

Oracle Designer

Generowanie bazy danych

Dane wejściowe

Diagramy związków encji, a w szczególności:

- definicje encji wraz z atrybutami
- definicje związków między encjami
- definicje dziedzin atrybutów encji

Wynik

Baza danych projektowanego systemu

Przebieg procesu

krok 1. Transformowanie diagramów związków encji do schematu logicznego bazy danych

krok 2. Generowanie schematu fizycznego bazy danych

Generowanie bazy danych

krok 1.

Transformowanie diagramów związków encji do schematu logicznego bazy danych

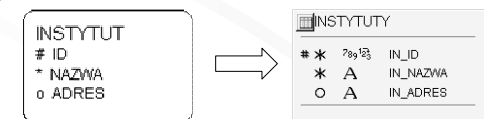
Reguły transformacji

Jak przetransformować:

- encję?
- hierarchię encji?
- związek?

Transformacja encji

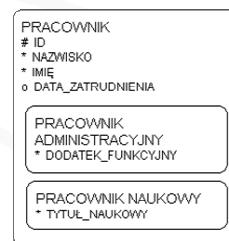
- Encja \Rightarrow relacja
- Atrybut encji \Rightarrow kolumna relacji
- Typ atrybutu \Rightarrow typ kolumny
- Dziedzina atrybutu \Rightarrow ograniczenie *check*
- Unikalny identyfikator encji \Rightarrow klucz podstawowy relacji



Transformacja hierarchii encji

Sposoby:

- transformacja do pojedynczej relacji
- transformacja do oddzielnych relacji
- transformacja do oddzielnych relacji połączonych ograniczeniami referencyjnymi w łańcuch



Transformacja hierarchii

Sposób pierwszy

Zasady:

- jedna relacja
- schemat relacji: atrybuty wszystkich encji z hierarchii + dodatkowa kolumna, określająca typ specjalizacji

Kiedy stosować:

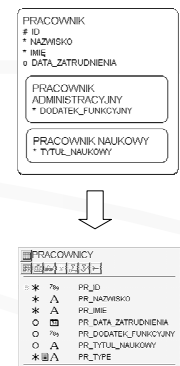
- większość atrybutów w nadtypie
- większość związków do nadtypu

Zalety:

- uproszczenie schematu bazy danych

Wady:

- atrybuty obowiązkowe podtypu stają się kolumnami opcjonalnymi



Sposób drugi

Zasady:

- jedna relacja dla każdego podtypu
- schemat relacji: atrybuty nadtypu + atrybuty podtypu

Kiedy stosować:

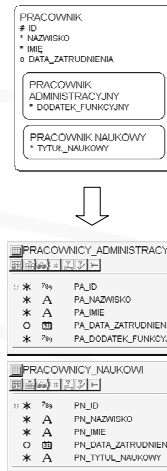
- większość atrybutów w podtypach
- większość związków do podtypów

Zalety:

- zachowanie obowiązkowości atrybutów w podtypach

Wady:

- komplikacja schematu
- konieczność powielenia kluczy obcych implementujących związki przyłączone do nadtypu



Sposób trzeci

Zasady:

- jedna relacja z atrybutami wspólnymi, dla każdego podtypu
- osobna relacja z jego atrybutami specyficznymi
- relacje połączone kluczami obcymi w łuku

Kiedy stosować:

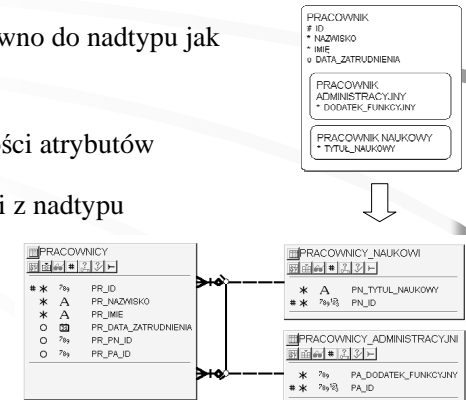
- związki przywiązane zarówno do nadtypu jak i podtypów

Zalety:

- zachowanie obowiązkowości atrybutów w podtypach
- łatwy dostęp do informacji z nadtypu

Wady:

- komplikacja schematu
- konieczność stosowania połączeń (SQL)



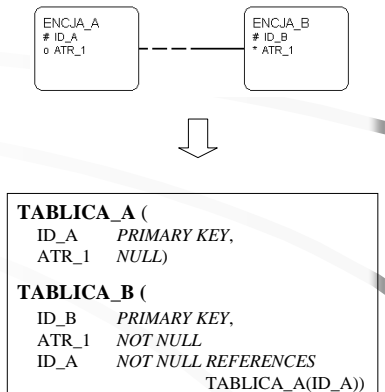
Transformacja związków

- Implementacja związku za pomocą ograniczeń referencyjnych (klucze obcych)
- Sposób transformacji zależy od parametrów związku:
 - krotności (1:1, 1:N, M:N)
 - obowiązkowości/opcjonalności

Związek 1:1 jednostronnie obowiązkowy

Zasady:

- do relacji impl. encję wiążaną obowiązkowo zostaje dodany klucz obcy, wskazujący na klucz podstawowy relacji impl. encję wiążaną opcjonalnie (z drugiej strony związku)
- na kolumny klucza obcego zostaje nałożone ograniczenie *not null*



Związek 1:1 obustronnie opcjonalny

Zasady:

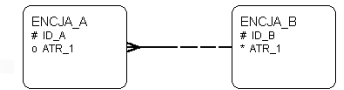
- do relacji impl. tą encję ze związku, dla której określono większy średni przyrost wystąpień, zostaje dodany klucz obcy, wskazujący na klucz podstawowy z relacji impl. drugą encję w związku
- na kolumny klucza obcego nałożone zostaje ograniczenie *null*



Związek 1:N

Zasady:

- do relacji impl. encję po stronie „N” związku zostaje dodany klucz obcy, wskazujący na klucz podstawowy relacji impl. encję po stronie „1” związku
- obowiązkowość związku po stronie „N” - ograniczenie *not null* na kolumny w kluczu obcym
- opcjonalność związku po stronie „N” - ograniczenie *null* na kolumny w kluczu obcym
- obowiązkowość/opcjonalność związku po stronie „1” nie ma wpływu na transformację



```

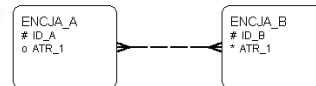
TABLICA_A (
  ID_A    PRIMARY KEY,
  ATR_1   NULL
  ID_B    NOT NULL REFERENCES
           TABLICA_B(ID_B))

TABLICA_B (
  ID_B    PRIMARY KEY,
  ATR_1   NOT NULL)
    
```

Związek M:N

Zasady:

- zostaje utworzona nowa relacja
- w nowej relacji zostają utworzone klucze obce, wskazujące na klucze podstawowe relacji w związku
- kolumny obu kluczy obcych tworzą klucz podstawowy relacji



```

TABLICA_A (
  ID_A    PRIMARY KEY,
  ATR_1   NULL)

TABLICA_B (
  ID_B    PRIMARY KEY,
  ATR_1   NOT NULL)

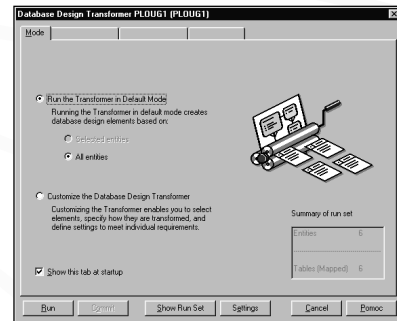
TABLICA_A_B (
  ID_A    NOT NULL REFERENCES
           TABLICA_A(ID_A),
  ID_B    NOT NULL REFERENCES
           TABLICA_B(ID_B),
  PRIMARY KEY(ID_A, ID_B))
    
```

Proces transformacji

Krok 1.



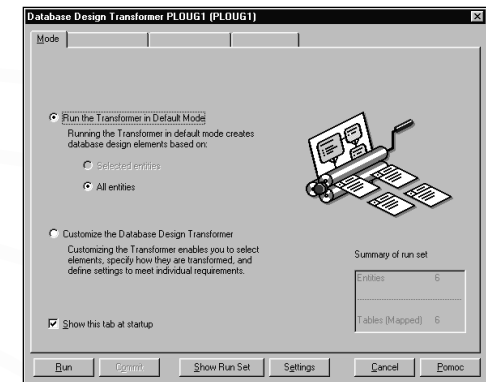
Uruchomić narzędzie *Database Design Transformer* z grupy *Transform Preliminary Designs*



Krok 2 - opcje transformacji



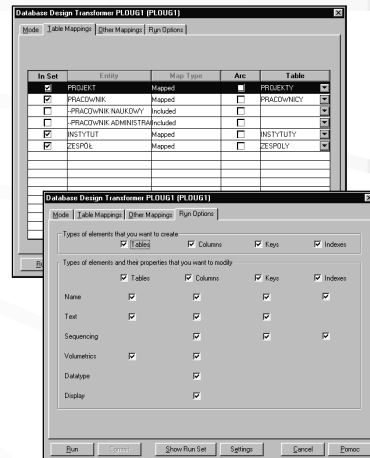
- transformacja wg ustawień domyślnych
- transformacja wg ustawień użytkownika



Dostępne ustawienia



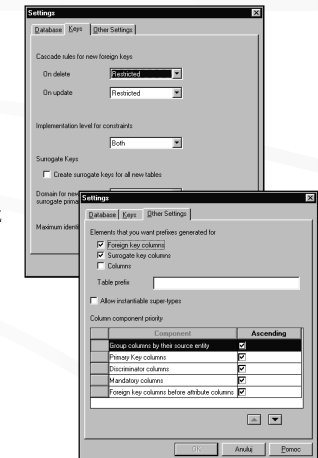
- wybór encji do transformacji - domyślnie wszystkie
- sposób transformacji hierarchii - domyślnie do jednej relacji
- wybór typów tworzonych elementów (relacje, kolumny, klucze, indeksy) - domyślnie wszystkie
- wybór typów modyfikowanych elementów (istniejących już w repozytorium relacji, kolumn, kluczy, indeksów) - domyślnie żadne



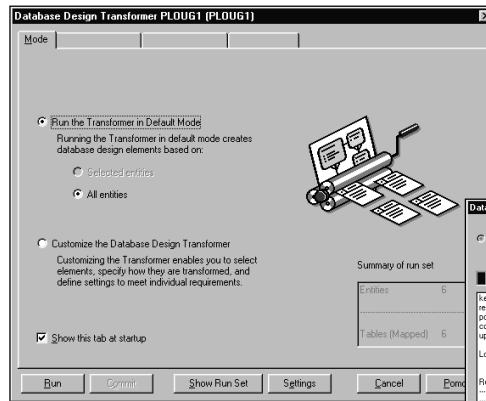
Dostępne ustawienia (2)



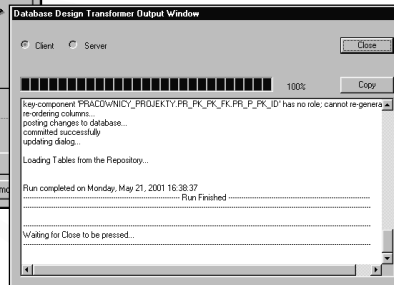
- opcje ograniczeń referencyjnych:
 - usuwanie kaskadowe - domyślnie zabronione
 - modyfikowanie kaskadowe - domyślnie zabronione
- tworzenie sztucznych kluczy podstawowych relacji (w postaci dodatkowej kolumny numerycznej) - domyślnie tylko dla encji bez unikalnych identyfikatorów
- przedrostek nazw relacji - domyślnie brak
- przedrostki nazw kolumn (na podstawie krótkich nazw encji) - domyślnie brak



Krok 3 - uruchomienie procesu



Uruchomienie transformacji - przycisk *Run*



Wynik



Umieszczone repozytorium systemu definicje:

- relacji
- kolumn
- ograniczeń integralnościowych
- indeksów
- liczników - dla sztucznych kluczy podstawowych

Wynik (2)

Podgląd definicji w repozytorium - narzędzie *Design Editor* z grupy *Design and Generate*



Design Editor



Umożliwia:

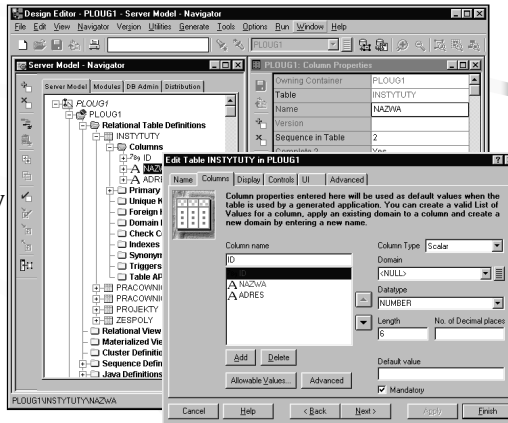
- przeglądanie i ręczną modyfikację powstałego w wyniku transformacji schematu logicznego bazy danych
- definiowanie dodatkowych obiektów schematu logicznego:
 - liczników
 - perspektyw
 - kodu PL/SQL
- utworzenie diagramu schematu modelu relacyjnego - pokazuje połączenia między relacjami (ograniczenia referencyjne)

Przeglądanie i modyfikacja schematu logicznego



Zakładka *Server Model*, gałęzie:

- *Relational Table Definitions* - relacje, kolumny, ograniczenia integralnościowe, inne
- *Relational View Definition* - perspektywy
- *Sequence Definitions* - liczniki
- *PL/SQL Definitions* - kod składany

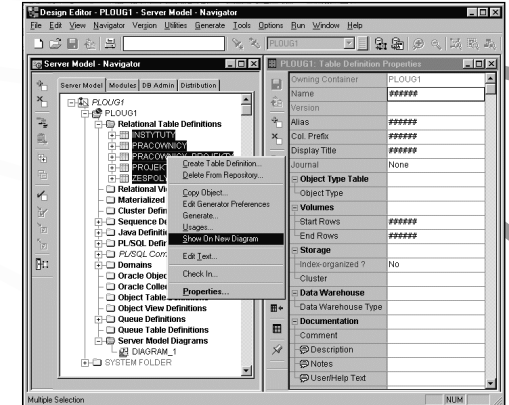


(C) Instytut Informatyki, Politechnika Poznańska

Tworzenie diagramu schematu logicznego

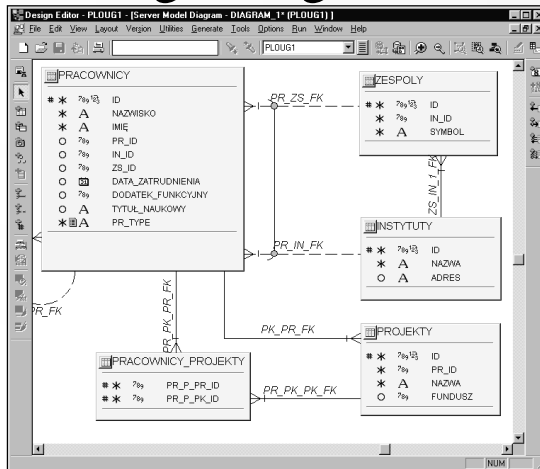


- Zaznaczyć obiekty (relacje lub perspektywy), które mają być uwidocznione na diagramie
- Z menu kontekstowego wybrać *Show on New Diagram*



(C) Instytut Informatyki, Politechnika Poznańska

Przykładowy diagramu schematu logicznego

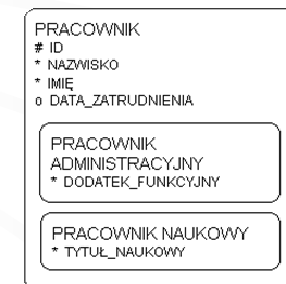


(C) Instytut Informatyki, Politechnika Poznańska

27

Jak to zrobić?

Jak przetransformować hierarchię encji w sposób inny niż domyślny?



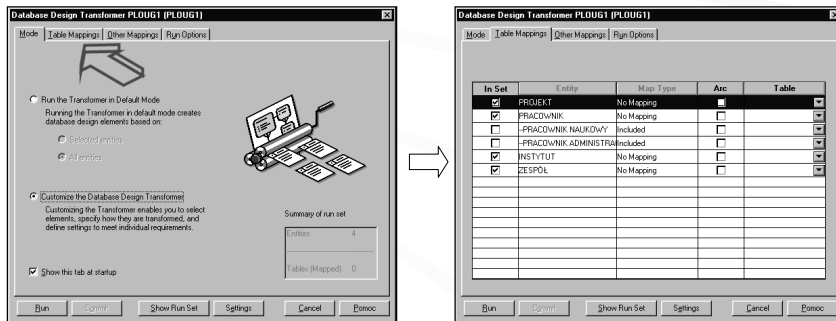
(C) Instytut Informatyki, Politechnika Poznańska

28

Transformacja do oddzielnych relacji

krok 1. Uruchomić *Database Design Transformer*

krok 2. Zaznaczyć opcję *Customize the Database Transformer* i wybrać zakładkę *Table Mappings*



Transformacja do oddzielnych relacji

krok 3. Zmienić zbiór encji do transformacji - wyłączyć ze zbioru encję-nadtyp, dodać encje-podtypy

| In Set | Entity | Map Type | Arc | Table |
|-------------------------------------|----------------------|------------|--------------------------|-------|
| <input type="checkbox"/> | PROJEKT | No Mapping | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | PRACOWNIK | No Mapping | <input type="checkbox"/> | |
| <input type="checkbox"/> | -PRACOWNIK_NAUKOWY | Included | <input type="checkbox"/> | |
| <input type="checkbox"/> | -PRACOWNIK_ADMINISTR | Included | <input type="checkbox"/> | |
| <input type="checkbox"/> | INSTYTUT | No Mapping | <input type="checkbox"/> | |
| <input type="checkbox"/> | ZESPOL | No Mapping | <input type="checkbox"/> | |



| In Set | Entity | Map Type | Arc | Table |
|-------------------------------------|----------------------|------------|--------------------------|-------|
| <input type="checkbox"/> | PROJEKT | No Mapping | <input type="checkbox"/> | |
| <input type="checkbox"/> | PRACOWNIK | No Mapping | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | -PRACOWNIK_NAUKOWY | Included | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | -PRACOWNIK_ADMINISTR | Included | <input type="checkbox"/> | |
| <input type="checkbox"/> | INSTYTUT | No Mapping | <input type="checkbox"/> | |
| <input type="checkbox"/> | ZESPOL | No Mapping | <input type="checkbox"/> | |

Transformacja do oddzielnych relacji

krok 4. Przystąpić do transformacji

Wynik:

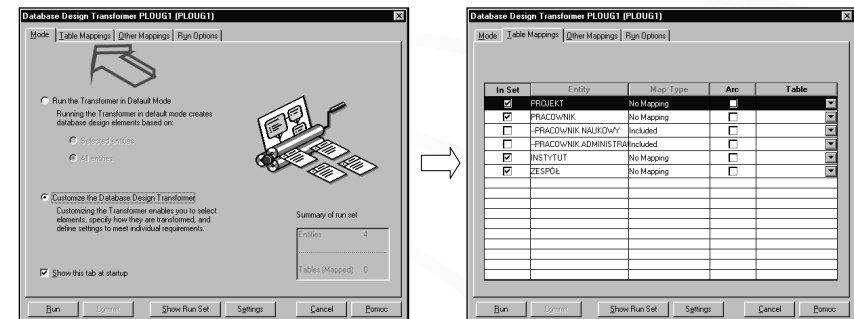
| PRACOWNICY_ADMINISTRACYJNI | |
|----------------------------|----------------------|
| * 7b | PA_ID |
| * A | PA_NAZWISKO |
| * A | PA_IMIE |
| o | PA_DATA_ZATRUDNIENIA |
| * 7b | PA_DODATEK_UNIKOJNY |

| PRACOWNICY_NAUKOWI | |
|--------------------|----------------------|
| * 7b | PN_ID |
| * A | PN_NAZWISKO |
| * A | PN_IMIE |
| o | PN_DATA_ZATRUDNIENIA |
| * A | PN_TYTUL_NAUKOWY |

Transformacja do relacji w łuku

krok 1. Uruchomić *Database Design Transformer*

krok 2. Zaznaczyć opcję *Customize the Database Transformer* i wybrać zakładkę *Table Mappings*



Transformacja do relacji w łuku

krok 3. Zmienić zbiór encji do transformacji - włączyć do zbioru encję-nadtyp wraz z encjami-podtypami

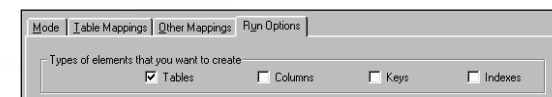
| In Set | Entity | Map Type | Arc | Table |
|-------------------------------------|----------------------|------------|--------------------------|-------|
| <input type="checkbox"/> | PROJEKT | No Mapping | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | PRACOWNIK | No Mapping | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | -PRACOWNIK NAUKOWY | Included | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | -PRACOWNIK ADMINISTR | Included | <input type="checkbox"/> | |
| <input type="checkbox"/> | INSTYTUT | No Mapping | <input type="checkbox"/> | |
| <input type="checkbox"/> | ZESPÓŁ | No Mapping | <input type="checkbox"/> | |



| In Set | Entity | Map Type | Arc | Table |
|-------------------------------------|----------------------|------------|--------------------------|-------|
| <input type="checkbox"/> | PROJEKT | No Mapping | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | PRACOWNIK | No Mapping | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | -PRACOWNIK NAUKOWY | Included | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | -PRACOWNIK ADMINISTR | Included | <input type="checkbox"/> | |
| <input type="checkbox"/> | INSTYTUT | No Mapping | <input type="checkbox"/> | |
| <input type="checkbox"/> | ZESPÓŁ | No Mapping | <input type="checkbox"/> | |

Transformacja do relacji w łuku

krok 4. Zmienić typ elementów do transformacji - zakładka *Run Options* - tylko definicje relacji (bez kolumn i ograniczeń integralnościowych)



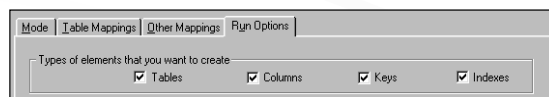
krok 5. Uruchomić transformację. Wygenerowane zostaną jedynie definicje relacji. Pozostać w narzędziu

Transformacja do relacji w łuku

krok 6. Przy encjach-podtypach zaznaczyć opcję *Arc*

| In Set | Entity | Map Type | Arc | Table |
|-------------------------------------|----------------------|------------|-------------------------------------|----------------|
| <input type="checkbox"/> | PROJEKT | No Mapping | <input type="checkbox"/> | |
| <input checked="" type="checkbox"/> | PRACOWNIK | Mapped | <input type="checkbox"/> | PRACOWNICY |
| <input checked="" type="checkbox"/> | -PRACOWNIK NAUKOWY | Arc | <input checked="" type="checkbox"/> | PRACOWNICY_NAU |
| <input checked="" type="checkbox"/> | -PRACOWNIK ADMINISTR | Arc | <input checked="" type="checkbox"/> | PRACOWNICY_ADM |
| <input type="checkbox"/> | INSTYTUT | No Mapping | <input type="checkbox"/> | |
| <input type="checkbox"/> | ZESPÓŁ | No Mapping | <input type="checkbox"/> | |

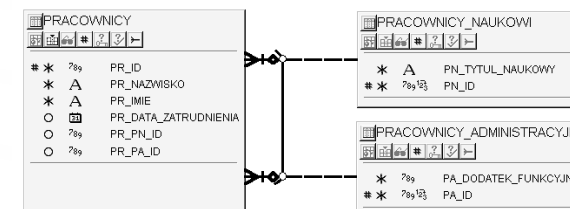
krok 7. Zmienić typ elementów do transformacji - zakładka *Run Options* - wszystkie elementy



Transformacja do relacji w łuku

krok 8. Przystąpić do transformacji

Wynik:



Generowanie bazy danych krok 2.

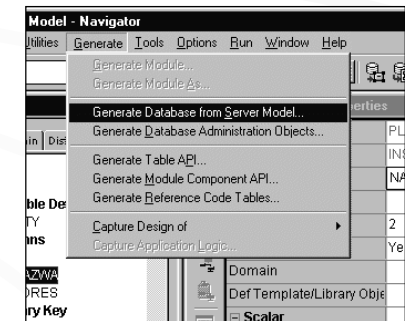
Generowanie schematu fizycznego bazy danych

Przebieg procesu



krok 1. Uruchomić narzędzie *Design Editor*. Przejść na zakładkę *Server Model*, rozwinąć gałąź systemu aplikacji

krok 2. Wybrać pozycję *Generate Database from Server Model* z menu *Generate*

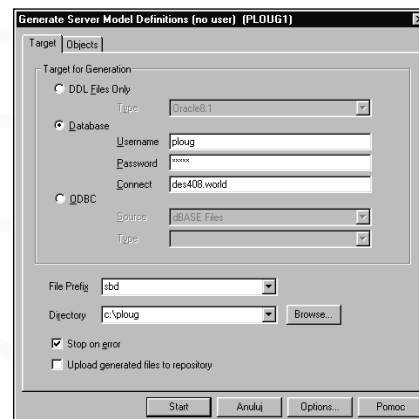


Przebieg procesu



krok 3. Ustalić parametry generacji - zakładka *Target*:

- Cel generacji:
 - skrypty *DDL* (różne formaty)
 - wskazany użytkownik bazy danych *Oracle*
 - baza danych *ODBC*
- Lokalizacja skryptów

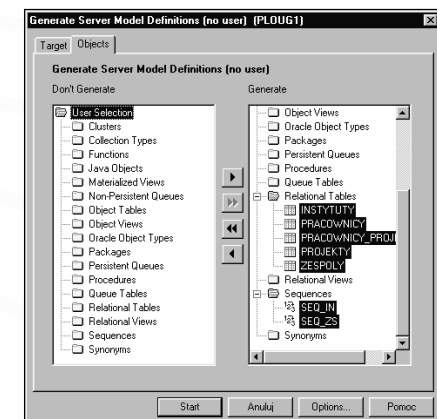


Przebieg procesu



krok 4. Wybrać obiekty do generacji - zakładka *Objects*:

- Typ obiektu:
 - relacje
 - liczniki
 - perspektywy i inne
- Konkretny obiekt





Przebieg procesu

krok 5. Uruchomić proces - przycisk *Start*

Wynik - w zależności od parametrów generacji:

- skrypty DDL we wskazanym katalogu
- obiekty w schemacie wskazanego użytkownika
- obiekty w bazie danych przyłączonej za pomocą *ODBC*