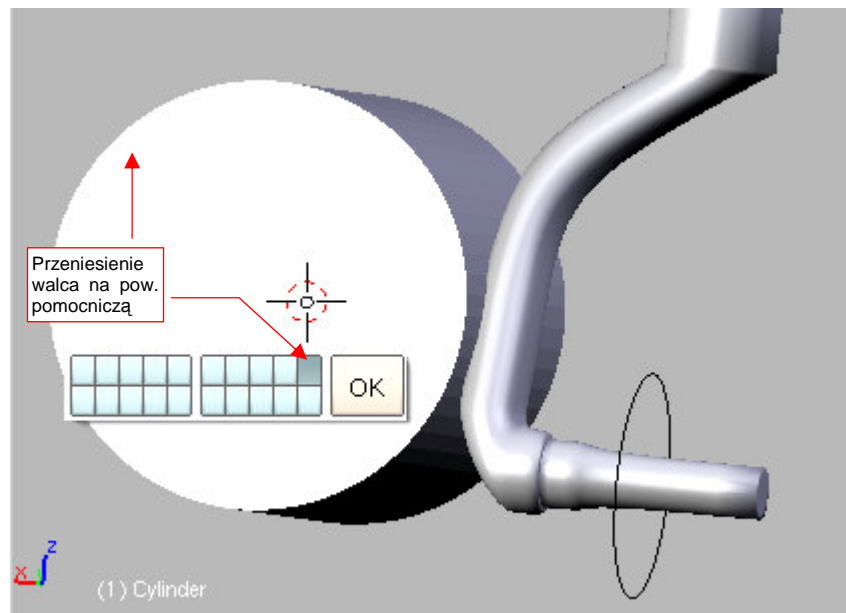
Rysunek 15.35.3 Dodanie do siatki goleni modyfikatora typu **Boolean**

W panelu modyfikatora zmień rodzaj operacji na **Difference**, i wpisz nazwę “narzędzia tnącego” — cylindra, który przed chwilą przygotowaliśmy (Rysunek 15.35.3b).

Rysunek 15.35.4 przedstawia rezultat operacji po ukryciu (**H**) walca:



Rysunek 15.35.4 Rezultat operacji **Boolean**



Rysunek 15.35.5 Przesunięcie walca na niewidoczną (podczas redneringu) warstwę pomocniczą

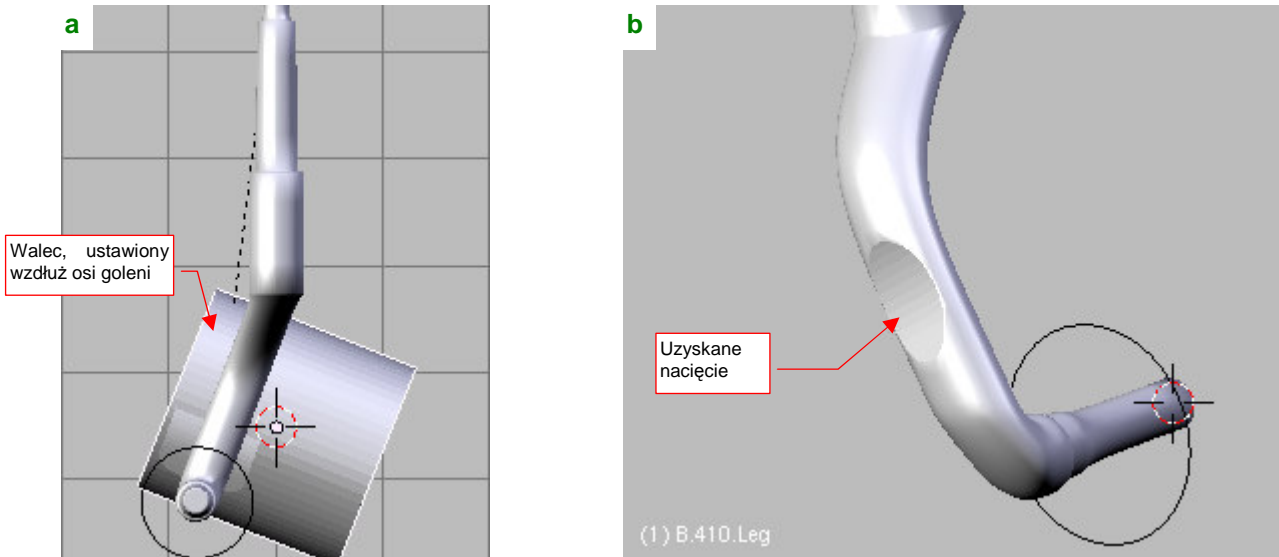
Uzyskany karb ma ładną, ostrą krawędź. Na ostatecznym renderze nie będzie widać lekkich załamania ścianek wewnątrz zagłębienia.

- Operacje typu **Boolean** nie są jeszcze dopracowane w tej wersji Blendera. Czasami w wyniku użycia tego modyfikatora możesz uzyskać dodatkowe ściany, których nie powinno być. Zmień wtedy układ siatki, poprzez dodanie jednej — dwóch nowych krawędzi, i spróbuj ponownie.

Oczywiście, nie chcemy widzieć walca na ostatecznym renderingu. Przesuń go na warstwę 10 (**M**, Rysunek 15.35.5). (Warstwę 10 przeznaczaliśmy na różne pomocnicze obiekty — por. str. 261).

Zaletą operacji *Boolean*, implementowanych jako modyfikator, jest dynamiczne połączenie obydwu brył — „narzędzia” i „obiektu”. Dzięki temu nie komplikujesz oryginalnych siatek i nadal możesz łatwo zmieniać ich kształt podczas edycji. W przypadku wygładzonych powierzchni, krawędź przecięcia jest wyznaczana na ścianach powierzchni podziałowych. Dzięki temu nie zaburzają kształtu reszty powłoki.

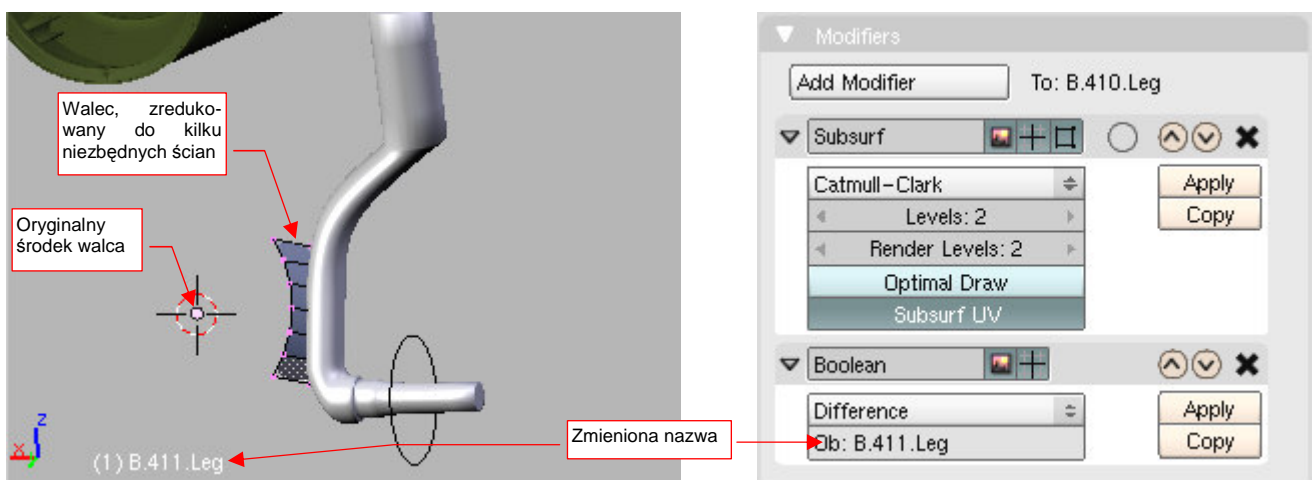
Na razie poprawmy trochę krawędzie karbu: powinny tworzyć elipsę równoległą do osi goleni, więc obróćmy odpowiednio nasz cylinder (Rysunek 15.35.6a):



Rysunek 15.35.6 Dodanie do siatki goleni modyfikatora typu *Boolean*

W rezultacie uzyskaliśmy pożądany kształt (Rysunek 15.35.6b). Zwróć jednak uwagę, że teraz walec obracał się dość „opornie”. Wadą zastosowania modyfikatora *Boolean* jest wolniejsze odświeżanie ekranu podczas przesuwania, obrotu lub innych zmian każdego „składnika” tej operacji. Przy każdym Twoim ruchu wykonywane są ponownie obliczenia przenikania obydwu brył. Na szczęście na finalny czas renderingu to nie ma wpływu. Być może zdecydujesz się wyłączać podczas co intensywniejszych edycji działanie tego modyfikatora. (Wystarczy usunąć z jego parametrów nazwę obiektu „tnącego”, a po zakończeniu zmian — wpisać ponownie).

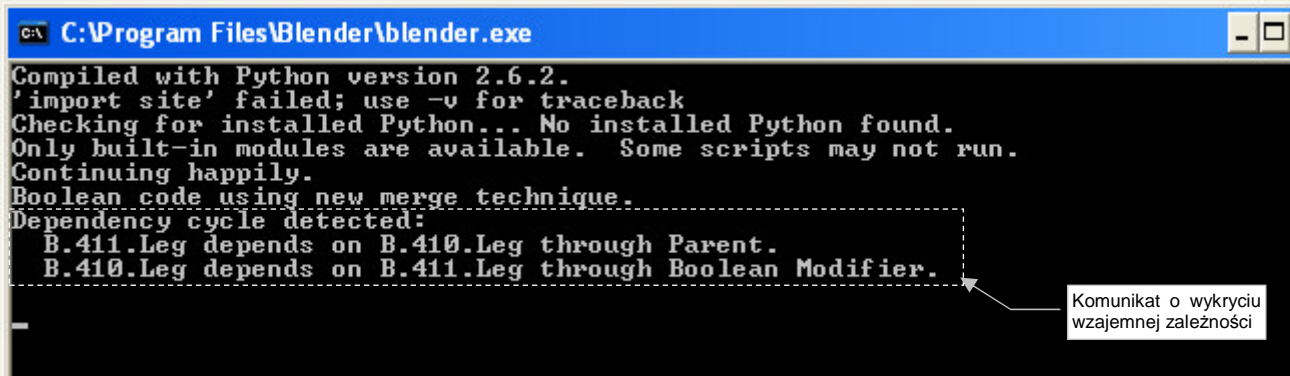
Do uzyskania poprawnego „wycięcia” nie jest właściwie potrzebny cały walec — wystarczy fragment jego ściany (Rysunek 15.35.7):



Rysunek 15.35.7 Walec „tnący”, zredukowany do niezbędnych ścian Rysunek 15.35.8 Uaktualnione parametry modyfikatora

Nadaj temu, co zostało z oryginalnego cylindra jakąś nazwą porządkową (w tym przykładzie — **B.411.Leg**). Zwróć uwagę, że odpowiedniej zmianie uległ także wpis w panelu modyfikatora (Rysunek 15.35.8). Aby karb nie „uciekał” podczas ruchu goleni (będziemy ją animować), przypisz ten obiekt do goleni (*Make Parent* — **Ctrl-P**).

W wyniku dołączenia obiektu **B.411.Leg** do goleni (**B.410.Leg**), w konsoli Blendera zacznie się pojawiać komunikat ostrzegawczy (Rysunek 15.35.9):



```

C:\Program Files\Blender\blender.exe
Compiled with Python version 2.6.2.
'import site' failed; use -v for traceback
Checking for installed Python... No installed Python found.
Only built-in modules are available. Some scripts may not run.
Continuing happily.
Boolean code using new merge technique.
Dependency cycle detected:
  B.411.Leg depends on B.410.Leg through Parent.
  B.410.Leg depends on B.411.Leg through Boolean Modifier.

```

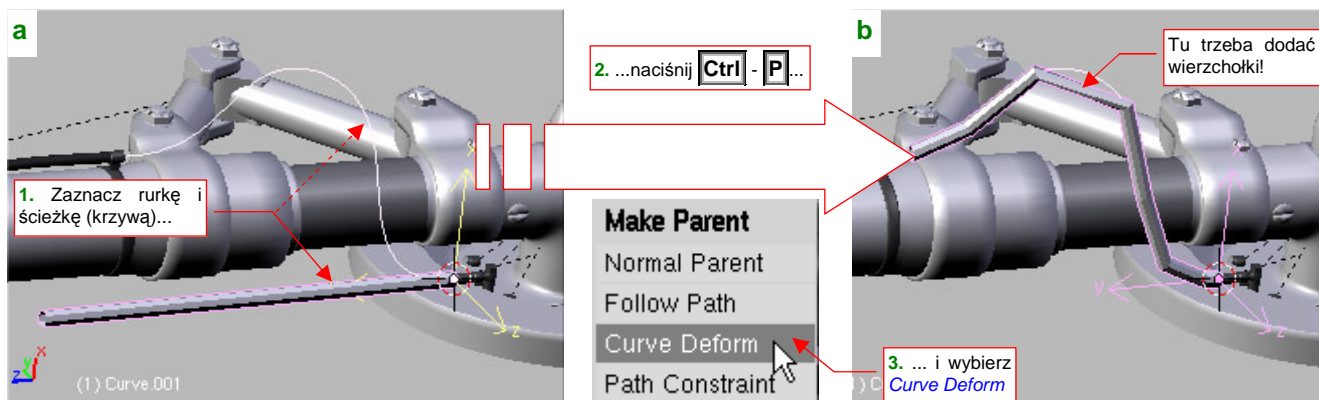
Rysunek 15.35.9 Komunikaty o wzajemnym powiązaniu goleni (**B.410.Leg**) i ściany „tnącej” (**B.411.Leg**)

To trochę „alergiczna” reakcja: **B.411.Leg** wpływa tylko na kształt siatki goleni, w żaden sposób nie zmienia położenia i orientacji obiektu **B.410.Leg**, który jest jego „rodzicem”. Nie ma się czym przejmować!

Oczywiście, gdy będziesz pewien że tego elementu nie będziesz już zmieniał, możesz „utrwalić” obydwie modyfikatory (*Subsurf* i *Boolean*), przypisane do siatki goleni (**B.410.Leg**), i wyrzucić z rysunku niepotrzebny już pomocniczy **B.411.Leg**. Zrób to jednak na sam koniec pracy, bo to jest w sumie krok nieodwracalny. Po „utrwaleniu” tych modyfikatorów, w przypadku konieczności poważniejszej zmiany kształtu, najprościej będzie stworzyć taką goleń od nowa.

15.36 Wyginanie wzdłuż krzywej (Curve)

Zazwyczaj wzdłuż krzywej są trzeba wygiąć jakieś cylindry — aby w ten sposób stworzyć imitację jakiegoś przewodu, kabla, lub rury. W najprostszej wersji polega to na wskazaniu cylindra, krzywej (*Path* — por. str. 842), i naciśnięciu **Ctrl**-**P** (Rysunek 15.36.1a):

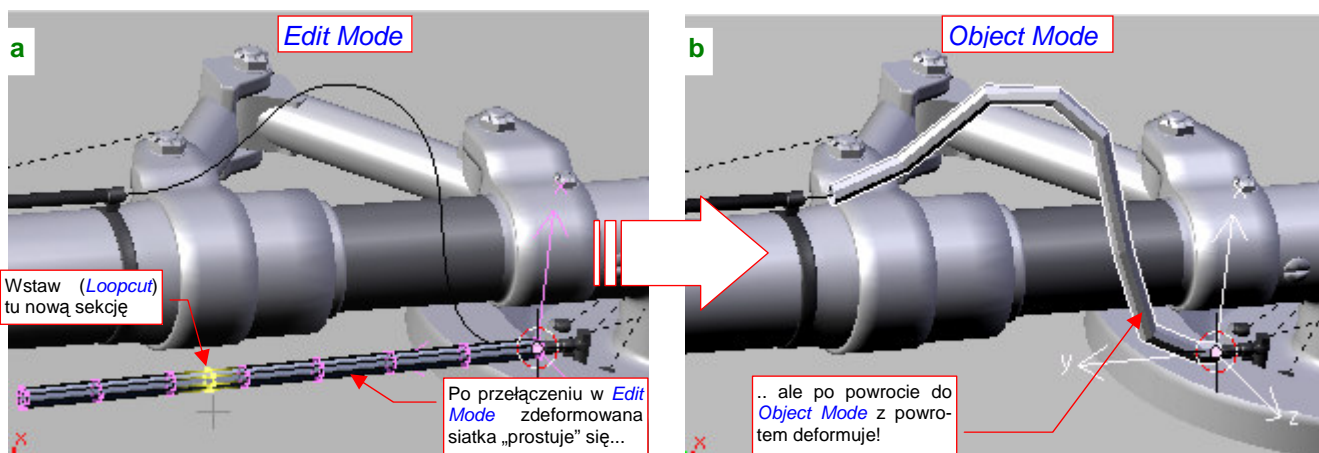


Rysunek 15.36.1 Wygięcie rury (Cylinder) wzdłuż krzywej (Path)

Z menu *Make Parent*, które się pojawi, wybierz opcję *Curve Deform* — i już (Rysunek 15.36.1b)!

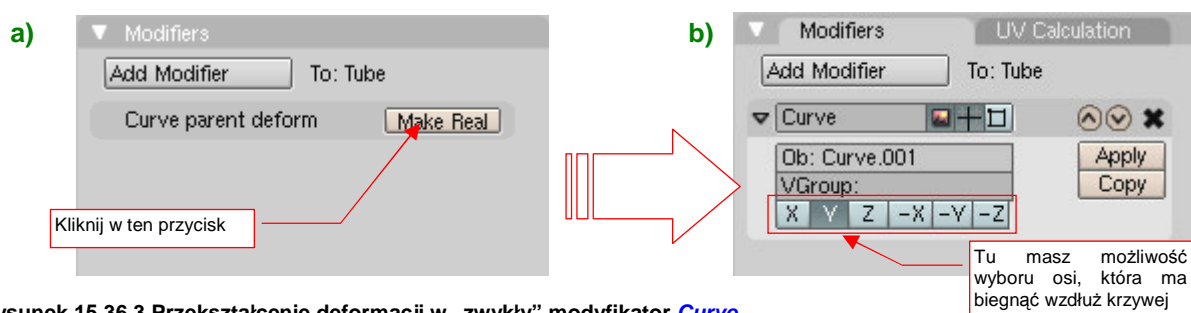
- Przed operacją warto dopilnować, by środki obydwu obiektów — cylindra i krzywej — leżały w tym samym miejscu, oraz by osie Y ich lokalnych układów współrzędnych pokrywały się ze sobą.

Powłóce, którą pokazuje Rysunek 15.36.1b), wyraźnie przydałby się dodatkowy segment. Gdy przejdziesz w tryb edycji wygiętej siatki — ta natychmiast się wyprostuje (Rysunek 15.36.2a):



Rysunek 15.36.2 Edycja siatki wygiętej rury

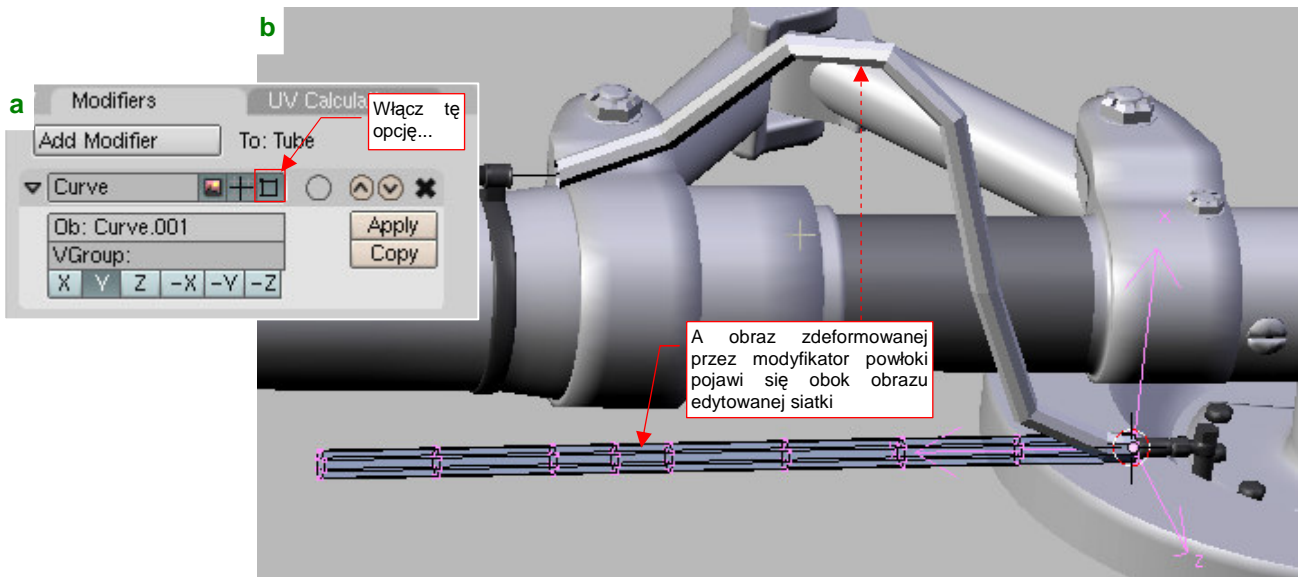
To może utrudnić dostosowanie gęstości siatki do promienia krzywizny. Tym niemniej gdy przełączysz się z powrotem w *Object Mode* — rura znów stanie się wygięta (Rysunek 15.36.2b). Gdy zajrzysz do panelu *Modifiers*, zobaczysz tam dziwny zapis (Rysunek 15.36.3a):



Rysunek 15.36.3 Przekształcenie deformacji w „zwykły” modyfikator Curve

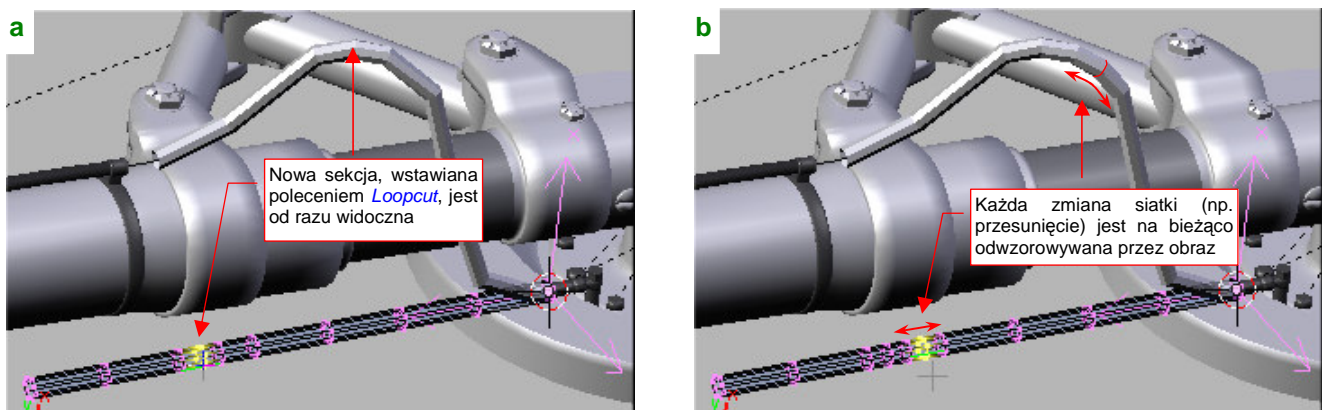
Naciśnięcie *Make Real* przekształci go w „zwykły” modyfikator typu *Curve* (Rysunek 15.36.3b).

W panelu *Modifiers* możesz skorzystać z przełącznika „*Enable modifier during Editmode*”, umieszczonego w „nagłówku” każdego modyfikatora (Rysunek 15.36.4a). Gdy go włączysz, obraz wygiętej siatki pojawi się także podczas edycji (Rysunek 15.36.4b):



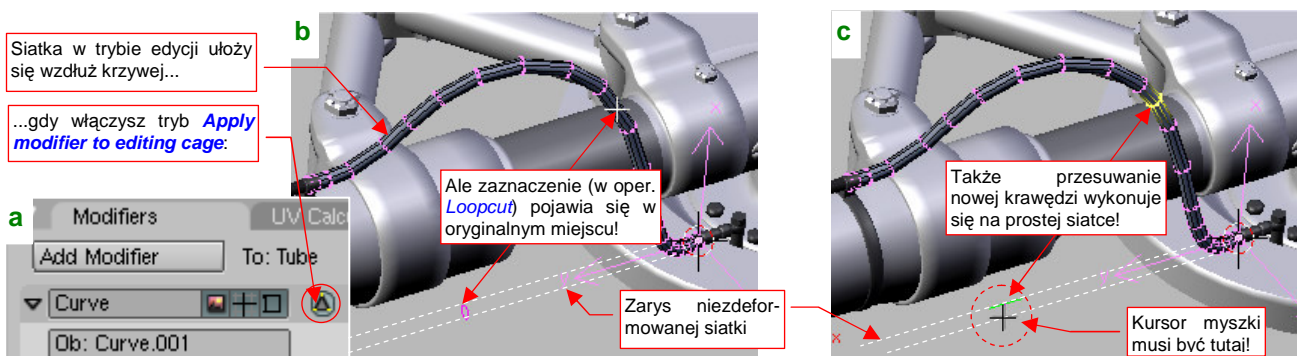
Rysunek 15.36.4 Włączenie podglądu kształtu podczas edycji siatki

Gdy teraz wstawisz nową sekcję w siatkę (np. poleceniem *Loopcut*), natychmiast zobaczysz ją także na obrazie zdeformowanej powłoki (Rysunek 15.36.5a):



Rysunek 15.36.5 Edycja zdeformowanej siatki z bieżącym podglądem zmian

W ten sposób będziesz widział zmiany ostatecznego kształtu „na bieżąco”, w trakcie przesuwania wierzchołków (Rysunek 15.36.5b). Można także jeszcze włączyć tryb *Apply to editing cage* (Rysunek 15.36.6):

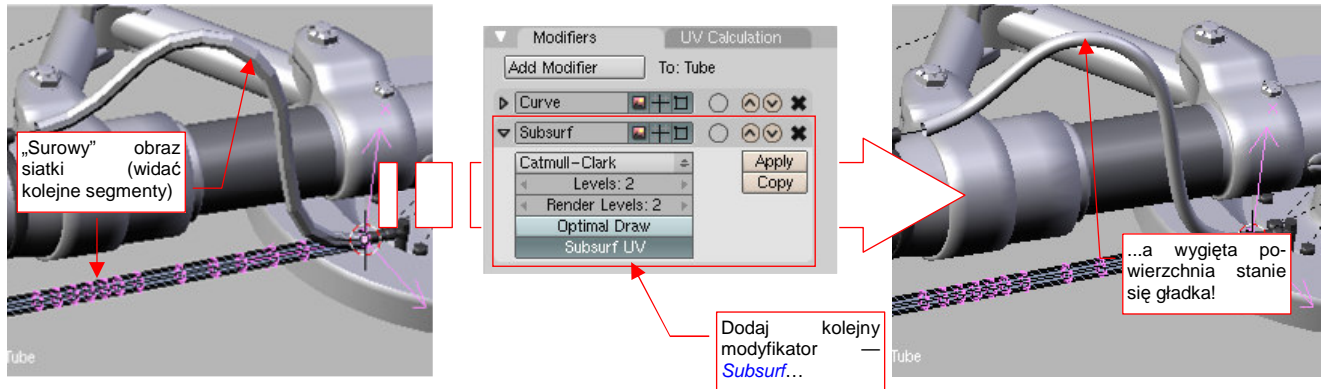


Rysunek 15.36.6 Wynik włączenia trybu *Apply modifier to editing cage* i „efekty uboczne”

Niestety, edycja zakrzywionej siatki nie jest w tym trybie do końca dopracowana. Rysunek 15.36.6b) i c) pokazują kolejne fazy operacji *Loopcut* (**Ctrl**-**R**), por. str. 875). Punkty odniesienia nadal leżą nadal na prostej siatce!

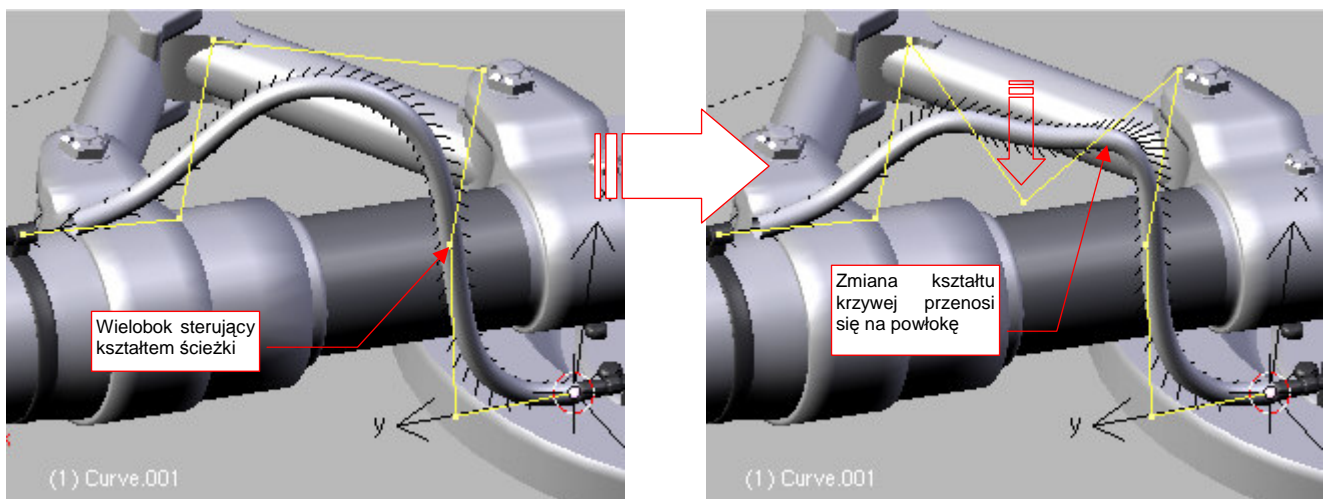
- W edycji z włączonym trybem *Apply modifier to editing cage* elementy interaktywne (zaznaczenia krawędzi, tory przesunięć *Edge Slide*) są rysowane zawsze tak, jak gdyby siatka nie była zdeformowana. To jakiś błąd Blendera. Gdy zostanie usunięty, praca w tym trybie będzie na pewno dużym ułatwieniem.

Po odpowiednim zagęszczeniu siatki „na zakrętach”, można dodać kolejny modyfikator — *Subsurf* — by ją dodatkowo wygładzić (Rysunek 15.36.7):



Rysunek 15.36.7 Dodanie kolejnego modyfikatora — wygładzania

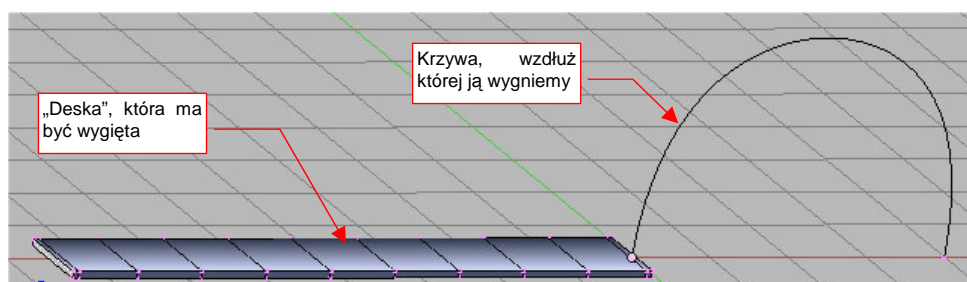
Modyfikator *Curve* można traktować jako specyficzną formę relacji hierarchicznej (*Parent — Child*). Jakkolwiek nie obrócimy lub przesuniemy krzywą, wygięta wzdłuż jej linii rura podąży za nią. Co więcej, zmiana kształtu krzywej zmienia odpowiednio kształt wygiętej powłoki (Rysunek 15.36.8):



Rysunek 15.36.8 Kształt krzywej steruje kształtem rury

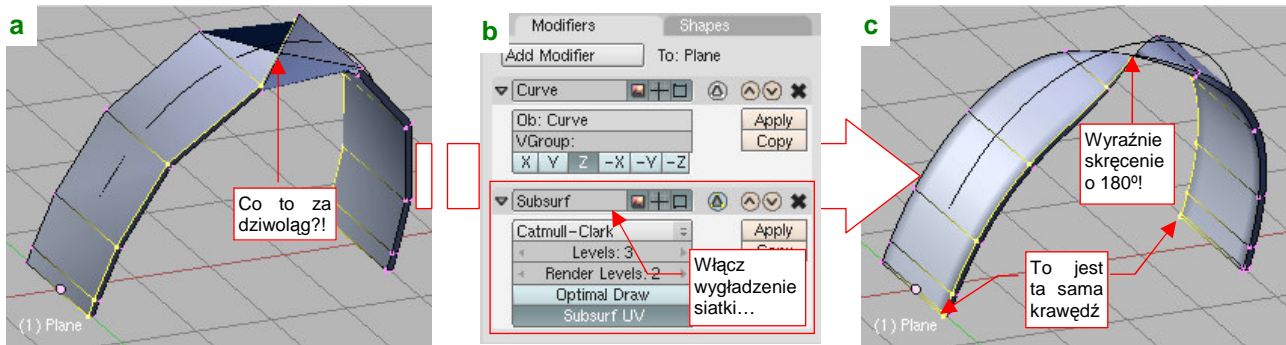
Kształt krzywej zmieniamy, przesuwając (w trybie edycji) jej punkty sterujące (por. str. 843). Gdy zaczniesz to robić, zauważysz że wygięty cylinder „podąży” za deformowaną linią. Takie dynamiczne zmiany, zachodzące w trakcie edycji, bardzo ułatwiają dopasowania zespołu do reszty modelu.

Gdy wyginamy taką „deskę” jaką pokazuje Rysunek 15.36.9, ważny staje się kierunek skręcenia poprzecznego:



Rysunek 15.36.9 Inny przypadek: płaska „deska” i jej krzywa

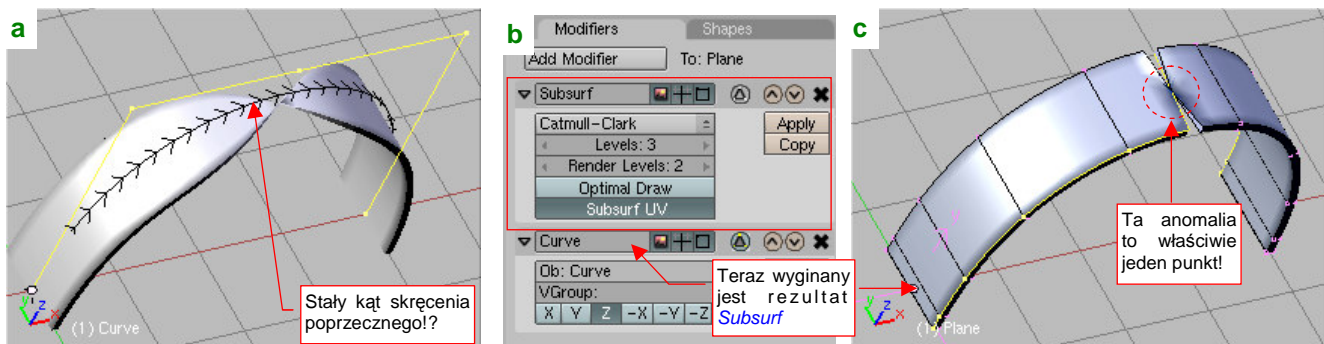
Po wygięciu siatki na środku krzywej może się pojawić nieoczekiwane skręcenie (Rysunek 15.36.10a):



Rysunek 15.36.10 Zaskakujące skręcenie na gładkiej krzywej

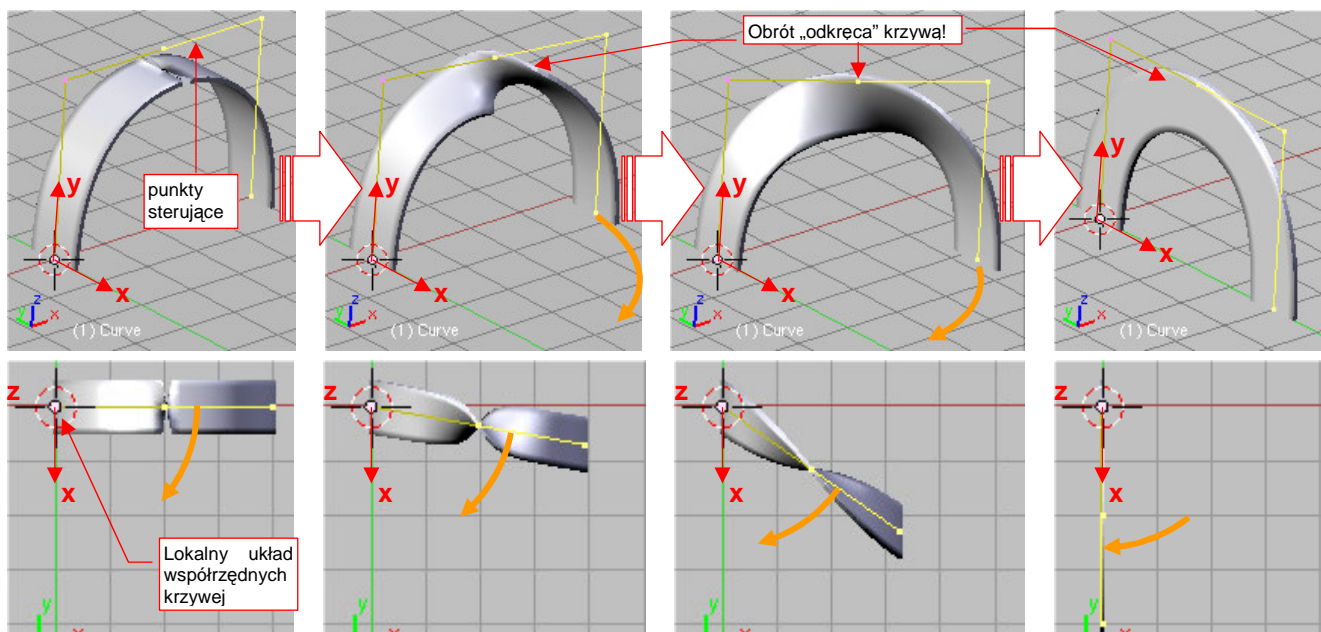
Aby się upewnić co do istoty tego dziwoląga, wygładzam siatkę modyfikatorem *Subsurf* (Rysunek 15.36.10b). Rysunek 15.36.10c) pokazuje dobitnie, że krzywa skręciła siatkę w środku o 180°.

Włączam więc szybko edycję punktów sterujących krzywej (Rysunek 15.36.11a). Strzałki, które powinny wskazywać na kierunek poprzeczny do krzywej, nie wykazują żadnej anomalii:



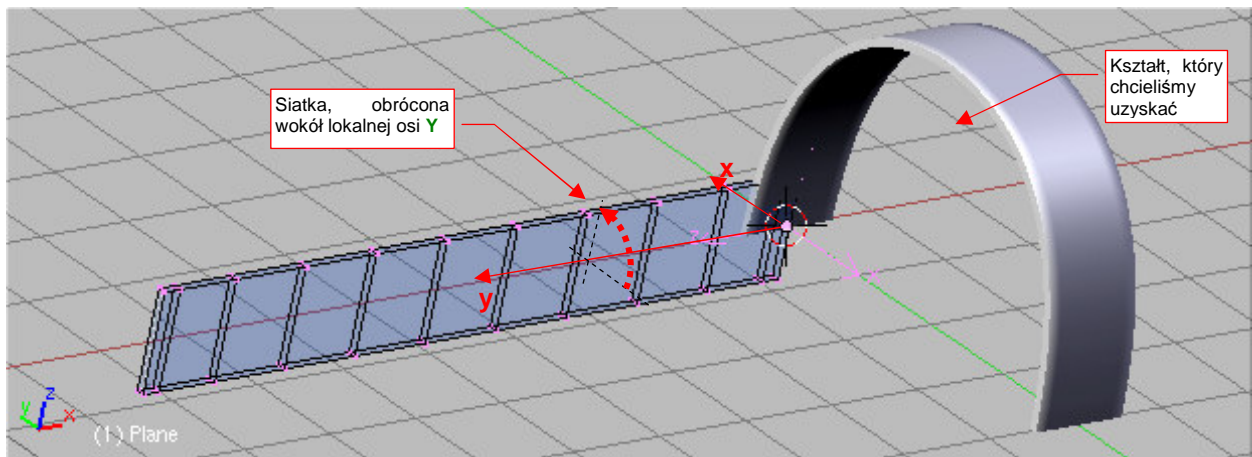
Rysunek 15.36.11 Weryfikacja kształtu krzywej

W takim razie zwiększę gęstość siatki wyginanej wzdłuż krzywej (wy starczy przesunąć modyfikator *Subsurf* przed *Curve* — Rysunek 15.36.11b). Rysunek 15.36.11c) pokazuje teraz wyraźnie, że mamy do czynienia z jednym punktem na krzywej, w którym kierunek poprzeczny „zwarował”. Co ciekawe, wystarczy obrócić punkty kontrolne krzywej o 90°, z lokalnej płaszczyzny *YZ* na *XY*, by wszystko wróciło do normy (Rysunek 15.36.12):



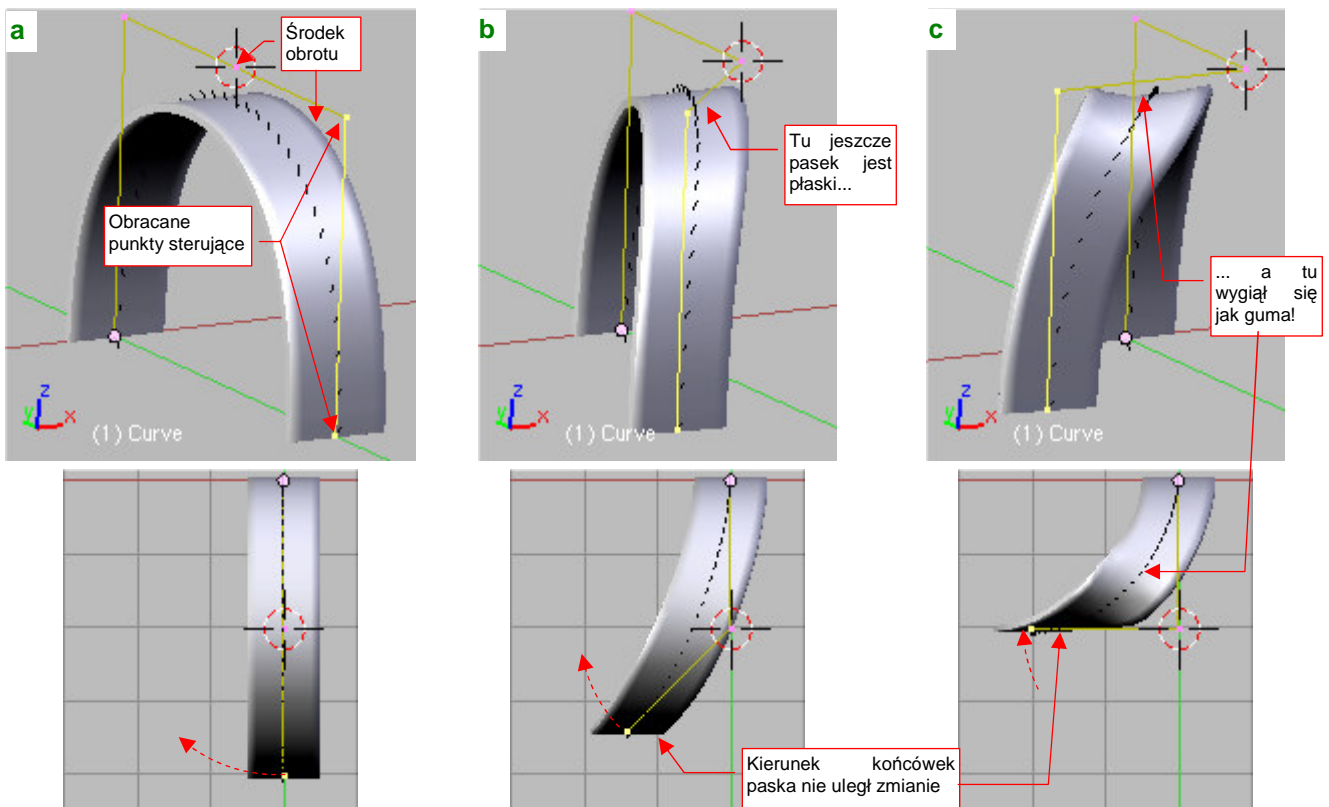
Rysunek 15.36.12 Rezultat obrotu punktów sterujących krzywą

Wygląda więc na to, że wystarczy teraz obrócić deformowaną siatkę wokół lokalnej osi Z, by uzyskać efekt, o który chodziło nam na początku (Rysunek 15.36.13):



Rysunek 15.36.13 Dostosowanie orientacji siatki do poprawionej orientacji krzywej

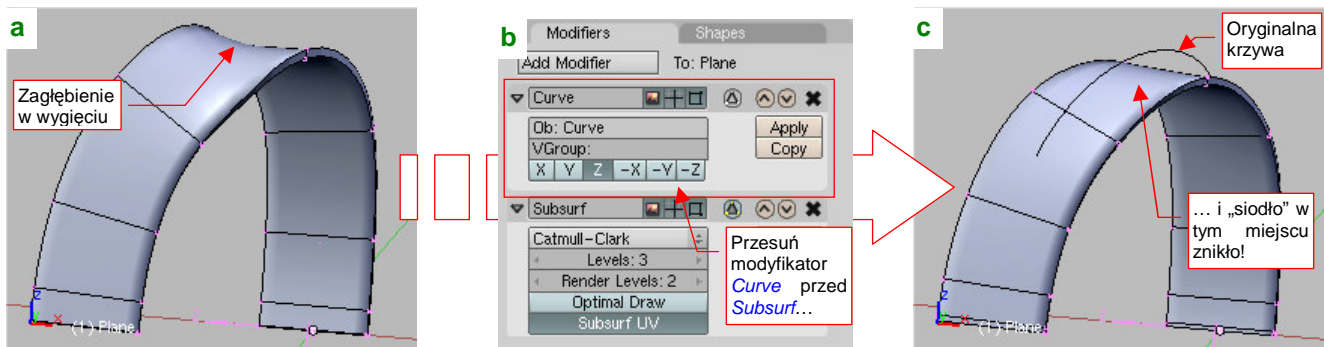
Na razie to były doświadczenia z krzywą płaską, a zobaczymy, jak skręci nasz pasek wygięcie wokół krzywej przestrzennej. Najprostszą operacją, która przekształci w ten sposób naszą linię, jest obrót wokół punktu środkowego dwóch ostatnich wierzchołków wieloboku sterującego (Rysunek 15.36.14a):



Rysunek 15.36.14 Deformacja paska na krzywej przestrzennej

Rysunek 15.36.14a) to siatka wygięta wokół płaskiej krzywej. Zacząłem obracać dwa ostatnie punkty sterujące wokół osi pionowej. Rysunek 15.36.14b) przedstawia kształt krzywej po obrocie końcówki o 45° . Wygląda na to, że żadna część wyginanego paska nie została skręcona wzdłuż osi podłużnej. Zwróć uwagę na koniec paska: przez cały czas nie zmienił swojej orientacji, i jest równoległy do początku. Wreszcie Rysunek 15.36.14c) pokazuje już bardzo zdeformowaną linię — wielobok kontrolny jest zgięty pod kątem 90° . O ile koniec paska nadal jest równoległy do początku, to w jego środku pojawiło się wygięcie w kształcie siodła.

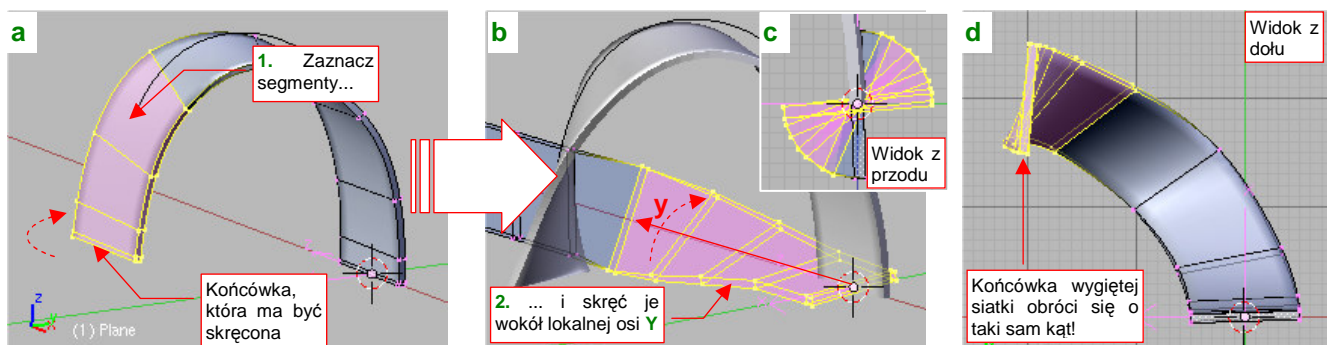
W większych wygięciach siatka ma tendencję do formowania charakterystycznego zagłębienia, w kształcie „siodła”. Przypomina przez to pasek z gumy (Rysunek 15.36.15a):



Rysunek 15.36.15 Prostowanie „gumowych zagłębień”

Jeżeli chcesz uzyskać efekt bardziej zbliżony do paska z papieru lub tkaniny, powinieneś zmienić miejscami modyfikatory siatki. Modyfikator *Subsurf* powinien się znaleźć po modyfikatorze *Curve* (Rysunek 15.36.15b). W efekcie zwiększą się promienie krzywizn, i zmniejszy się, lub wręcz zniknie efekt „siodła” (Rysunek 15.36.15c). Ceną za ten efekt jest odstawanie siatki od krzywej — ale w wielu przypadkach (np. przy modelowaniu pasów uprząży fotela pilota) nie jest to takie ważne.

Blender automatycznie wyznacza poprzeczne skręcenie siatki na podstawie lokalnego promienia krzywizny ścieżki *Path*. Dlatego może się zdarzyć, że koniec lub jakiś inny fragment siatki nie jest skręcony tak, jakbyśmy sobie tego życzyli. Na szczęście można to łatwo skorygować — wystarczy „ręcznie” obrócić wybrane segmenty siatki (Rysunek 15.36.16):



Rysunek 15.36.16 Zmiana kąta skręcenia fragmentu wygiętej siatki

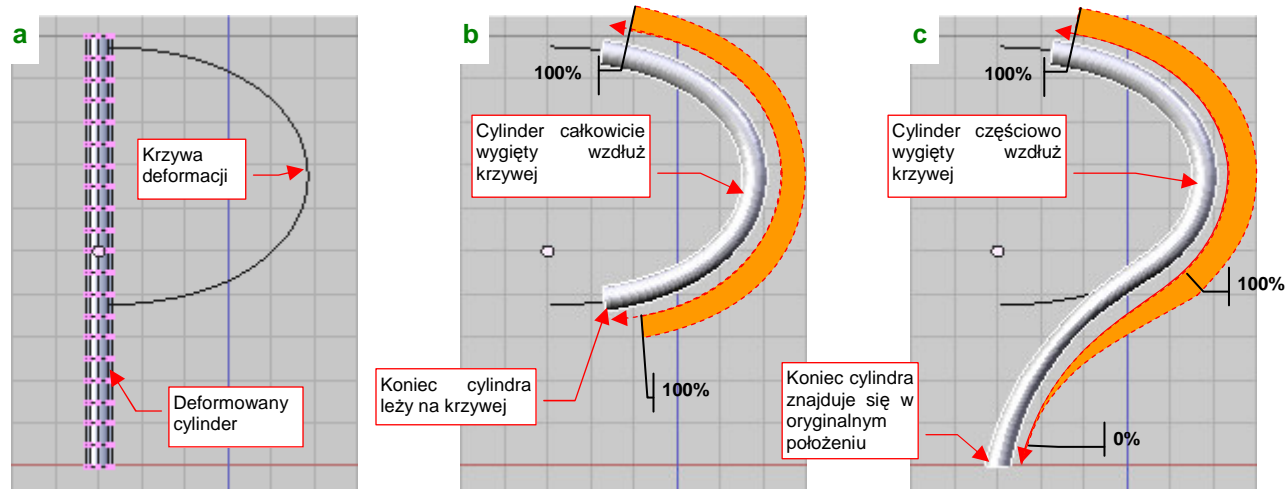
Załóżmy, że zależy nam na skręceniu końca pasa, który pokazuje Rysunek 15.36.16a), o 90° . Zaczynij od zaznaczenia (przy włączonym trybie *Apply modifier to editing cage* — por. Rysunek 15.36.6, str. 914) odpowiednich sekcji siatki. (Chodzi o to, byś dokładnie wiedział, co trzeba obrócić). Następnie umieść kursor 3D w środku siatki (**Shift-S**, *Cursor* → *Selection*), wyłącz tryb *Apply modifier to editing cage*, i po kolei poobracaj poszczególne sekcje wokół lokalnej osi **Y** (Rysunek 15.36.16b, c). Gdy będziesz to robił, koniec wygiętej siatki będzie się obracał o te same kąty (Rysunek 15.36.16d).

- Uwaga: po naciśnięciu w panelu modyfikatora *Curve Deform* przycisku *Make Real* (por. str. 913, Rysunek 15.36.3) koniecznie zmień rodzica (*Parent*) obiektu! Jeżeli tego nie zrobisz, w wyniku jakiegoś błędu Blendera krzywa zaczyna „tańczyć” w kolejnych klatkach animacji (2, 3, 4, 5, 6, ...).

Może Cię to zaskoczyć podczas komponowania sceny w locie: tam, aby uzyskać rozmycie śmigła, musisz ustawić kilkanaście klatek animacji (por. str. 511). Jeżeli czytasz to ostrzeżenie, gdy już masz takie objawy, to skorzystaj ze skryptu do wykrywania przypisanych w ten sposób obiektów: [source/python/ParentCurves.py](#). Skrypt podświetla pierwszy znaleziony obiekt, oraz rodzica jego krzywej („dziadek”?). W większości przypadków wystarczy nacisnąć po prostu **Ctrl-P**, by „dziadek” stał się „rodzicem”. To rozwiązuje problem.

15.37 Malowanie wag (*Weight paint*)

W poprzedniej sekcji (str. 913) pokazałem, jak wygiąć siatkę cylindra wzdłuż linii krzywej. Przekształcaliśmy prostą „rurę” (Rysunek 15.37.1a) w zakrzywioną powłokę, jaką pokazuje Rysunek 15.37.1b):

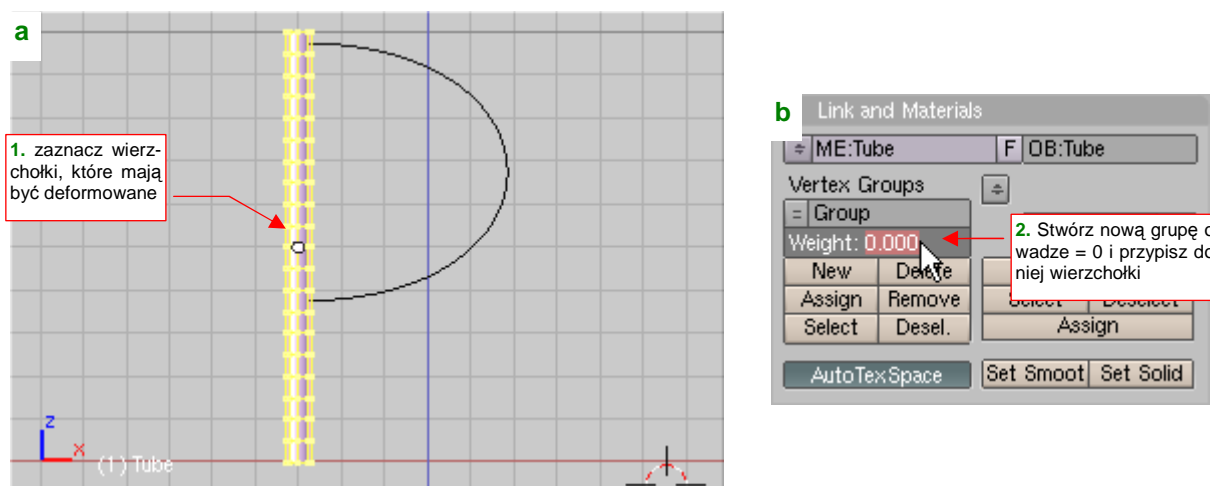


Rysunek 15.37.1 Cylinder wygięty wzdłuż krzywej: całkowicie (b) i częściowo (c)

Czasami potrzebna jest jednak taka deformacja, gdzie tylko część siatki ma się „ułożyć” wzdłuż krzywej, a reszta ma płynnie przechodzić w fragment bez zniekształceń (Rysunek 15.37.1c). Jak taki efekt uzyskać?

W Blenderze służą do tego wagi (*weight*). Waga to przypisana do każdego wierzchołka siatki liczba z zakresu od 1.0 do 0.0. Określa wpływ modyfikatorów (takich, jak *Curve*) na jego położenie. Na przykład wszystkie wierzchołki powłoki, którą pokazuje Rysunek 15.37.1b), mają *Weight* = 1.0 (100% — pełna deformacja). Za to wierzchołki końcówki cylindra, który pokazuje Rysunek 15.37.1c), mają *Weight* = 0.0 (0% — brak deformacji). Punkty leżące na „odchodzącym” od krzywej odcinku cylindra mają odpowiednie wartości pośrednie.

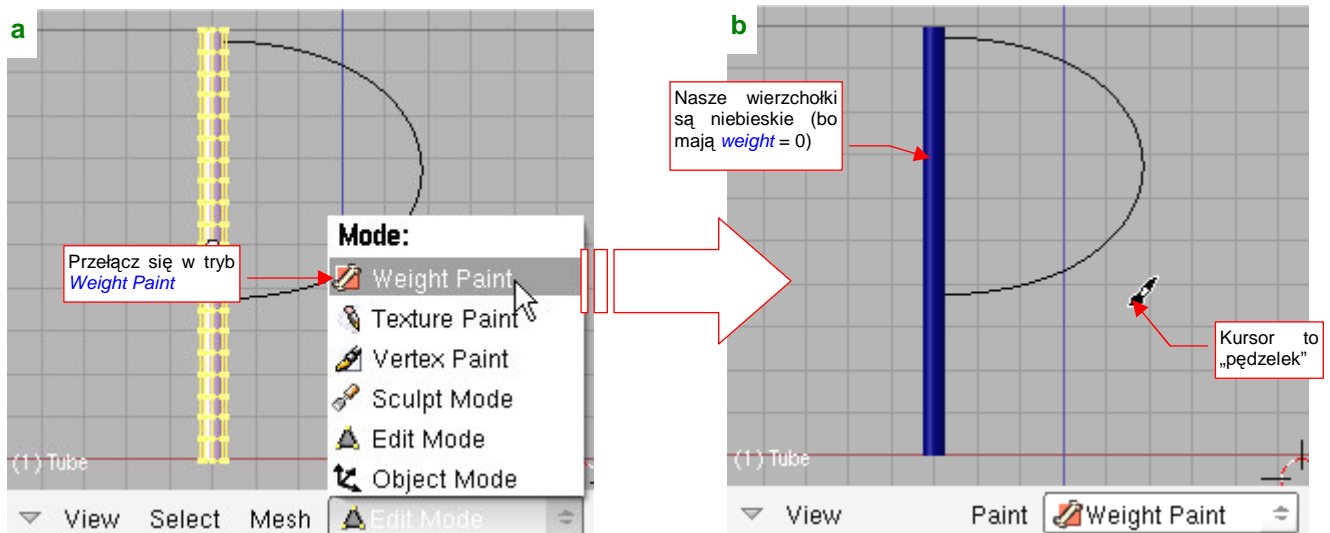
Jak nanosi się wagi na siatkę? Najpierw musisz zaznaczyć obszar, którego w ogóle ma dotyczyć deformacja (Rysunek 15.37.2a):



Rysunek 15.37.2 Zdefiniowanie grupy wierzchołków o początkowej wadze = 0

Następnie stwórz nową grupę wierzchołków (*Vertex Group*, w panelu *Link and Materials* — por. str. 903). W polu *Weight* wpisz początkową wartość wagi (Rysunek 15.37.2b). Na koniec naciśnij przycisk *Assign*, by przypisać zaznaczone na siatce wierzchołki do tej grupy. Na potrzeby tego przykładu nie będę jej zmieniał nazwy — będzie się nazywać *Group*. (W modelu zmieniaj je raczej na coś znaczącego. Nazwy grup wierzchołków muszą być unikalne tylko w obrębie tej samej siatki. Mogą się powtarzać w różnych powłokach, więc zazwyczaj nie ma problemu z ich doborem).

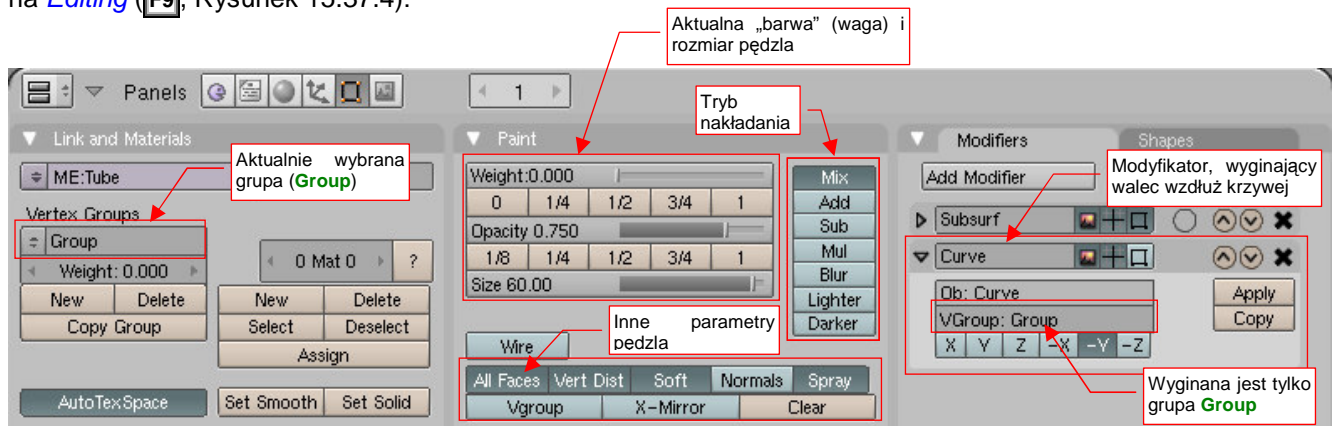
W Blenderze wagi wierzchołków się „maluje”. Zmień tryb pracy z *Edit Mode* na *Weight Paint* (Rysunek 15.37.3):



Rysunek 15.37.3 Przelączenie w tryb *Weight Paint*

Spowoduje to, rzecz jasna, zmianę menu w nagłówku (a właściwie stopce) okna — ze starych pozostanie na nim tylko menu *View*, i pojawi się nowe: *Paint*. Jednocześnie kursor myszki zmieni się w „pędzelek”, a nasza grupa wierzchołków stanie się niebieska (jest to kolor odpowiadający *Weight* = 0.0, którą jej przypisaliśmy).

Nim zaczniemy malować wagi, należy odpowiednio przygotować „środowisko pracy”. Przeląc okno przycisków na *Editing* (**F9**, Rysunek 15.37.4):



Rysunek 15.37.4 Przygotowanie „środowiska pracy” przed malowaniem wag na siatce cylindra

W panelu *Modifiers* dodaj do siatki (o ile nie zrobiłeś tego wcześniej) modyfikator *Curve*. W polu *VGroup* wpisz nazwę odpowiedniej grupy wierzchołków (w naszym przypadku to **Group**).

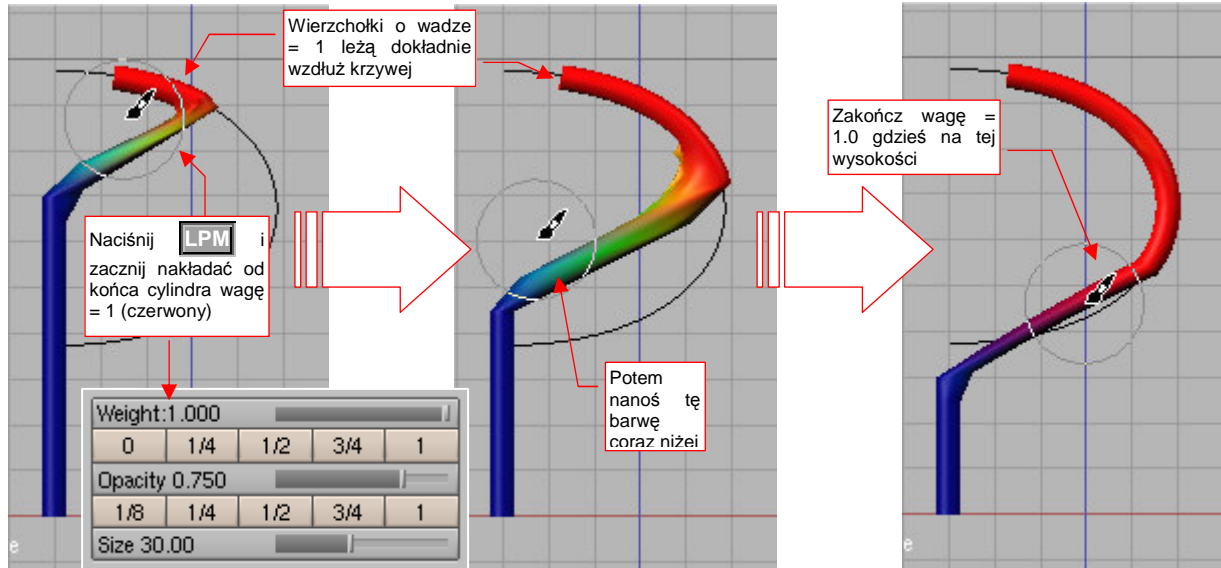
W panelu *Link and Materials*: upewnij się, że aktualnie wybrana grupa jest grupą, której wagi chcesz zmieniać.

- Każda siatka może mieć wiele alternatywnych grup wierzchołków (a każdy wierzchołek może należeć jednocześnie do wielu różnych grup).
- Każda grupa wierzchołków ma swoje własne wagi (więc ten sam wierzchołek siatki może mieć różne wagi — w zależności od grupy, na którą się powołasz).

W trybie *Weight Paint*, w zestawie *Editing* pojawia się dodatkowa panel *Paint* (Rysunek 15.37.4). To „paleta” do malowania wag. W górnej części jest zestaw kontrolki, za pomocą których ustalisz aktualną wartość, którą będziesz nanosił na wierzchołki (*Weight*), rozmiar „pędzla” (*Size*), oraz jego „nieprzejrzystość” (*Opacity*). Ten ostatni parametr jest uwzględniany podczas „mieszania” wartości, którą malujesz, z wartością wagi, którą już posiada „malowany” wierzchołek. Po prawej strony paneli możesz wybrać jeden z trybów tego mieszania (najczęściej używany jest domyślny *Mix*).

Waga 1.0 (100%) jest oznaczana kolorem czerwonym, a waga 0.0 — kolorem niebieskim. Wartościom pośrednim odpowiada spektrum pomiędzy tymi barwami. Powierzchnie z naniesionymi wagami mienia się więc w Blenderze wszystkimi kolorami tęczy.

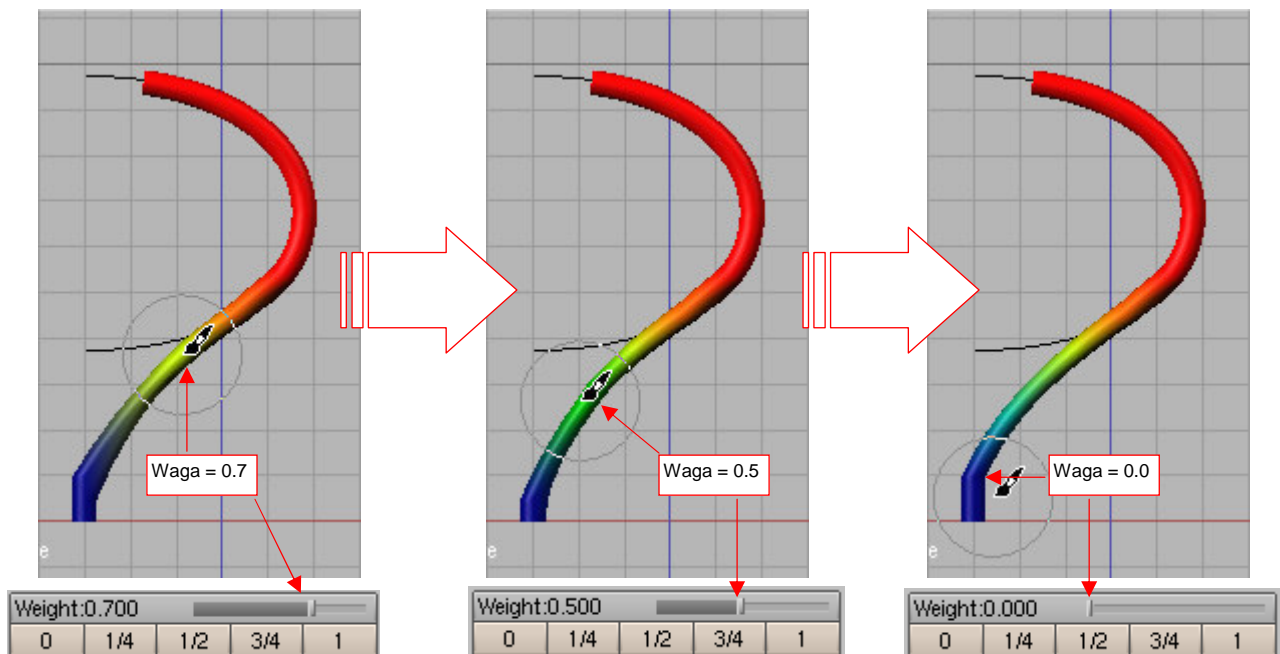
Ustaw najpierw w panelu *Paint* aktualną wagę (*Weight*) na 1.0 i umiarkowany rozmiar pędzla (*Size*) — 30. Malujesz, naciskając **LPM** ponad siatką. Zaczynaj nanosić wagę 100% od końca cylindra (Rysunek 15.37.5):



Rysunek 15.37.5 „Namalowanie” obszaru całkowicie wygiętego

Pomalowane wagą „100%” fragmenty siatki natychmiast „dokleją” się do krzywej, przypisanej przez modyfikator. Podczas malowania staraj się nie robić zbyt gwałtownych ruchów, by „trafić” z odpowiednią wagą na wszystkie ściany.

Po pomalowaniu „na czerwono” mniej więcej 60% długości cylindra, zacznij zmniejszać aktualną wagę co 0.1 (10%). Zbliżając się do drugiego końca powłoki, nanosząc coraz mniejsze wartości (Rysunek 15.37.6):

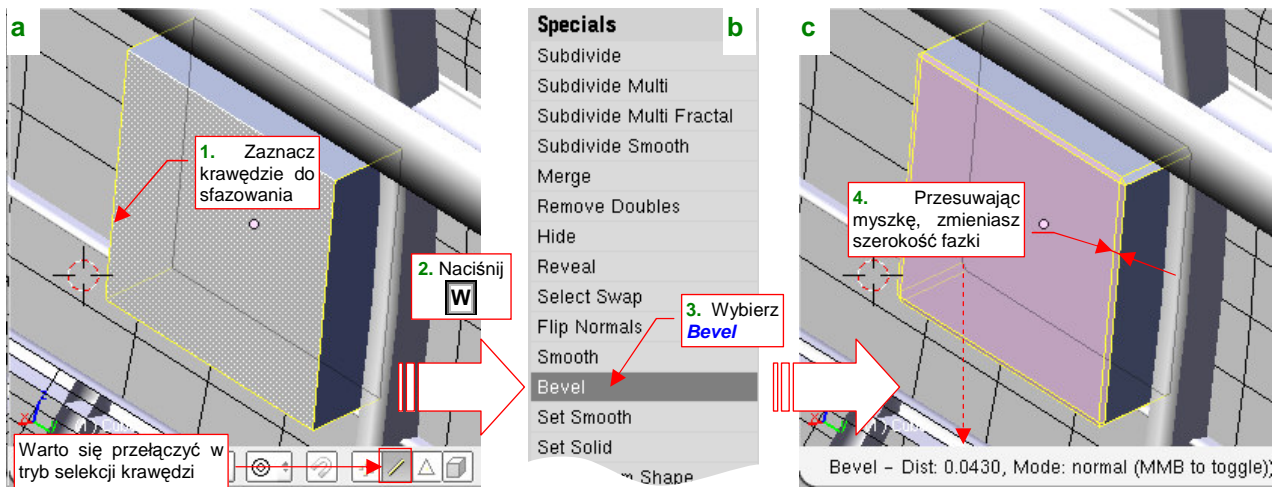


Rysunek 15.37.6 Namalowanie obszaru przejściowego

Spowoduje to pojawienie się na powłoce odcieni żółci, zieleni (waga = 50%), a potem błękitu. Pośrednie wagi nanosząc w ten sposób, by na końcu siatki pozostawić wagę = 0 (kolor niebieski). W ten sposób uzyskasz efekt, który pokazywał Rysunek 15.37.1c (str. 919).

15.38 Fazowanie i zaokrąglanie krawędzi (*Bevel*)

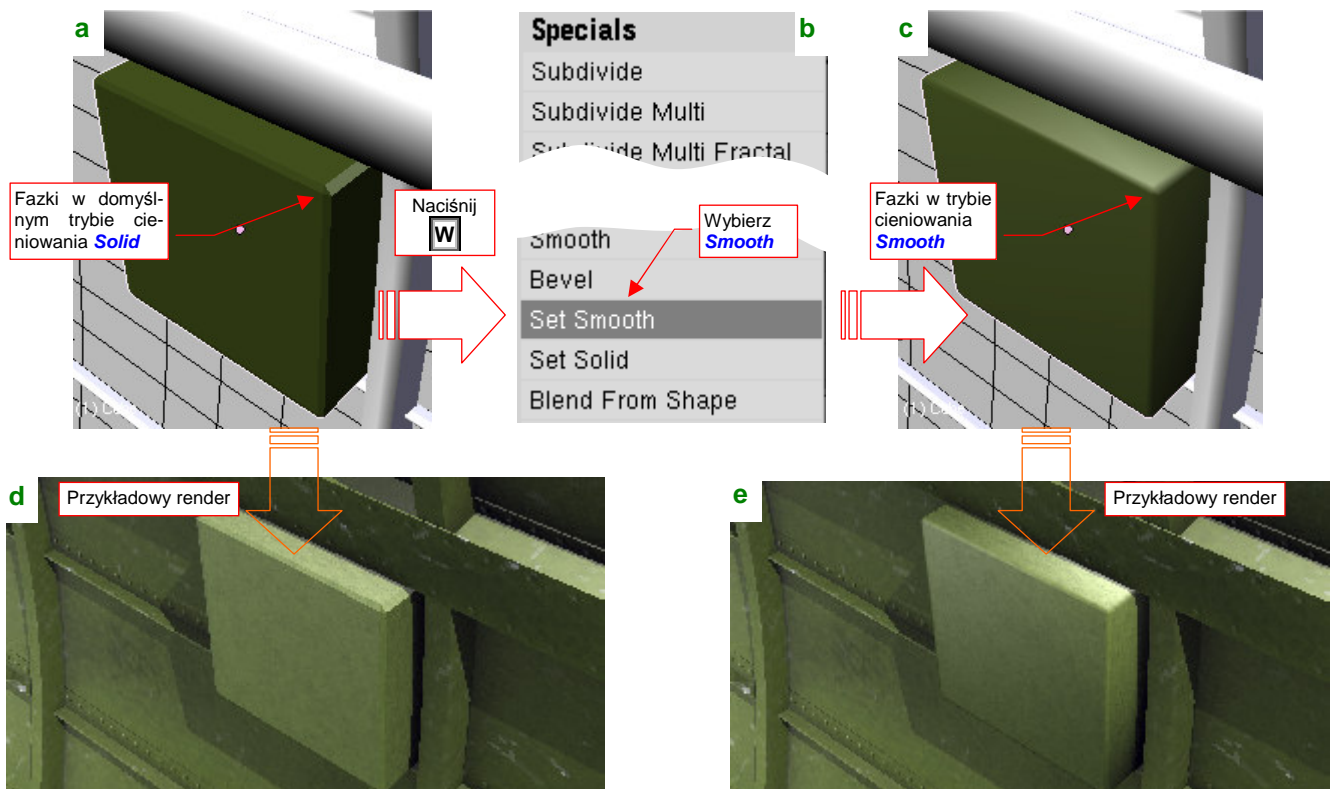
Najpierw zaznacz na siatce krawędzie, które chcesz sfazować. Najwygodniej jest na ten moment przełączyć się z domyślnego trybu selekcji wierzchołków w tryb selekcji krawędzi (Rysunek 15.38.1a):



Rysunek 15.38.1 Przebieg fazowania (*Bevel*)

Następnie naciśnij klawisz **W**, i z menu **Specials** wybierz polecenie **Bevel** (Rysunek 15.38.1b). (To samo polecenie znajdziesz w menu: **Mesh** → **Edges** → **Bevel**). Wzdłuż zaznaczonych krawędzi pojawią się fazki, których szerokość możesz zmieniać dynamicznie, przesuwając myszkę (Rysunek 15.38.1c). Gdy ustawisz właściwą szerokość, potwierdzasz ją naciskając **LPM**. (Możesz też w każdej chwili z niej zrezygnować, naciskając **Esc**).

Rysunek 15.38.2a,d) przedstawia rezultat operacji:



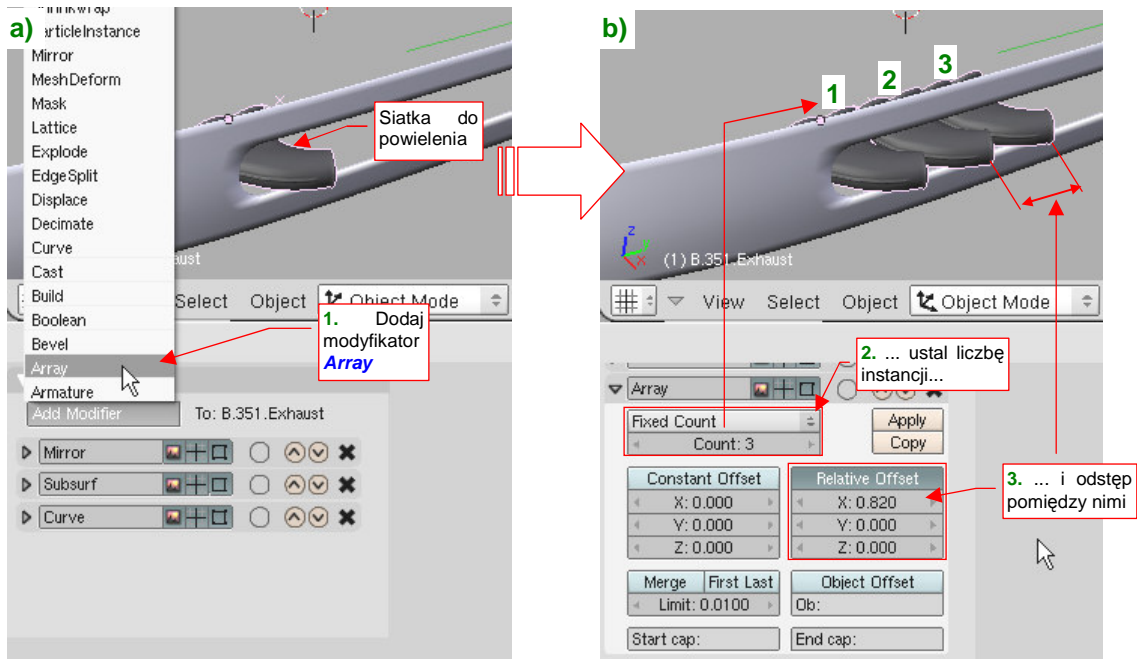
Rysunek 15.38.2 Użycie fazowania do zaokrąglania krawędzi siatki

Zazwyczaj fazujemy po to, by uzyskać wrażenie zaokrąglonych krawędzi. W tym celu należy zaznaczyć ściany siatki, i wybrać z menu **Specials** polecenie **Set Smooth** (Rysunek 15.38.2b). Efekt przedstawia Rysunek 15.38.2c,e).

15.39 Powielenie siatki (**Array**)

Siatkę można powielić poprzez sklonowanie obiektu, który ją zawiera (**Alt-D**, por. str. 796). Jeżeli jednak zależy Ci na szybkim rezultacie i równych odstępach pomiędzy poszczególnymi kopiami, użyj zamiast tego modyfikatora **Array**. Dla przykładu przekształcę za pomocą modyfikatorów pojedynczą rurę wydechową w pełen kolektor spalin. W oryginalnym samolocie cały ten zespół składał się z 12 takich rur, po sześć z każdej strony.

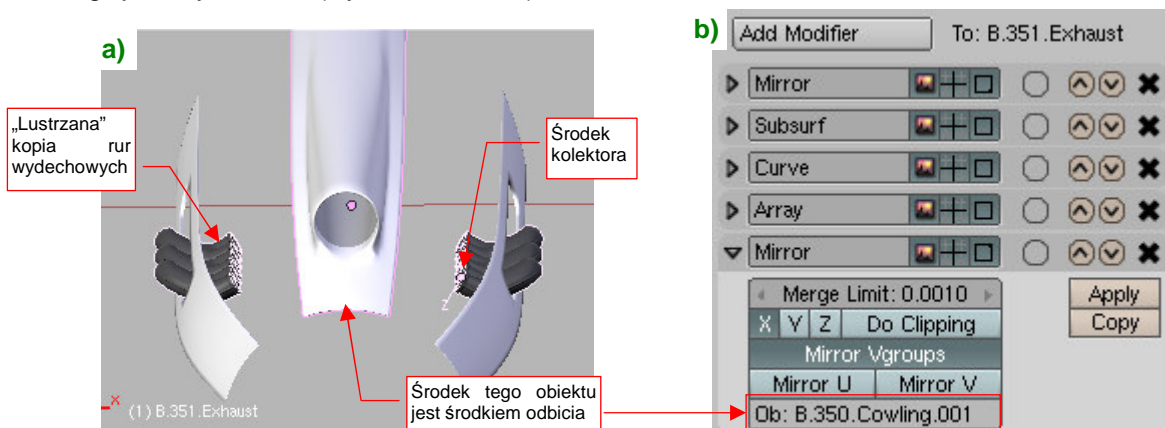
Na początek wygenerujemy za pomocą modyfikatora pierwsze trzy (Rysunek 15.39.1):



Rysunek 15.39.1 Użycie modyfikatora **Array**

Zaznacz odpowiedni obiekt, i w z listy **Add Modifier** wybierz **Array** (Rysunek 15.39.1a). W panelu modyfikatora **Array**, który się pojawi, ustaw licznik instancji (**Count**) na **3**, i dystans pomiędzy nimi (**Relative Offset**) na **0.82** (Rysunek 15.39.1b). Jak sam widzisz na ilustracji, modyfikator **Array** ma jeszcze wiele innych przełączników (a więc i możliwości). Nie były mi jednak potrzebne do stworzenia modelu kolektora spalin, więc nie będę ich tutaj omawiał. (Mówiąc szczerze: jeszcze ich nie sprawdziłem).

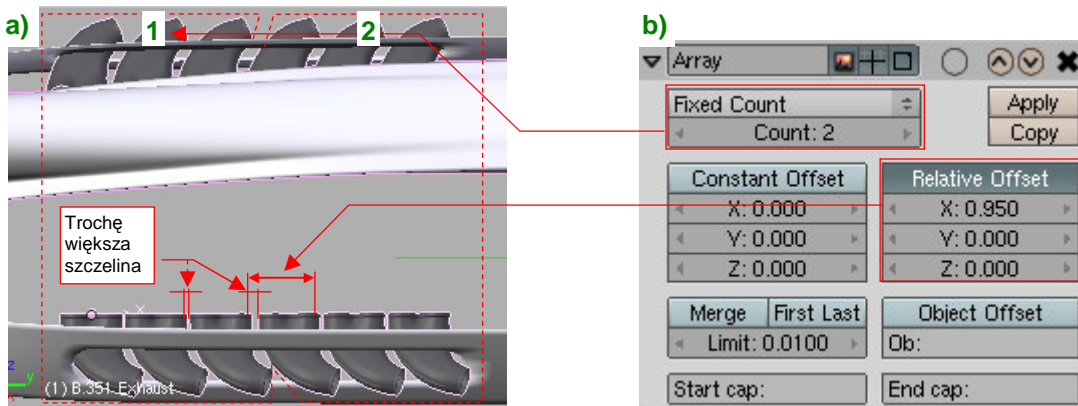
Dlaczego stworzyłem 3 rury, a nie docelowe 6? Wyjaśnię za chwilę. Na razie stwórzmy lustrzane odbicie kolektora z drugiej strony kadłuba (Rysunek 15.39.2):



Rysunek 15.39.2 Powielenie kolektora spalin za pomocą modyfikatora **Mirror**

Wykorzystałem w tym celu modyfikator **Mirror**, wskazując mu jako punkt odniesienia, środek innego obiektu. (Ten obiekt ma środek w osi kadłuba, więc druga część kolektora pojawi się we właściwym miejscu).

Wróćmy do pytania: dlaczego nie stworzyłem od razu rzędu sześciu rur wydechowych? Ponieważ w odwzorowywanym silniku były podzielone na dwie sekcje, po 3 rury każda. Odstęp pomiędzy rurą trzecią i czwartą (na granicy „sekcji”) był nieco większy niż pomiędzy pozostałymi. Aby ten efekt uzyskać, wystarczy jeszcze raz użyć modyfikatora **Array** (Rysunek 15.39.3):

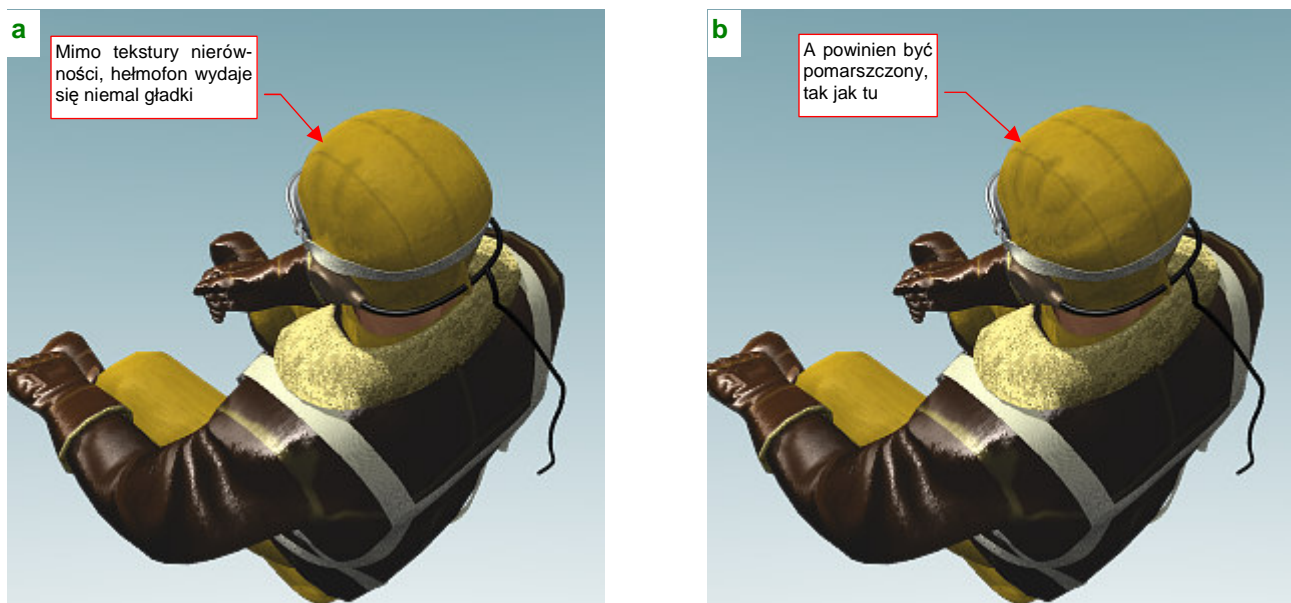


Rysunek 15.39.3 Dodanie kolejnych rur wydechowych — powtórne użycie modyfikatora **Array**

Tym razem powielamy cały rezultat poprzednich modyfikatorów (Rysunek 15.39.3a), więc należy w licznik (**Count**) wpisać 2 (Rysunek 15.39.3b). Także odstęp (**Relative Offset**) pomiędzy segmentami należy zrobić nieznacznie większy — 0.95, a nie 0.82. (Jest o całe 7 mm szerszy — to coś dla koneserów silników Allison ☺).

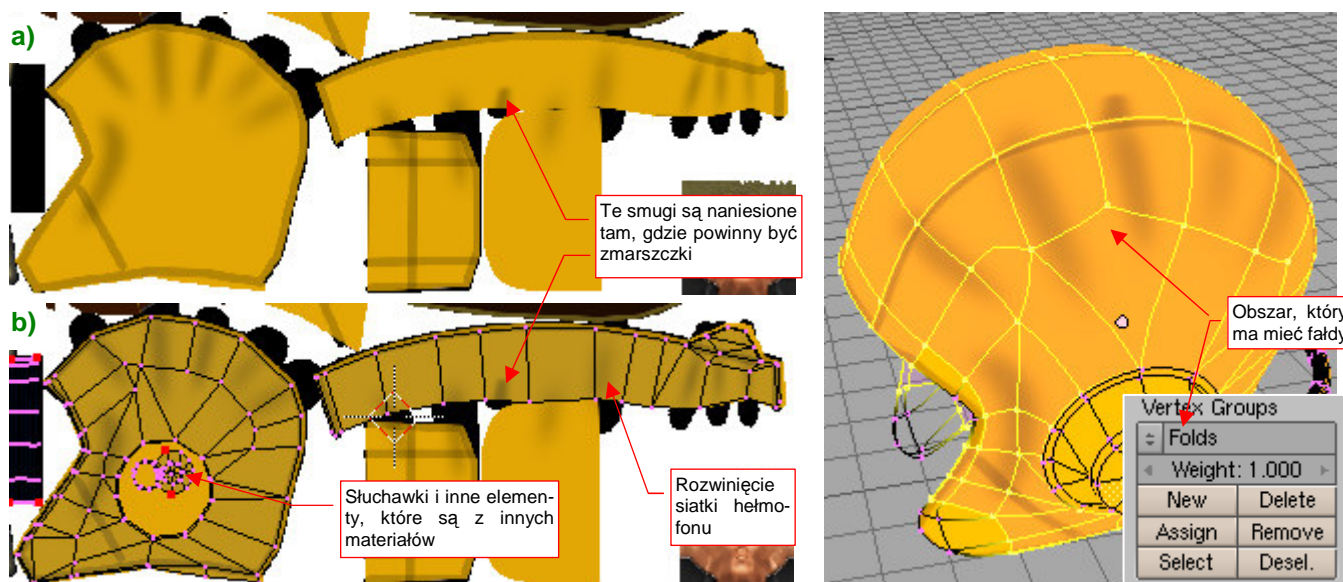
15.40 Deformacja według tekstury (*Displace*)

Czasami potrzebna jest deformacja siatki, która wprowadzi w jej kształt jakiś „nieład”. Na przykład — hełmofon pilota, który pokazuje Rysunek 15.40.1a), wydaje się gładki. Właściwie taki kształt wyglądałby w miarę poprawnie, gdyby to nakrycie z głowy było ze skóry. Niestety, w tym przypadku jest to płócienna „pilotka” USAAC, która powinna się marszczyć na szwach (Rysunek 15.40.1b):



Rysunek 15.40.1 Hełmofon pilota: gładki i z fałdami

Kolory hełmofonu i linie szwów, które pokazuje Rysunek 15.40.1, uzyskałem za pomocą tekstury (Rysunek 15.40.2a). To fragment większego obrazu, naniesiony na rozwinięcie **UV** siatki hełmofonu (Rysunek 15.40.2b):



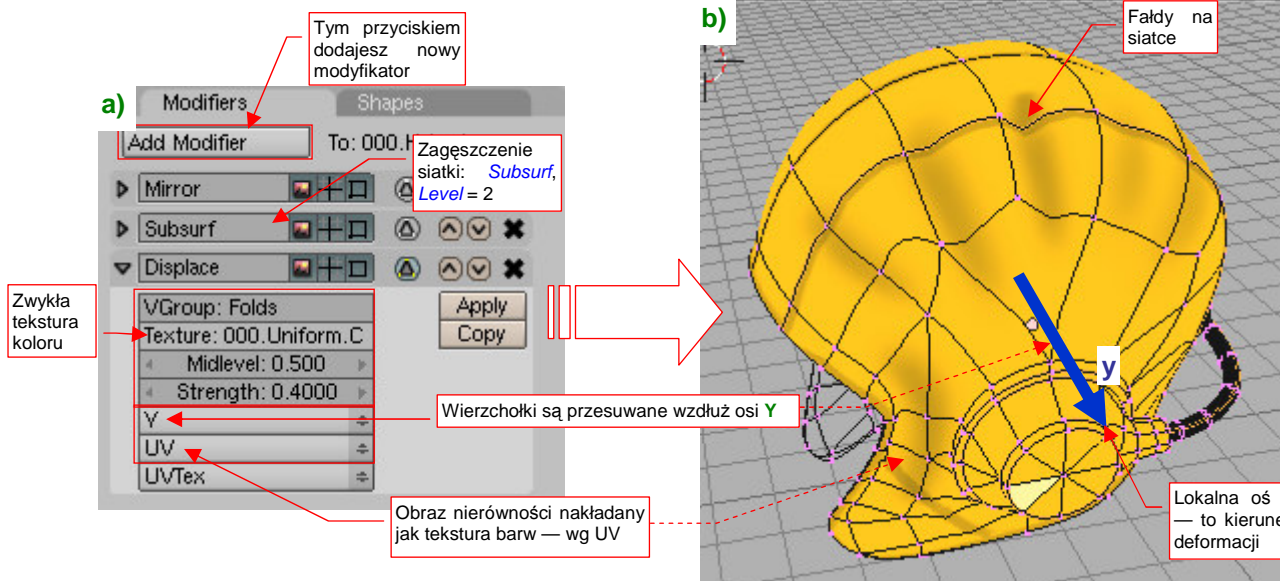
Rysunek 15.40.2 Obraz, nałożony na rozwinięciu UV hełmofonu

Rysunek 15.40.3 Definicja grupy **Folds**

Tekstura, przedstawiana przez Rysunek 15.40.2, służy zarówno do mapowania koloru (*Col*), jak i nierówności (*Nor*). Jak pokazuje render (Rysunek 15.40.1a), wychodzi to zbyt słabo.

Skoro zwykła tekstura nierówności jest zbyt „słaba”, musimy nanieść na siatkę hełmofonu prawdziwe fałdy. Aby to było możliwe, najpierw należy ją odpowiednio zagęścić (modyfikatorem *Subsurf*, z *Level* = 2). Warto także wydzielić nową grupę wierzchołków (por. str. 903), które będą podlegały tej deformacji. Zaznacz odpowiednie wierzchołki i nadaj tej *Vertex Group* nazwę **Folds** (Rysunek 15.40.3).

W charakterze obrazu nierówności wykorzystamy teksturę barwy. Jest dopasowana do rozwinięcia **UV** siatki, więc uzyskanie pofałdowania jest bardzo proste. Wystarczy dodać na „wierzch” stosu modyfikatorów siatki kolejny, o nazwie **Displace** (Rysunek 15.40.4a):



Rysunek 15.40.4 Dodanie modyfikatora **Displace** i jego wpływ na siatkę

W poszczególnych polach modyfikatora wpisujemy:

- **VGroup**: nazwę grupy wierzchołków, na której mają się pojawić „fałdki” (w tym przypadku — **Folds**);
- **Texture**: nazwa tekstury (w tym przypadku ta sama, którą stosujemy w materiale hełmofonu: **000.Uniform.Color**) — chodzi tu o obraz przez Rysunek 15.40.2a);
- **Midlevel**: poziom odniesienia (0.0..1.0). Barwy ciemniejsze będą tworzyć wgłębienia, jaśniejsze — wypukłości. Ustawiłem na 0.5;
- **Strength**: skala (wysokość) deformacji — dobrałem tak, by „fałdki” nie były zbyt wysokie (0.4).

Dodatkowo w polach bez etykiety wybrałem: kierunek deformacji: **Y**. (Domyślne ustawienie — **Normal** — tworzyło na głowie pilota wybrzuszenia godne starej babci). Typ mapowania — **UV** — pozwala wykorzystać przedstawiony przez chwilę obraz fałdek (nie ma co wymyślać ich po raz drugi). Rysunek 15.40.4b) przedstawia działanie modyfikatora **Displace**. Drobne ściany, stworzone przez **Subsurf**, zostały przesunięte wzdłuż osi **Y**.

Rysunek 15.40.5 przedstawia render tak zmodyfikowanego hełmofonu:

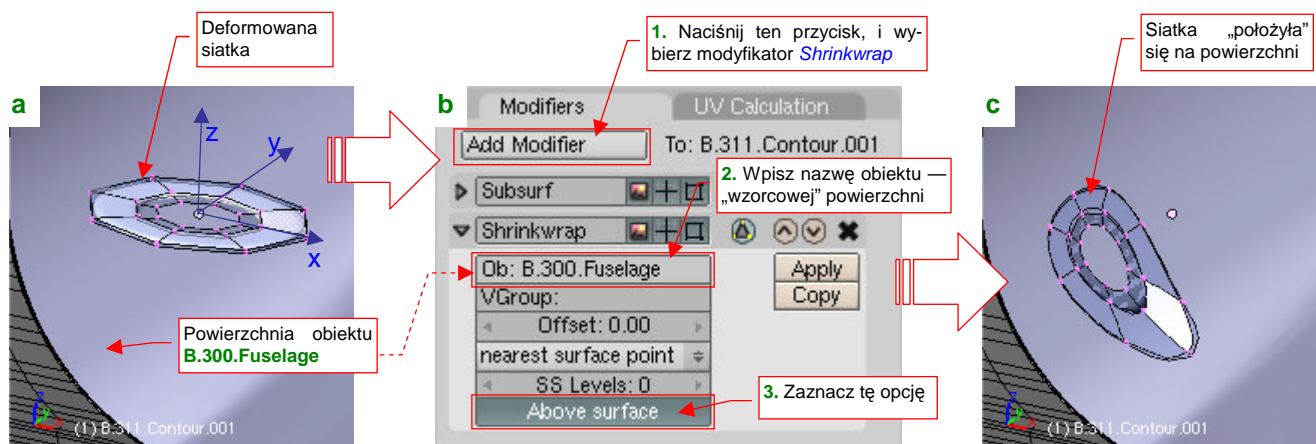


Rysunek 15.40.5 Rezultat (render hełmofonu)

Zwróć uwagę, że dzięki **Displace** mogliśmy zachować do modyfikacji oryginalną, niezdeformowaną siatkę.

15.41 Rzutowanie na powierzchnię (*Shrinkwrap*)

Modyfikator *Shrinkwrap* pozwala położyć siatkę na wskazanej powierzchni („jak naleśnik”) (Rysunek 15.41.1):

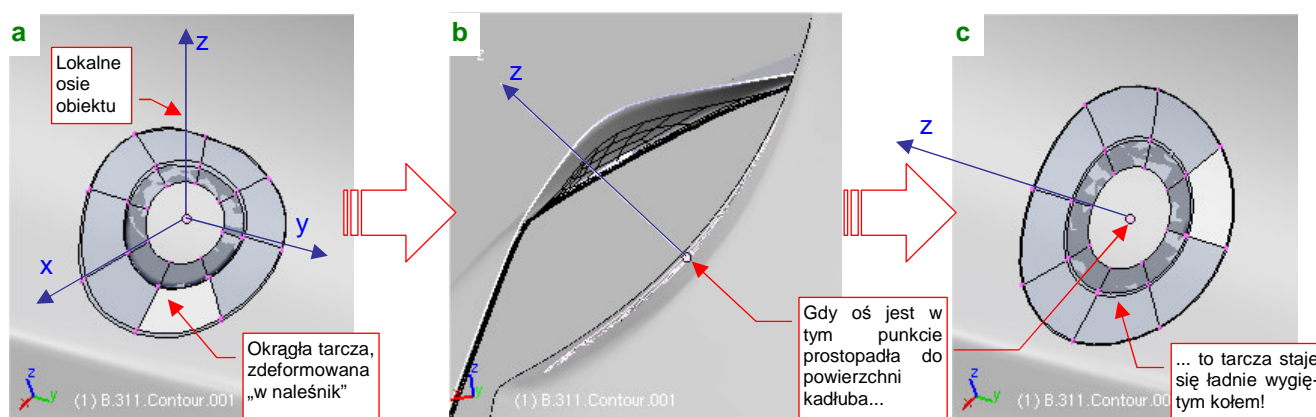


Rysunek 15.41.1 Rzutowanie siatki na inną powierzchnię

Dla przykładu „położymy” na powierzchni kadłuba (obiekt **B.300.Fuselage**) siatkę w kształcie pierścienia (Rysunek 15.41.1a). W zestawie *Editing*, panelu *Modifiers*, naciśnij przycisk **Add Modifier**. Spowoduje to rozwinięcie listy dostępnych modyfikatorów siatki. Wybierz spośród nich **Shrinkwrap**. W panelu modyfikatora (Rysunek 15.41.1b) wpisz w pole **Ob** nazwę obiektu — powierzchni, na którą rzutujemy siatkę. Dodatkowo możesz zaznaczyć jeszcze opcję **Above surface**. W rezultacie nasz pierścień „opadł” na kadłub, zupełnie tak jak naleśnik (Rysunek 15.41.1c)

Na razie ten rezultat jest daleki od satysfakcjonującego: chcielibyśmy, aby nasz pierścień po deformacji wyglądał jak wygięte koło, a nie jak jakiś kawałek gumy!

Na szczęście nie jest to takie trudne do osiągnięcia (Rysunek 15.41.2):



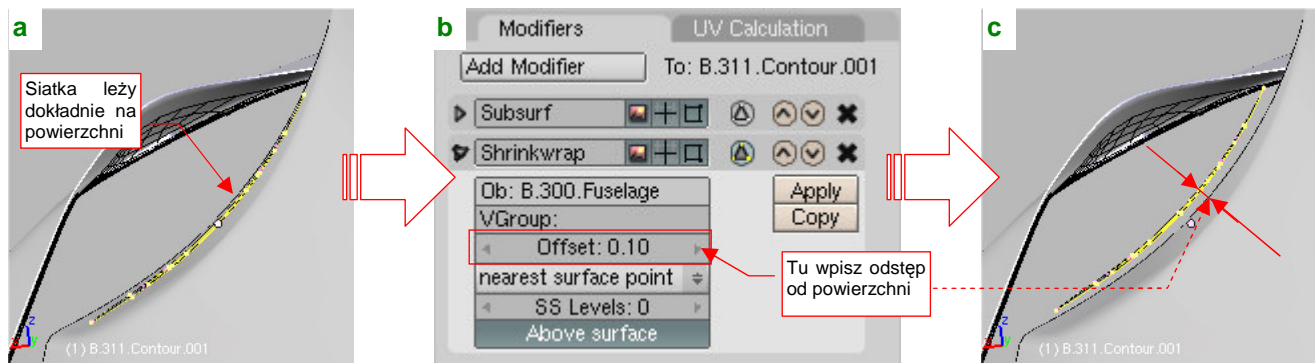
Rysunek 15.41.2 Korygowanie kształtu rzutu

Umieść niezdeformowaną siatkę tak, by środek pierścienia znalazł się na powierzchni kadłuba. (W tym przykładzie to jednocześnie środek obiektu). Następnie obróć obiekt tak, by powierzchnia siatki przed deformacją stała się styczna w tym punkcie (Rysunek 15.41.2b). Widzisz, co się stało?

- Kształt rzutu siatki na powierzchnię zależy od orientacji w przestrzeni obiektu rzutowanego

Jeżeli przed deformacją środek siatki będzie już leżał na powierzchni odniesienia (tutaj: kadłubie), i cała siatka będzie styczna do powierzchni w tym punkcie, to zmiana kształtu rzutowanej siatki będzie minimalna.

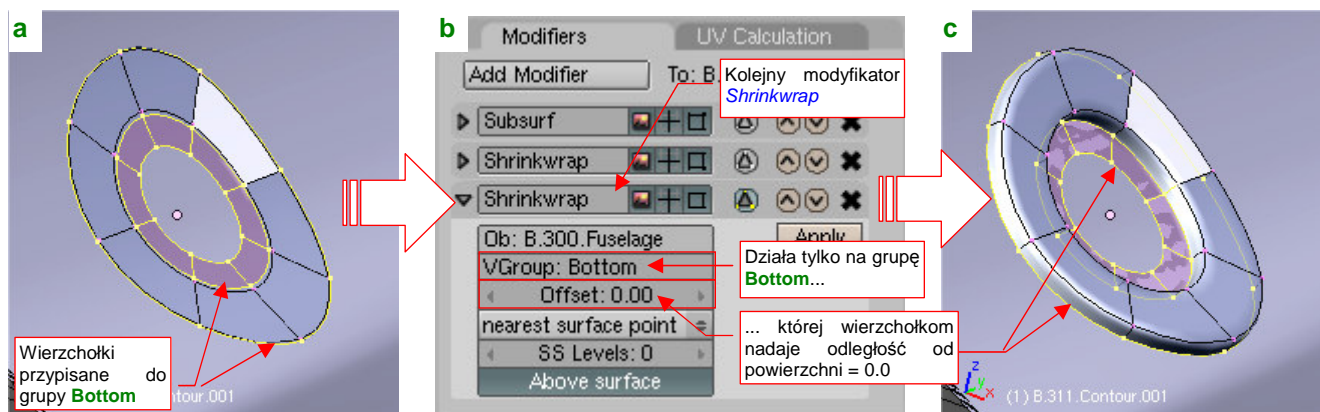
Na razie nasza siatka leży dokładnie na powierzchni kadłuba (Rysunek 15.41.3a):



Rysunek 15.41.3 Odsunięcie od powierzchni

Przy modelowaniu konstrukcji zazwyczaj jednak chcemy, by miała pewną grubość. Do ustalania tego odstęp służy pole **Offset**, umieszczone na panelu modyfikatora (Rysunek 15.41.3b). Gdy wpiszesz tam jakąś niezerową wartość, wygięta siatka ulegnie odsunięciu (Rysunek 15.41.3c).

To jednak dopiero połowa sukcesu. W tej chwili nasz wygięty pierścień „lewituje” nad kadłubem. Aby uzyskać solidne boczne ścianki, zaznacz zewnętrzne krawędzie siatki, i przypisz je do jakiejś nowej grupy wierzchołków (por. str. 903). Nadajmy jej nazwę, np. **Bottom** (Rysunek 15.41.4a):



Rysunek 15.41.4 Nadanie rzutowanej siatce „grubości”

Następnie dodaj kolejny modyfikator **Shrinkwrap** (Rysunek 15.41.4b). Tym razem jednak wpisz w pole **VGroup** nazwę **Bottom**, by działał tylko na krawędzie należące do tej grupy. Potem zmniejsz w tym modyfikatorze **Offset** do zera. W efekcie mamy oczekiwany rezultat: wygięty pierścień z bocznymi ściankami, nadające siatce „grubość”.