



Fundusze Europejskie
Wiedza Edukacja Rozwój



Politechnika Wroclawska

Unia Europejska
Europejski Fundusz Społeczny



„ZPR PWr – Zintegrowany Program Rozwoju Politechniki Wrocławskiej”

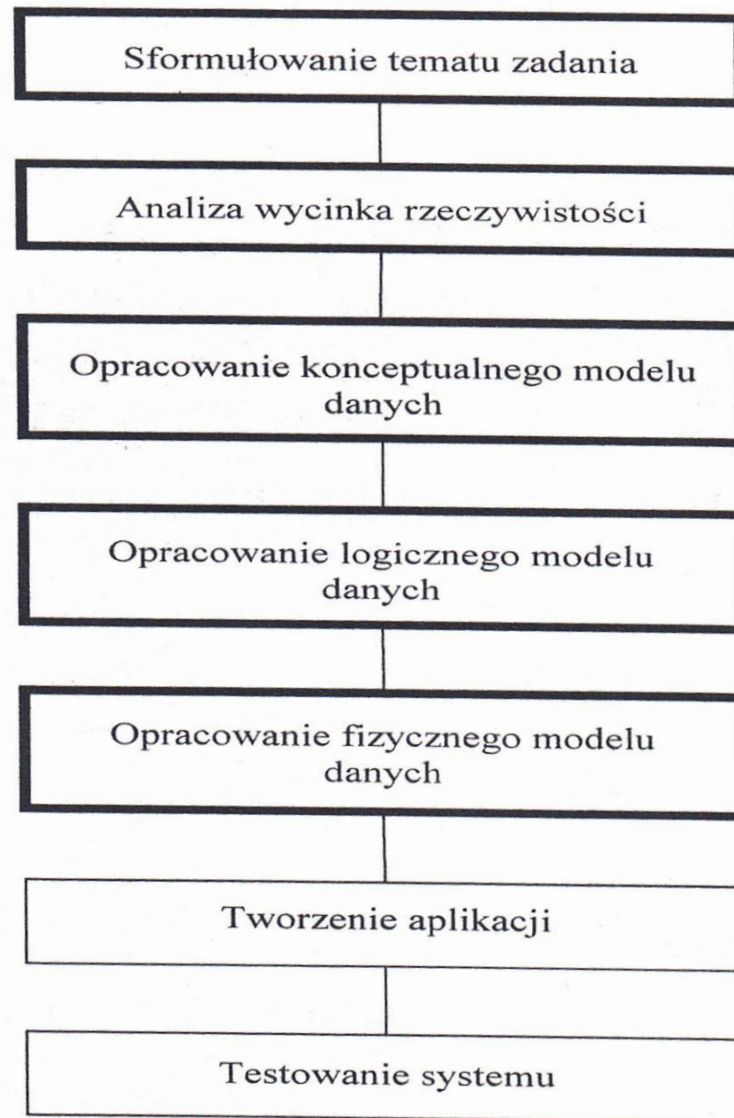
BAZY DANYCH

Wykład

Projektowanie relacyjnych baz danych

Dr hab. Zygmunt Mazur, prof. uczelni

Etapy projektowania bazy danych



Etapy projektowania bazy danych oraz tworzenia systemu bazy danych

Projektowanie relacyjnych baz danych

- Projektowanie relacyjnych baz danych ma na celu opracowanie poprawnego i spełniającego wymagania użytkowników logicznego schematu bazy danych.
- Opracowany projekt bazy danych musi być poprawny i spełniać oczekiwania przyszłych użytkowników.
- Przystępując do projektowania bazy należy precyzyjnie określić: w jakim **celu** projektujemy bazę, jej potencjalnych **użytkowników** oraz ich **oczekiwania, potrzeby, ograniczenia i wymagania**.
- Głównym celem projektowania relacyjnej bazy danych jest przedstawienie jej w postaci odpowiedniego zbioru schematów relacji.

Etapy projektowania relacyjnej bazy danych

W procesie projektowania relacyjnej bazy danych wyróżnia się opracowanie projektu:

- **Modelu konceptualnego** (CDM – ang. *Conceptual Data Model*)
- **Modelu logicznego** (LDM – ang. *Logical Data Model*)

Implementacja bazy danych związana jest z opracowaniem

- **Modelu fizycznego** (PDM – *Physical Data Model*)

Model danych

- **Model danych** (ang. *database model*) pozwala za pomocą ustalonego zestawu pojęć (metajęzyka) opisać określony wycinek świata rzeczywistego.
- W modelu danych wyróżnia się opis:
 - **struktury** danych w tym więzów integralności danych
 - **operacji** wykonywanych na danych

Metodyka strukturalna projektowania relacyjnych baz danych

- Poprawny projekt relacyjnej bazy danych można wykonać np.
 - przeprowadzając normalizację zaproponowanych schematów bazy danych
- lub
 - wykonując poprawnie etapy proponowanej autorskiej metodyki strukturalnej obejmującej projekt modelu conceptualnego i logicznego, które pozwolą na wykonanie modelu fizycznego, a następnie na implementację bazy danych w wybranym narzędziu.
- Projektowanie bazy danych w metodyce strukturalnej obejmuje wykonanie 13 określonych etapów i udokumentowanie ich w wybranym edytorze tekstów np. MS Word.

Dokumentacja projektu bazy danych

- Dokumentacja projektu bazy danych powinna zawierać szczegółowe opisy poszczególnych etapów zgodnie z ustaloną notacją, podanym szablonem, wymaganiami edytorskimi itp.
- Powinna być poprawna i czytelna.
- Zawartość poszczególnych rozdziałów powinna odpowiadać ich tytułom.
- W celu zapewnienia poprawności jako całości wymagane jest weryfikowanie zgodności wielu etapów i powrotów do etapów wcześniej wykonanych.
- Znalezienie błędów w późniejszych etapach często wymaga ponownego projektowania nawet od etapu 1.

Wytyczne do projektu bazy danych

- Dane w bazie danych mogą zawierać polskie litery, spacje, apostrofy i inne ustalone znaki.
- W nazwach kategorii, atrybutów, encji, związków i relacji nie należy używać polskich liter i znaków specjalnych.
- Nazwy powinny być jednoczłonowe, krótkie i czytelne.
- Nie należy nadawać nazw takich jak słowa kluczowe, czy nazwy systemowe używane w bazach danych, np.
zamiast nazw: *Nazwa*, *Klasa*, *Wiersz*, *Kolumna*, *Data*,
lepiej dać nazwę: *NazwaP*, *KlasaU*, *WierszX*, *KolumnaK*, *DataUr*
stosownie do okoliczności użycia.

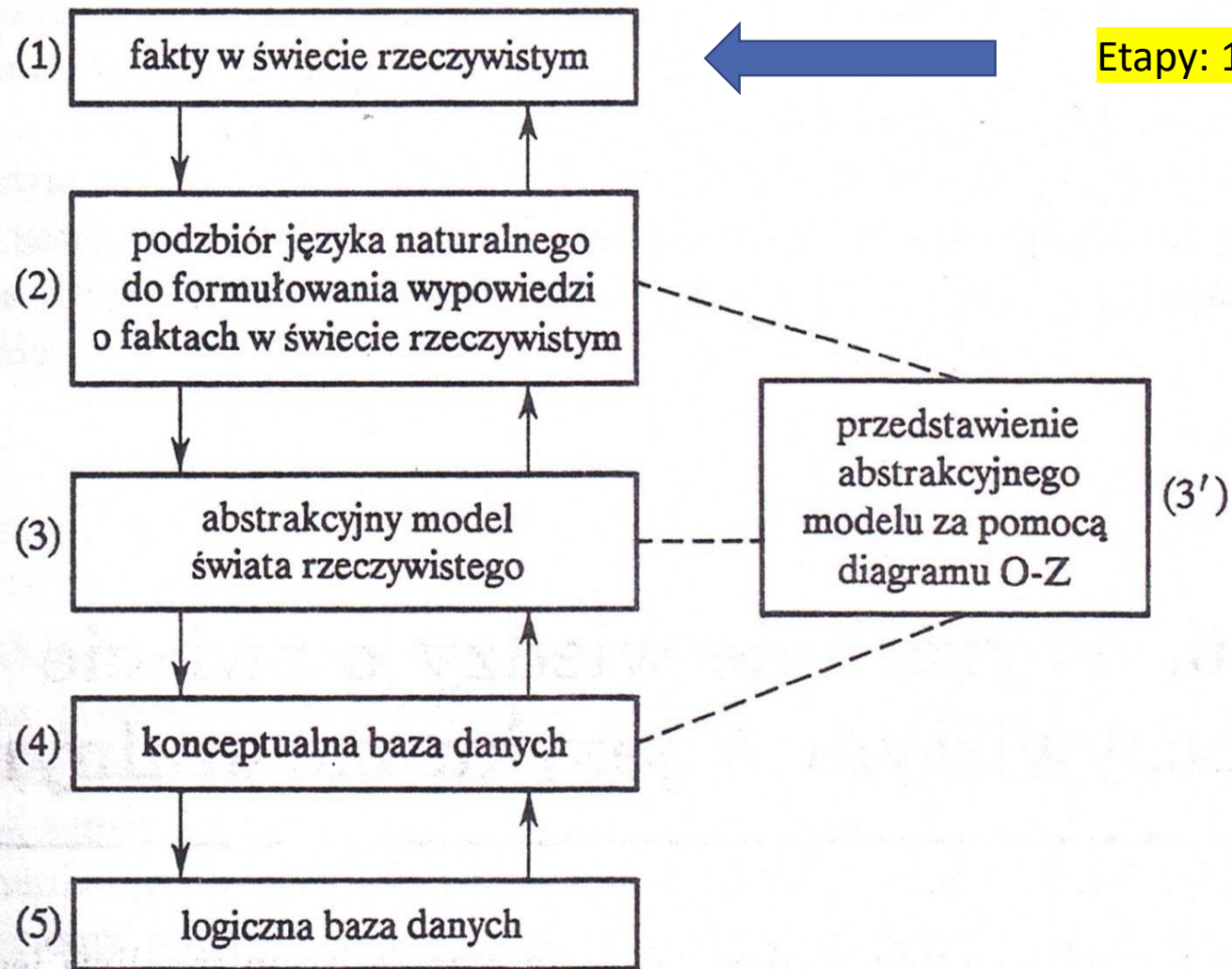
Etapy projektowania modelu konceptualnego

- Etap 1.** Zdefiniowanie tematu, celu, zakresu i użytkowników bazy danych
- Etap 2.** Analiza rzeczywistości
- Etap 3.** Zdefiniowanie kategorii.
- Etap 4.** Wyodrębnienie reguł funkcjonowania
- Etap 5.** Wyodrębnienie ograniczeń dziedzinowych.
- Etap 6.** Zdefiniowanie transakcji.
- Etap 7.** Identyfikację typów encji i związków.
- Etap 8.** Definicje predykatowe encji i związków.
- Etap 9.** Diagram związków encji.

Etapy projektowania modelu logicznego

- Etap 10.** Transformacja modelu conceptualnego do modelu logicznego.
- Etap 11.** Zdefiniowanie schematów relacji i podanie danych przykładowych.
- Etap 12.** Opracowanie schematu bazy danych i słownika atrybutów.
- Etap 13.** Zdefiniowanie perspektyw dla wszystkich użytkowników bazy.

Jak przedstawić semantykę świata w bazie danych



Schemat ustalania związku między danymi w bazie a faktami w świecie rzeczywistym

Opis Etapu 1

- Pierwszym i zasadniczym elementem projektowania bazy danych jest ustalenie tytułu (tematu) przedsięwzięcia, zwięzłe i czytelne sformułowanie, w jakim celu i zakresie będzie wykonywany projekt bazy oraz zidentyfikowanie wszystkich potencjalnych użytkowników bazy i operacje jakie będą wykonywali na danych, a w szczególności, czy operowanie danymi będzie wymagało ich identyfikacji (zalogowania).
- Poprawnie określony cel umożliwia przystąpienie do etapu 2 – analizy wycinka rzeczywistości.

Etap 1

1. Temat, cel, zakres i użytkownicy bazy danych

1.1. Temat

1.2. Cel

1.3. Zakres

1.4. Użytkownicy

Etap 1 – przykład

1. Temat, cel, zakres i użytkownicy bazy danych

1.1. Temat

Katastrofy budowlane.

1.2. Cel

Celem przedsięwzięcia jest zaprojektowanie bazy danych umożliwiającej gromadzenie i przetwarzanie danych o katastrofach budowlanych przez osobę z Ministerstwa Infrastruktury.

1.3. Zakres

Baza danych powinna umożliwiać gromadzenie, wyszukiwanie, sortowanie i czytelne prezentowanie danych katastrof budowlanych w Polsce oraz tworzenie raportów i statystyk.

1.4. Użytkownicy

- Specjalista – po zalogowaniu ma możliwość wprowadzania, edycji i usuwania danych,
- Internauta – przeglądanie danych (nie wymaga logowania).

Etap 2

2. Analiza wycinka rzeczywistości

2.1. Szczegółowy opis wycinka rzeczywistości

2.2. Słownik pojęć

2.3. Użytkownicy i zakres uprawnień

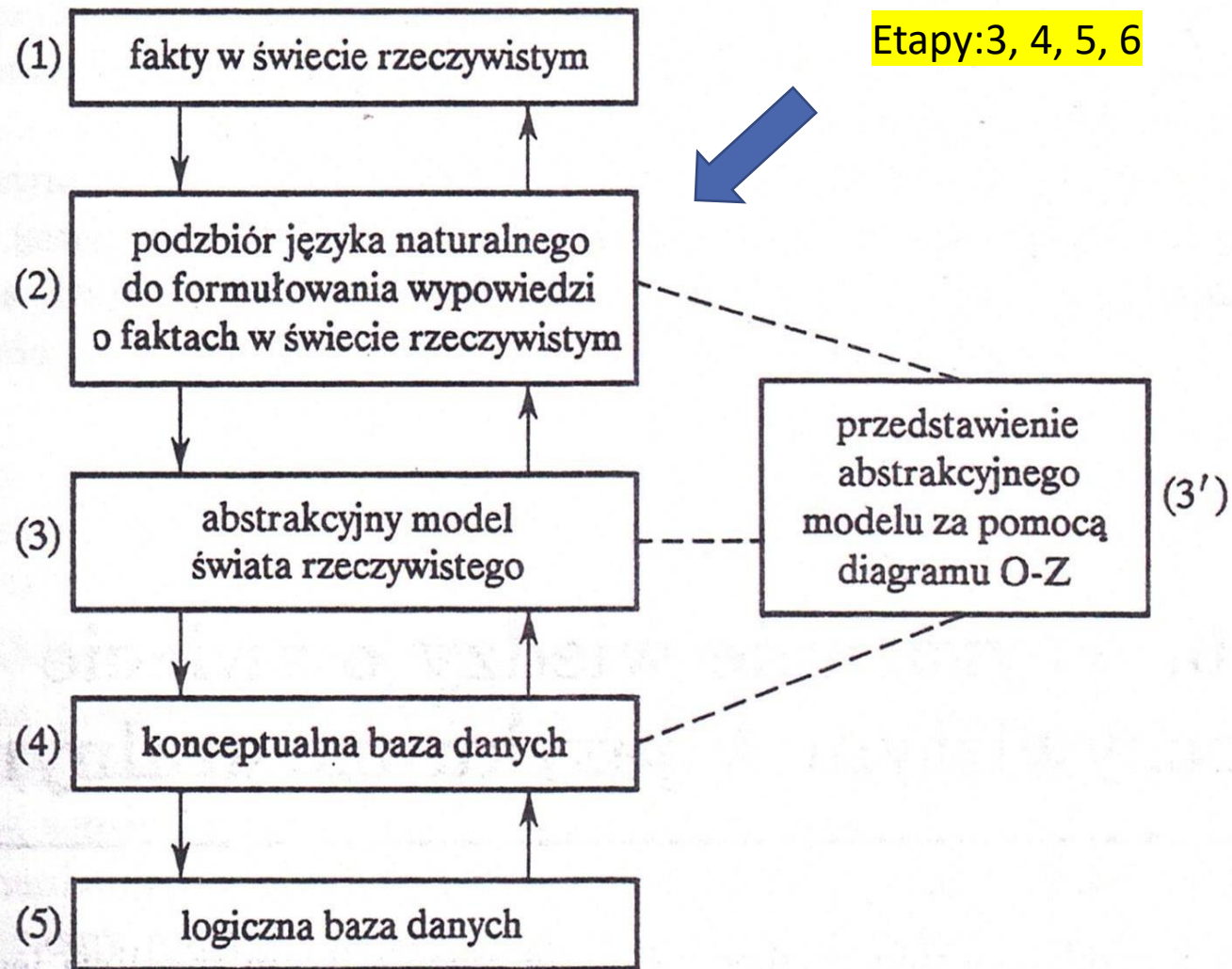
2.4. Wymagania funkcjonalne

2.5. Wymagania нефunkcjonalne

2.6. Analiza istniejących baz danych

2.7. Analiza kosztów

Jak przedstawić semantykę świata w bazie danych



Schemat ustalania związku między danymi w bazie a faktami w świecie rzeczywistym

Opis Etapu 3 (1)

- Ze szczegółowej analizy rzeczywistości, na podstawie słownika pojęć, należy wyodrębnić rzeczy istotne, których dane chcemy przechowywać, np. towar, uczeń, kaseta, lek, faktura, zwane kategoriami (obiektami, typami danych).
- Dla każdej kategorii wyodrębniamy atrybuty, czyli ich cechy charakterystyczne, a wśród nich powinien być atrybut jednoznacznie identyfikujący każdą daną.
- Jeśli wśród wyodrębnionych atrybutów nie ma atrybutu jednoznacznie identyfikującego dane, to należy sztucznie wprowadzić atrybut identyfikujący każdą daną, np. symbol, numer, identyfikator.
- Atrybuty chronione nie powinny być używane do identyfikowania danych, np. PESEL czy login.
- Jeśli atrybut email ma także pełnić rolę loginu użytkownika, to taka informacja musi być zawarta w opisie atrybutu.

Opis Etapu 3 (2)

- Nazwy kategorii są rzeczownikami w liczbie pojedynczej, nie powinny zawierać polskich liter ani znaków specjalnych, powinny być krótkie i czytelne.
- Z opisu kategorii powinny wynikać powiązania z innymi kategoriami, np. jeśli potrzebne są dane o dostawcach i dostarczanych przez nich towarach, to kategoria **Towar** nie powinna mieć atrybutu *Dostawca* (**Dostawca** powinien być odrębną kategorią), a w opisie powinna być informacja o związku towaru z dostawcą.
- Należy zwrócić uwagę, by wśród kategorii i ich atrybutów były wszystkie potrzebne do gromadzenia danych, a w szczególności np. danych użytkowników, loginów, haseł itp.
- Nie należy wprowadzać atrybutów nie istotnych, zbędnych (na wszelki wypadek).
- Dla każdego atrybutu należy podać co najmniej jedną wartość przykładową.

Etap 3 – Definicje kategorii

KAT/xxx **Nazwa kategorii** (xxx kolejny trzycyfrowy numer kategorii)

Opis: Słowny opis kategorii

Atrybuty: Wykaz atrybutów i ich opisy

Etap 3 – przykład

KAT/001 Towar

Opis: Kategoria **Towar** służy do opisu towaru przechowywanego w magazynie, który może być dostarczany przez różnych dostawców.

Atrybuty:

- | | |
|------------------|---|
| <i>IdT</i> | – identyfikator towaru, np. 1, 2, 3 |
| <i>NazwaT</i> | – nazwa grupy towarów, np. cukier biały |
| <i>SymbolT</i> | – unikalny symbol towaru, np. cuk001/99 |
| <i>Jednostka</i> | – jednostka miary towaru, np. kg, km, szt. |
| <i>Uwaga</i> | – dowolne uwaga, np. towar sprzedawany od 1.1.2000 r. |

Opis Etapu 4

- Na podstawie sporządzonej wcześniej analizy rzeczywistości należy wyodrębnić **reguły funkcjonowania** (reguły biznesowe).
- Prawidłowe i wyczerpujące określenie reguł funkcjonowania oraz ich przestrzeganie we wszystkich kolejnych etapach, jest warunkiem koniecznym do poprawnego zaprojektowania bazy danych.
- Reguły należy wypisywać w sposób systematyczny – można pogrupować je z podziałem na kategorie, których dotyczą oraz zdefiniować reguły ogólne.
- Reguły powinny być formułowane jednoznacznie i precyzyjnie, ponieważ służą do definiowania związków w kolejnych etapach.
- Wśród reguł powinny być zarówno reguły dotyczące funkcjonowania (realizowanych procesów biznesowych) jak i dotyczące uprawnień użytkowników.

Etap 4

4. Reguły funkcjonowania

4.1. Reguły ogólne

REG/001 Tekst

REG/002 Tekst

4.2. Reguły dotyczące **KAT/001** Nazwa kategorii

REG/003 Tekst

REG/004 Tekst

4.3. Reguły dotyczące **KAT/002** Nazwa kategorii

REG/005 Tekst

REG/006 Tekst

Etap 4 – przykład

- REG/001** Dane towarów wpisuje kierownik magazynu.
- REG/002** Każdy towar musi mieć przypisanego dostawcę.
- REG/003** Każdy towar może mieć co najwyżej jednego dostawcę.
- REG/004** Dostawca nie musi dostarczać towary.
- REG/005** Dostawca może dostarczać wiele towarów.
- REG/006** Faktury są przechowywane przez 5 lat.

Opis Etapu 5

- Na podstawie sporządzonej wcześniej analizy rzeczywistości i zdefiniowanych kategorii należy wyodrębnić **ograniczenia dziedzinowe**.
- Ograniczenia dziedzinowe dotyczą atrybutów kategorii: ich typów, wartości, obligatoryjności/opcjonalności, unikalności itp.
- Ograniczenia dziedzinowe należy wypisywać w sposób systematyczny – można pogrupować je z podziałem na kategorie, których dotyczą oraz zdefiniować ograniczenia ogólne.
- Jeśli w opisie ograniczeń są używane jakieś oznaczenia, to należy je wcześniej lub w danym ograniczeniu zdefiniować.

Etap 5

5. Ograniczenia dziedzinowe

5.1. Ograniczenia ogólne

OGR/001 Tekst

OGR/002 Tekst

5.2. Reguły dotyczące **KAT/001** Nazwa kategorii

OGR/003 Tekst

OGR004 Tekst

5.3. Reguły dotyczące **KAT/002** Nazwa kategorii

OGR/005 Tekst

OGR/006 Tekst

Etap 5 – przykład

- OGR/001** Nazwiska i imiona są ciągiem maksymalnie 30 znaków, mogą to być litery, myślnik, apostrof ' , np. O'Neil, Abacka-Nowak, Lis.
- OGR/002** Nazwisko, imię jest atrybutem obligatoryjnym.
- OGR/003** Każdy towar ma unikalny symbol nadany przez kierownika magazynu.
- OGR/004** Wszystkie atrybuty dostawcy są obligatoryjne.
- OGR/005** Kod pocztowy ma postać 99-999, gdzie 9 oznacza dowolną cyfrę.
- OGR/006** Data urodzenia jest wcześniejsza niż data zatrudnienia.
- OGR/007** Liczba uczniów w klasie jest liczbą całkowitą $\in [0, 35]$.

Transakcje

- **Transakcje** są to operacje wykonywane na danych.
- Wynikają one z przeprowadzonej analizy i są opisywane przy pomocy pojęć języka naturalnego używanych w opisie rzeczywistości.
- W przypadku zerwania transakcji (ang. *abort*) jest przywracany stan bazy sprzed rozpoczęcia transakcji na podstawie dziennika transakcji (plik o nazwie *log*), w którym są zapisywane poszczególne stany bazy danych przed i po wykonaniu każdej transakcji. Dziennik taki jest czytany od ostatniego zapisu do początku i przywracany jest stan bazy (ang. *rollback*).

Własności transakcji

Transakcje powinny mieć cztery podstawowe własności w skrócie nazywane *własnościami* **ACID** (**A**tomicity, **C**onsistency, **I**solation, **D**urability):

- **niepodzielność, atomowość** (*ang. **atomicity***) – transakcja jest wykonywana w całości albo wcale. Po przerwaniu transakcji musi być odtworzony stan bazy danych sprzed rozpoczęcia transakcji,
- **spójność** (*ang. **consistency***) – transakcje zachowują spójność bazy danych, chociaż w trakcie działania transakcji stan bazy może być chwilowo niespójny,
- **izolacja** (*ang. **isolation***) – transakcje są całkowicie od siebie niezależne, jedna transakcja od drugiej jest odizolowana,
- **trwałość** (*ang. **durability***) – zmiany dokonane przez pomyślnie zakończoną transakcją są zachowywane na trwałe.

Opis Etapu 6

- Etap wyodrębniania transakcji jest w zasadzie uszczegółowieniem etapu analizy wymagań funkcjonalnych. Opisując poszczególne transakcje należy określić, które kategorie biorą udział w transakcji, co jest źródłem danych wejściowych oraz co generowane jest na wyjściu.
- W określeniu, skąd pochodzą dane wejściowe i gdzie kierowane są wyniki transakcji, można użyć oznaczeń: U – użytkownik, BD – baza danych.
- Przez użytkownika rozumiemy tutaj zarówno osobę korzystającą z bazy danych jak i system obsługujący bazę danych.
- W etapie tym należy wymienić wszystkie przewidywane transakcje dla wszystkich grup użytkowników.

Etap 6 – Transakcje

6. Transakcje

TRA/xxx Nazwa transakcji

Opis: Słowny opis transakcji

Uwarunkowania: Specyfikacja warunków dla prawidłowego wykonania transakcji oraz opis ścieżki alternatywnej, do której może dojść, jeśli warunki te nie będą spełnione.

Wejście/Wyjście: Opis wejścia i wyjścia transakcji lub tabela transakcji gdzie xxx jest kolejnym numerem transakcji.

Etap 6 – przykład transakcji edycji danych

TRA/001 Edycja danych towaru

Opis: Zadaniem transakcji jest wyszukanie danych o wybranym towarze i edycja tych danych np. ceny sprzedaży. Edycję może wykonać tylko kierownik magazynu.

Uwarunkowania: Dane towaru muszą istnieć w bazie. Jeśli dane nie zostaną znalezione, zostanie wyświetlony komunikat *Brak danych*. Wyszukanie danych towaru może być wykonane po podaniu jego nazwy lub symbolu. Po wykonaniu edycji danych towaru, użytkownik otrzymuje komunikat *Edycja zakończona*.

	WEJŚCIE	WYJŚCIE
Użytkownik	Dane identyfikujące towar Nowe dane towaru	Komunikat
Baza danych	Dane towarów	Dane towarów

Etap 6 – przykład transakcji usuwania danych

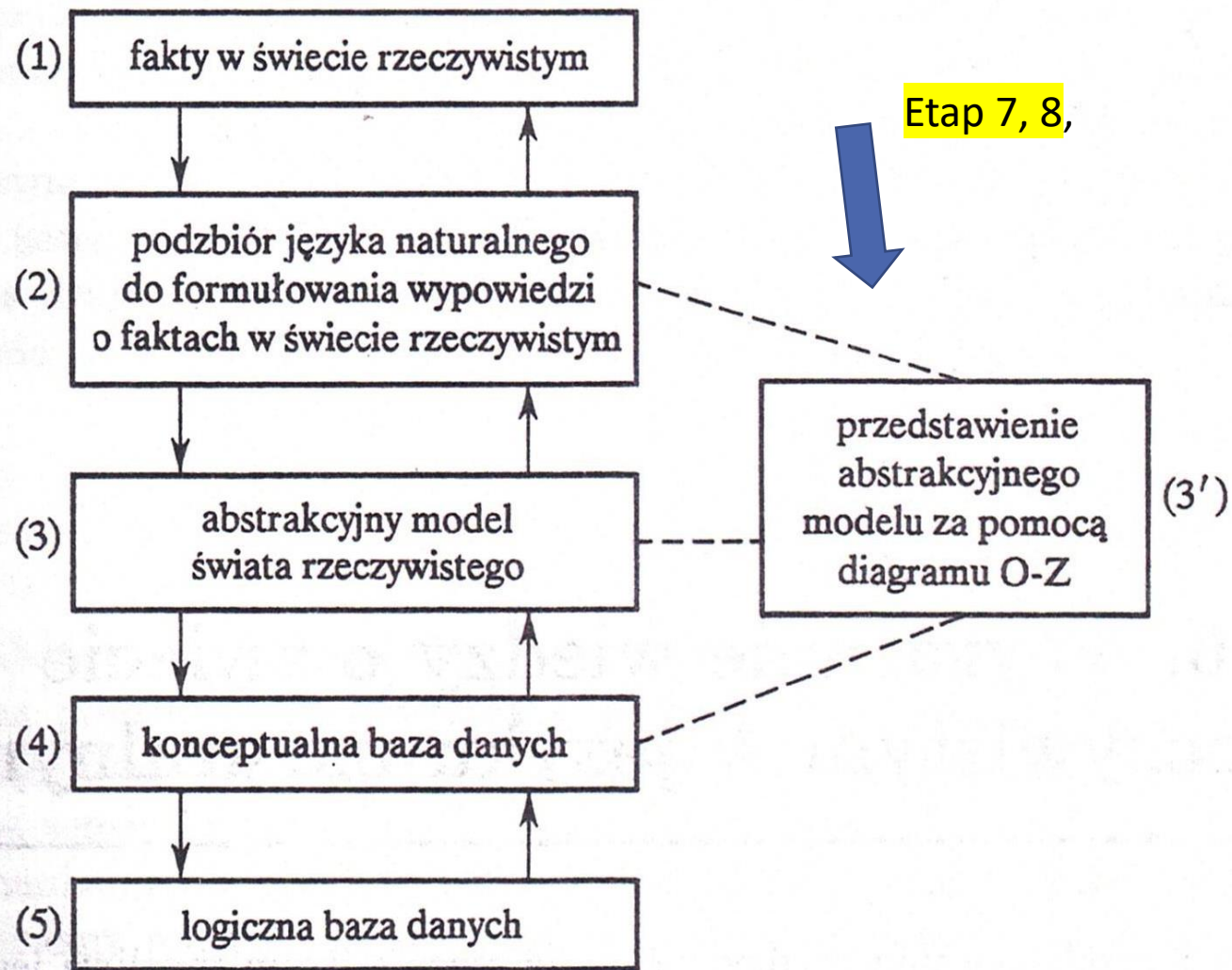
TRA/002 Usuwanie danych towaru

Opis: Zadaniem transakcji jest wyszukanie towaru i usunięcie jego danych. Usunięcie może wykonać tylko kierownik magazynu. Usuwanie danych jest ograniczone, tzn. jeśli dany towar jest powiązany chociaż z jedną fakturą sprzedaży, to dane nie są usuwane i jest wyświetlany komunikat *Dane towaru nie można usunąć*.

Uwarunkowania: Dane towaru muszą istnieć w bazie. Jeśli dane towaru nie zostaną znalezione, zostanie wyświetlony komunikat *Brak danych*. Wyszukanie danych towaru może być wykonane po podaniu jego nazwy lub symbolu. Po usunięcia danych towaru, użytkownik otrzymuje komunikat *Dane towaru zostały usunięte*.

	WEJŚCIE	WYJŚCIE
Użytkownik	Dane identyfikujące towar Nowe dane towaru	Komunikat
Baza danych	Dane towarów, Dane faktur	Dane towarów

Jak przedstawić semantykę świata w bazie danych



Schemat ustalania związku między danymi w bazie a faktami w świecie rzeczywistym

Model konceptualny

- Opracowanie modelu konceptualnego (konceptyjnego, pojęciowego, ang. *Conceptual Data Model – CDM*) polega na wyodrębnieniu i zdefiniowaniu **encji** oraz **związków** między nimi.
- Model danych jest pewnym uproszczonym opisem, zgodnym z modelowanym wycinkiem rzeczywistości.
- Celem modelowania konceptualnego jest zapewnienia wyższego poziomu abstrakcji widzenia danych z perspektywy użytkownika oraz niezależności danych od implementacji (aplikacji).
- Aby zapewnić poprawność modelu danych w sensie zgodności, często podaje się *metamodel danych* czyli model modelu. Metamodel obejmuje definicje pojęć (np. encji, atrybutu) i reguły dotyczące pojęć wykorzystywanych w modelu, np. jeśli model zawiera co najmniej dwie encje, to każda encja musi być co najmniej w jednym związku z dowolną inną encją.

Identyfikacja encji

- **Encja** (ang. *entity*) –nobiekt, rzecz, zjawisko, zdarzenie, byt.
- Encje posiadają atrybuty, czyli cechy je opisujące (charakteryzujące), istotne z punktu widzenia eksperta dziedzinowego, przyjmujące wartości z ustalonego zbioru – **dziedziny**.
- Encjom nadajemy nazwy, które powinny być rzeczownikami w liczbie pojedynczej.
- Nazwy encji piszemy wielkimi literami.
- Na diagramie związków encji encję przedstawiamy w postaci prostokąta rysowanego linią pojedynczą (encja silna) lub podwójną (encja słaba).

Encja silna i słaba

- **Encja silna** (regularna, właściwa) oznacza obiekt, który jest niezależny od innych obiektów. Reprezentowana jest przez prostokąt narysowany linią pojedynczą, ma nazwę, która jest rzeczownikiem w liczbie pojedynczej.

ENCJA SILNA

- **Encja słaba** (ang. *weak entity*) oznacza obiekt, który może istnieć dopiero po zdefiniowaniu innych obiektów, np. dane dzieci pracownika można wprowadzać dopiero wtedy, gdy istnieją dane odpowiedniego pracownika. W tym wypadku encja DZIECKO byłaby encją słabą, a encja PRACOWNIK encją silną. Encja słaba reprezentowana jest przez prostokąt narysowany linią podwójną, gdyż na ogół na nią trzeba zwrócić szczególną uwagę. Nazwa encji jest rzeczownikiem w liczbie pojedynczej.

ENCJA SŁABA

Atrybut

- **Atrybut** (ang. *attribute*) – cecha charakterystyczna encji lub związku, element służący do opisu, określania ilości lub wyrażenia stanu encji lub związku.
- Atrybut określa dokładnie jedną wartość, która z punktu widzenia projektanta jest atomowa (niepodzielna).
- Nazwa atrybutu powinna być rzeczownikiem w liczbie pojedynczej, jeśli występuje w liczbie mnogiej to jest to sygnał, że powinniśmy utworzyć nową encję, np. *Imiona dzieci*, *Znane języki obce*.

Atrybut opcjonalny i obligatoryjny, wartość NULL

- **Atrybut opcjonalny** (ang. *optional attribute*) – atrybut nieobowiązkowy, jego wartość może być nieokreślona, pusta, nieznana (oznaczana jako NULL).
- **Atrybut obligatoryjny** (ang. *mandatory attribute*) – atrybut wymagany, jego wartość musi być zawsze określona. Nie może przyjmować wartości NULL.
- **Wartość NULL** – wartość pusta, różnie interpretowana, np. jako wartość nieznana, nieokreślona, niepewna, zastrzeżona. Dwa atrybuty o wartości NULL nie są traktowane jako sobie równe bo mogą być różnie interpretowane, np. wartość NULL jako wartość atrybutu *NIP* może oznaczać jego brak lub że jest nieznany albo, że jest daną chronioną.

Typ encji

- Encje o takich samych cechach charakterystycznych możemy pogrupować w typy encji. Zwróćmy uwagę, że encja to nie to samo co typ encji. **Typ encji** jest kategorią a encja jest instancją (wystąpieniem) danego typu encji [Beynon2000]. Typem encji jest np. OSOBA o atrybutach (*Nazwisko, Imię*), a encją jest jeden obiekt danego typu np. („Jan”, „Kowalski”). Powszechnie używa się terminu encja zarówno na określenie typu encji jak i samej encji. Tam, gdzie nie prowadzi to do nieporozumień, będziemy używać pojęcia encja zamiast typ encji.
- Wyodrębnione typy encji należy wypisać porządkując je np. alfabetycznie według nazw lub w inny uporządkowany logicznie sposób.
- Definiując atrybuty typu encji powinno się określić nazwy dla atrybutów, podać opis (znaczenie) każdego atrybutu, jego typ (format, dziedzinę), przykładowe wartości lub zakresy wartości, określić, czy jest jednoznaczny identyfikatorem lub wchodzi w skład jednoznacznego identyfikatora, czy jest atrybutem obligatoryjnym (wymaganym) czy opcjonalnym.

Poprawność typów encji

- Każdy atrybut powinien przyjmować tylko jedną wartość w danej chwili (w danym stanie bazy).
- Jeśli, w celu zapewnienia trwałości bazy danych (historii danych) istnieje potrzeba przechowywania kolejnych zmieniających się wartości atrybutu (np. stanowisko pracownika, cena towaru, adres zamieszkania), to należy poprawić projekt definiując nowe typy encji, nowe atrybuty.
- W celu zweryfikowania, czy w projektowanej bazie danych nie będzie zbędnych (nigdy nie wykorzystywanych) danych, należy sprawdzić, czy każda encja będzie uczestniczyła w co najmniej jednej operacji bazodanowej.

Etap 7 – Definicje typów encji i związków

Etap 7 obejmuje dwa podetapy:

Etap 7.1 – Definicje typów encji

Etap 7.2 – Definicje typów związków

Etap 7.1 – Definicje typów encji

ENC/xxx NAZWA TYPU ENCJI (*xxx – kolejny trzycyfrowy numer typu encji*)

Semantyka encji – opis encji danego typu,

Wykaz atrybutów – lista atrybutów, ich opis i dziedzina (typ),
przedstawiona w formie tabeli atrybutów,

Klucze kandydujące – min. zbiory atrybutów jednoznacznie identyfikujących encje,

Klucz główny – jeden z kluczy kandydujących wybrany jako główny,

Charakter encji – encja silna lub słaba.

Klucze typów encji

Często na tym etapie wprowadza się **sztuczne klucze typów encji**, takie jak identyfikator encji (symbol, numer), jeśli:

- brak jest klucza kandydującego,
- klucze kandydujące są złożone z wielu atrybutów,
- klucze kandydujące są proste (jeden atrybut), ale są to np. atrybuty chronione, atrybuty o wartościach znakowych o dużym rozmiarze (dużej szerokości pola) lub liczby o wielu cyfrach znaczących.

Dane osobowe i dane chronione nie powinny być kluczami głównymi np. PESEL, login, numer dowodu osobistego lub paszportu, numer konta w banku, szczególnie, gdy nie ma całkowitej pewności co do unikalności ich wartości wynikających z błędnego tworzenia numerów lub z błędnego wprowadzenia danych.

Definicja typu encji – przykład

ENC/001 PRZEDMIOT

Semantyka encji – encja zawiera dane o przedmiocie nauczonym w szkole

Wykaz atrybutów:

Nazwa atrybutu	Opis atrybutu	Typ	OBL (+) OPC (–)
<i>NrP</i>	Numer porządkowy	Liczba całkowita dodatnia (< 100)	+
<i>NazwaP</i>	Nazwa przedmiotu nauczanego w szkole	Ciąg max. 30 znaków	+

Klucze kandydujące: NrP, NazwaP

Klucz główny: NrP

Charakter encji: encja silna

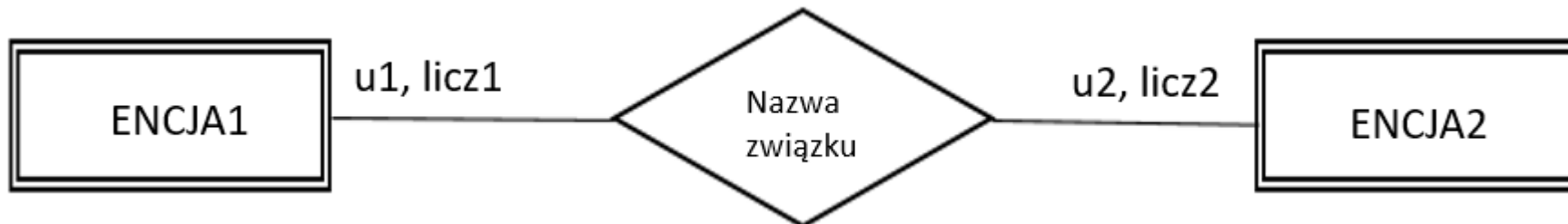
Związki (typy związków) pomiędzy encjami

- **Związek** (ang. *relationship*) jest to powiązanie (ang. *association*) pomiędzy encjami (najczęściej dwiema).
- n –argumentowy związek **Z** można zapisać jako relację o n argumentach (encjach E_1, \dots, E_n):

$$Z(E_1, \dots, E_n)$$

co oznacza, że encje E_1, \dots, E_n uczestniczą w związku **Z**.

- Graficznie związki (typy związków) przedstawiamy jako romby zawierające nazwę związku.



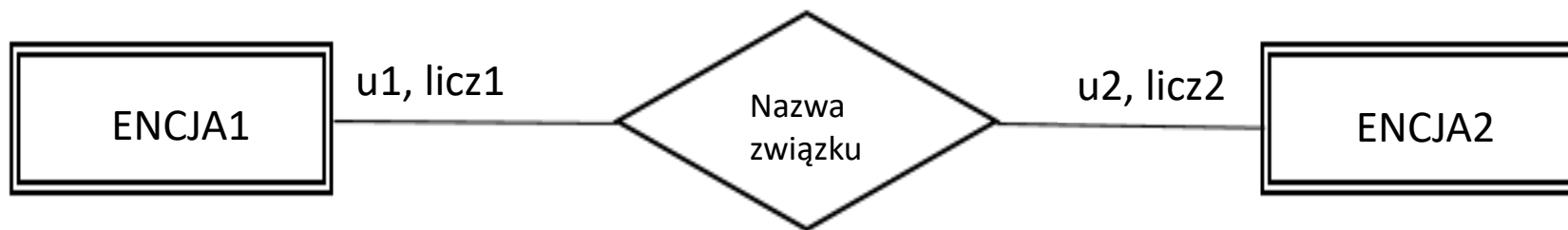
Stopień, liczność i typ uczestnictwa

- Opisując związki używamy pojęć: **stopień**, **liczność** i **typ uczestnictwa**.
- **Stopień związku** (ang. *degree*) – określa liczbę encji uczestniczących w związku, np. stopień związku binarnego wynosi 2.
- **Liczność** (ang. *cardinality*) związku określamy dla każdego końca związku jako:
 - **jeden** (co zapisujemy jako 1), tzn. w związku może uczestniczyć co najwyżej jedna encja danego typu
 - **wiele** (co zapisujemy jako N), tzn. w związku może uczestniczyć zero lub więcej encji
- **Uczestnictwo** encji w związku może być:
 - **opcjonalne** (ang. *optional*) – oznaczamy jako 0, nie każda encja musi uczestniczyć w związku czyli, mogą istnieć encje danego typu nie uczestniczące w związku
 - **obligatoryjne** (ang. *obligatory*) – oznaczamy jako 1, wszystkie encje tego typu biorą udział w związku.

Etap 7.2 – Definicje typów związków

ZWI/xxx Nazwa związku(ENCJA1(u1, licz1) : ENCJA2(u2, licz2); lista)

gdzie: xxx – kolejny numer związku,
u1, u2 – typ uczestnictwa encji w związku: 0 – opcjonalny, 1 – obligatoryjny,
licz1, licz2 – liczność encji w związku (1 lub N),
lista – lista atrybutów związku.



REG/xxx *Tekst reguły1*

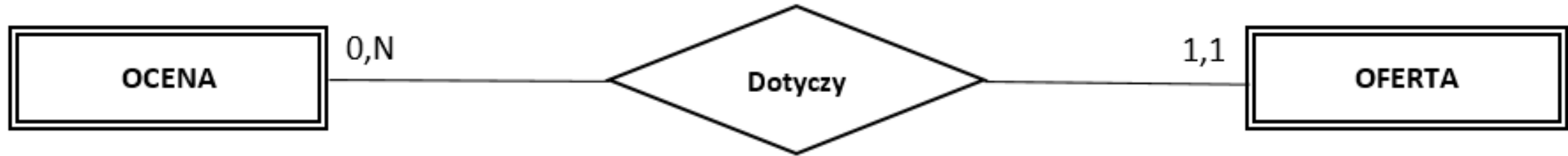
REG/xxx *Tekst reguły2*

REG/xxx *Tekst reguły3*

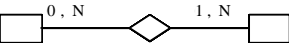
REG/xxx *Tekst reguły4*

Definicja typu związku – przykład

ZWI/001 Dotyczy(OCENA(0,N) : OFERTA(1,1))



- | | |
|---------|---|
| REG/001 | Ocena musi dotyczyć oferty |
| REG/002 | Ocena może dotyczyć maksymalnie jednej oferty |
| REG/003 | Oferta nie musi posiadać oceny |
| REG/004 | Oferta może posiadać wiele ocen |



Typy związków

Związki występujące często	Związki występujące rzadko	Związki niemożliwe



Uwaga

**Przy projektowaniu bazy danych zaleca się związki typu N:N
zastępować związkami typu 1:N**

Etap 8 – Definicje predykatowe typów encji i związków

- Definicje predykatowe typów encji przedstawiają w zwarty i czytelny sposób typy encji zdefiniowane w etapie 7.1.
- Definicje predykatowe typów związków przedstawiają w zwarty i czytelny sposób typy związków zdefiniowane w etapie 7.2.
- Definicje predykatowe typów encji i związków ułatwiają opracowanie etapu 9 – diagramu związków encji.

Etap 8.1 – Definicje predykatowe typów encji

Definicje predykatowe typów encji przedstawiamy w następujący sposób:

ENC/xxx NAZWA TYPU ENCJI (lista atrybutów) xxx – kolejny numer typu encji

- Nazwy typów encji podajemy w kolejności zgodnej z definicjami kategorii i encji we wcześniejszych etapach.
- W nawiasach podajemy nazwy atrybutów.
- Klucz główny typu encji zaznaczamy przez podkreślenie i umieszczamy na początku listy atrybutów.

Etap 8.2 – Definicje predykatowe typów związków

Definicje predykatowe typów związków przedstawiamy w następujący sposób:

ZWI/xxx Nazwa związku(ENCJA1(u1, licz1) : ENCJA2(u2, licz2); lista)

gdzie: xxx – kolejny numer związku,

u1, u2 – typ uczestnictwa encji w związku: 0 – opcjonalny, 1 – obligatoryjny,

licz1, licz2 – liczność encji w związku (1 lub N),

lista – lista atrybutów związku.

Etap 8 – Definicje predykatowe typów encji i związków – przykład

Etap 8.1. Definicje predykatowe typów encji

ENC/001 PRODUKT(IdP, NazwaP, DataDodania, Opis)

ENC/002 SKLEP(IdS, NazwaS, Adres, DataDodaniaS, OpisS)

ENC/003 OFERTA (IdO, Cena, Ilosc, Jednostka, DataAktualizacji, DataWygasniecia)

ENC/004 KATEGORIA(IdK, NazwaK)

ENC/005 OCENA(IdOc, Ocena, DataDodaniaOc, Komentarz)

Etap 8.2. Definicje predykatowe typów związków

ZWI/001 Ma(PRODUKT(0,N) : KATEGORIA(1,1))

ZWI/002 Dotyczy(OCENA(0,N) : OFERTA(1,1))

ZWI/003 Obejmuje(OFERTA(O,N) : PRODUKT(1,1))

ZWI/004 Oferuje(SKLEP(1,1) : OFERTA(0,N))

Etap 9 – Diagram związków encji (ERD)

Po wyodrębnieniu encji i związków można opracować ERD
diagram związków encji (diagram obiektowo-związkowy)
(ang *Entity Relationship Diagram*)

ERD – graficzne przedstawienie conceptualnego modelu danych za pomocą ustalonej notacji np. Chena, Martina, Yourdona, UML.

Diagram związków encji – przykład

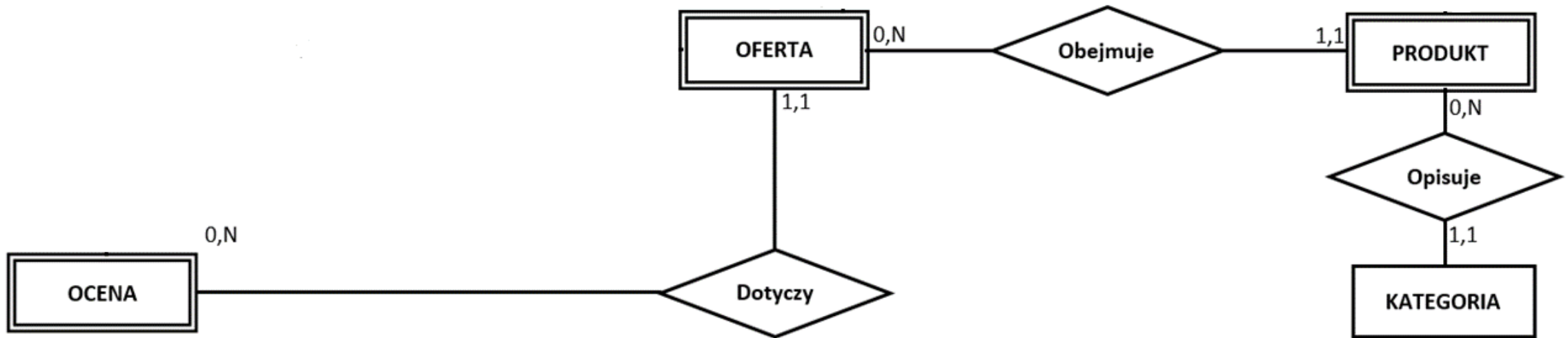
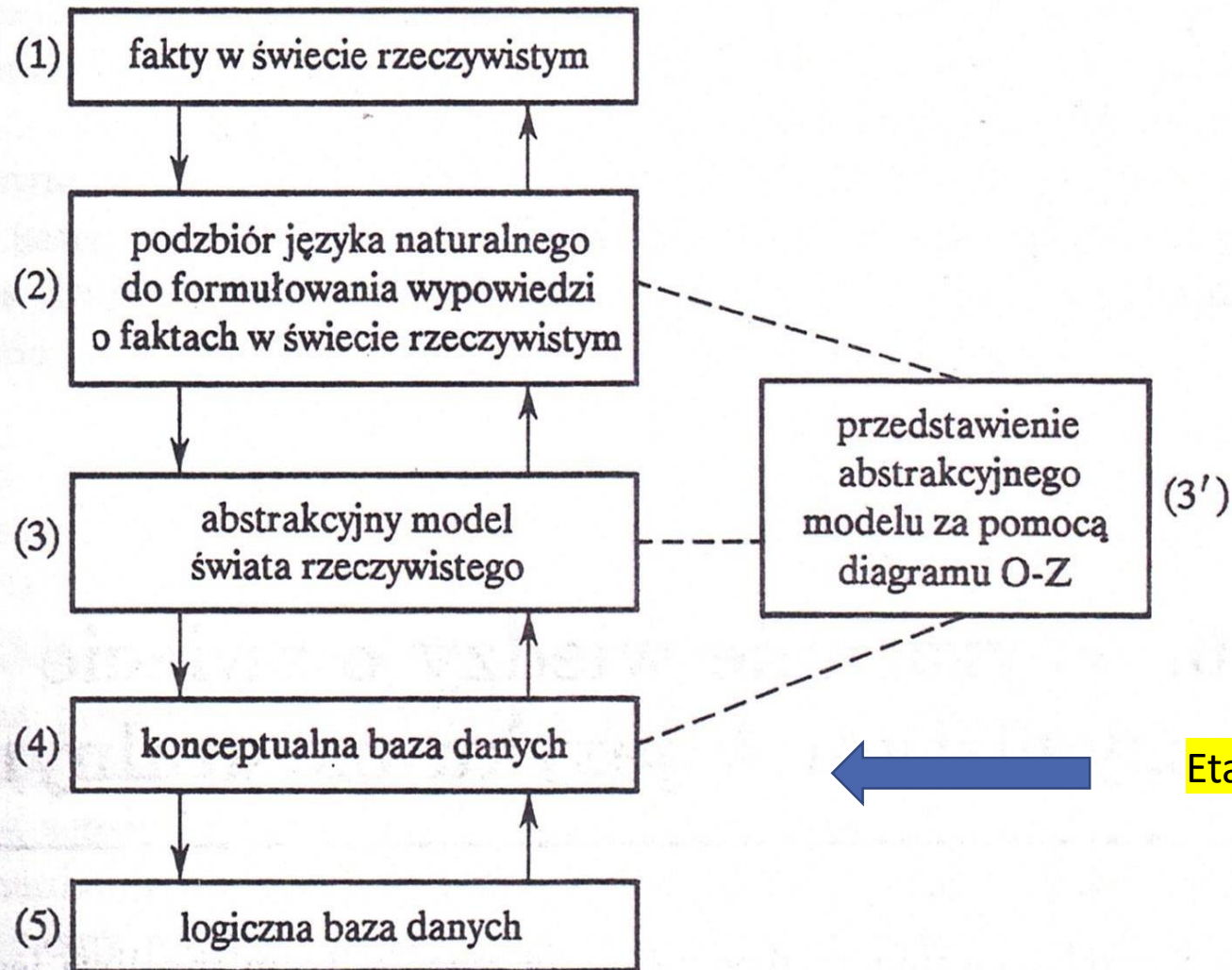




Diagram związków encji (ERD)

Opracowanie diagramu ERD kończy etap
modelowania konceptualnego bazy danych

Jak przedstawić semantykę świata w bazie danych



Schemat ustalania związku między danymi w bazie a faktami w świecie rzeczywistym

Etap 10 – Transformacja ERD do modelu logicznego

- Wykorzystując reguły transformacji przekształcamy modelu konceptualnego do relacyjnego modelu logicznego będącego zbiorem struktur danych reprezentowanych przez relacje.
- Reprezentacją relacji o zadanym schemacie jest tabela.
- Dla poprawnie zdefiniowanych encji i związków między nimi, na ogół otrzymujemy schematy relacji w trzeciej postaci normalnej.

Reguły transformacji (1)

- Dla każdej encji z ERD tworzymy schemat relacji.
- Najczęściej nazwa schematu relacji (i relacji) jest taka sama jak encji tylko w liczbie mnogiej ze względu na to, że relacja zawiera wiele wystąpień obiektu.
- Atrybuty encji stają się atrybutami w schemacie relacji (o takich samych nazwach jak odpowiadające im atrybuty encji).
- Atrybuty odpowiadające kluczom głównym encji stają się kluczami głównymi relacji.
- Atrybuty opcjonalne są atrybutami o dopuszczalnych wartościach NULL, atrybuty obligatoryjne (wymagane) – NOT NULL.
- Dla każdego związku binarnego 1:N (*jeden do wiele*), obligatoryjnego po stronie *jeden*, wstawiamy klucz główny ze strony *jeden* do schematu relacji reprezentującej stronę *wiele* związku. W ten sposób otrzymujemy klucz obcy (ang. *foreign key*) w schemacie relacji ze strony *wiele* związku (NOT NULL).

Reguły transformacji (2)

- Związek binarny typu *jeden do wiele*, opcjonalny po obu stronach, reprezentujemy zazwyczaj nowym schematem relacji, w którym umieszczamy klucze główne obu encji (ze związku opcjonalnego po stronie jeden otrzymujemy kolumny NULL. Jeżeli co najmniej 60% instancji encji pozostaje w związku, to uznaje się związek za prawie obligatoryjny i zaleca się zastosować transformację według poprzedniej reguły.
- Dla związku opcjonalnego *wiele do wiele* tworzymy nowy schemat relacji (ewentualnie ze sztucznym kluczem głównym). Klucze główne z encji po obu stronach związku stają się kluczami obcymi w utworzonym schemacie relacji. Jeśli związek posiadał atrybuty dodatkowe, to stają się one atrybutami w nowo utworzonym schemacie relacji. Kluczem głównym może być klucz złożony z obu kluczy obcych lub nowo wprowadzony klucz sztuczny.
- Atrybutom, które odpowiadają kluczom kandydującym encji, należy zapewnić unikalność wartości.
- Jeśli, w wyniku transformacji, dla utworzonego schematu relacji pojawi się klucz naturalny, a schemat posiada również klucz sztuczny, to można zbędny klucz sztuczny usunąć.

Definicja schematu relacji – przykład

REL/01 Kategorie/KATEGORIA
Opis schematu relacji **Kategorie**

Nazwa atrybutu	Dziedzina	Maska	OBL	Wart. dom.	Ograniczenia	Unikalność	Klucz	Referencje	Źródło danych
<u>IdK</u>	<u>Int+</u>		+			+	PK		SZBD
<u>NazwaK</u>	String[30]		+			+			USER

Opis atrybutów w schemacie relacji

Nazwa	Opis

Opis atrybutów w schemacie relacji – przykład

Znaczenie atrybutów w schemacie relacji **Kategorie**

Nazwa	Opis
<u>IdK</u>	Unikalny identyfikator kategorii, automatycznie nadawany przez system, klucz główny
<u>NazwaK</u>	Unikalna nazwa kategorii

Przykładowe dane w relacji

Przykładowe dane tabeli o schemacie relacji **Kategorie**

<u>IdK</u>	<u>NazwaK</u>
1	Warzywa
2	Owoce
3	Produkty mleczne
4	Mięso
5	Produkty zbożowe

Etap 12 – Schemat bazy danych i słownik atrybutów

Schemat bazy danych (ang. *database schema*) to zbiór schematów relacji o różnych nazwach.

Schemat relacji (ang. *relation schema*) to nazwa schematu relacji oraz zbiór atrybutów relacji.

Postać ogólna schematu bazy danych:

NAZWA BAZY DANYCH

Nazwa_schematu_relacji_1(*lista_atrybutów_schematu_relacji_1*)

Nazwa_schematu_relacji_2(*lista_atrybutów_schematu_relacji_2*)

....

Etap 12 – Schemat bazy danych i słownik atrybutów – przykład

12.1 Schemat bazy danych

Kategorie(IdK, NazwaK)

Produkty(IdP, NazwaP, #IdK, #IdProducenta, DataDodania, Opis)

Oferty(IdO, #IdP, #IdS, Cena, Ilosc, Jednostka, DataAktualizacji, DataWygasniecia)

Sklepy(IdS, NazwaS, DataDodaniaS, Autoryzacja, #IdU, OpisS, Adres)

Oceny(#IdU, #IdO, Ocena, Komentarz, DataDodaniaOc)

Etap 12.2 – Słownik atrybutów

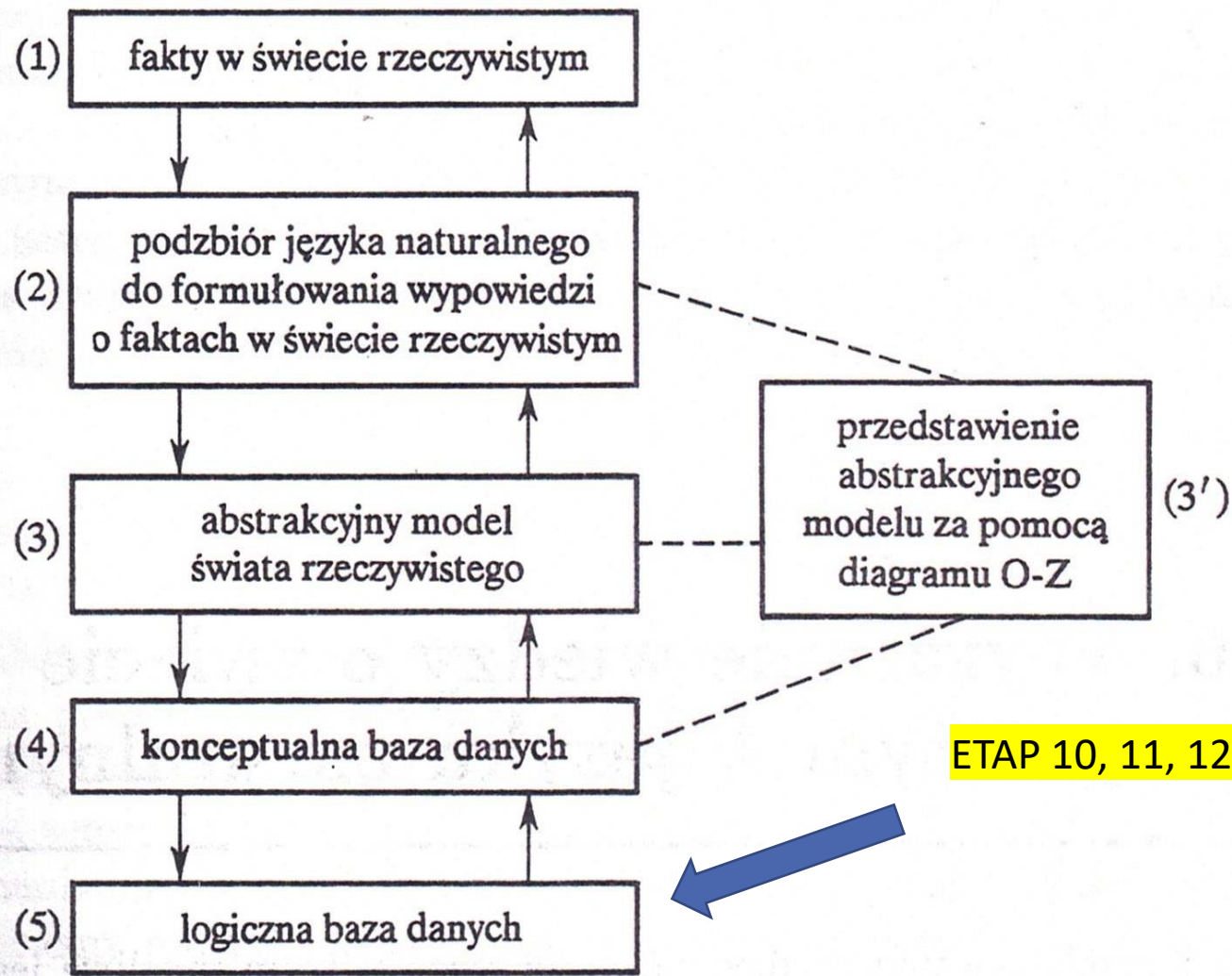
- Słownik atrybutów ułatwia wyszukiwanie informacji o wybranym atrybucie.
- Słownikiem (ang. *dictionary*) ogólnie nazywa się miejsce przechowywania terminów i ich objaśnień.
- **Słownik atrybutów** lub inaczej słownik danych (ang. *data dictionary*) tworzy się w celu zestawienia nazw kolumn i domen (dziedzin), określających typ danych, jakie mogą być przechowywane w danej kolumnie oraz ograniczenia, którym podlegają. Jest to pewnego rodzaju baza danych projektanta bazy danych.
- Słownik danych przedstawiamy w postaci 3-kolumnowej tabeli, w której wyszczególniamy nazwy atrybutów schematów relacji, dziedziny atrybutów oraz przynależność atrybutu do schematu relacji.
- Nazwy atrybutów powinny być uporządkowane alfabetycznie.

Słownik atrybutów – przykład

12.2 Słownik atrybutów

Nazwa atrybutu	Dziedzina atrybutu	Przynależność do schematu relacji
<u>IdK</u>	<u>Int+</u>	Kategorie
<u>IdK</u>	<u>Int+</u>	Produkty
<u>NazwaK</u>	String[30]	Kategorie
<u>IdP</u>	<u>Int+</u>	Oferty
<u>IdP</u>	<u>Int+</u>	Produkty
<u>NazwaP</u>	String[30]	Produkty
<u>DataDodania</u>	<u>Date</u>	Produkty
Opis	String[255]	Produkty

Jak przedstawić semantykę świata w bazie danych



Schemat ustalania związku między danymi w bazie a faktami w świecie rzeczywistym

Perspektywy

- **Perspektywa** (ang. *view*) – nazwana tabela (wirtualna)
- W odróżnieniu od tabeli bazowej (wyjściowej) nie może istnieć samodzielnie.
- Definiowana jest za pomocą jednej lub wielu tabel wyjściowych (bazowych lub perspektyw), ich wszystkich lub niektórych atrybutów.
- W szerszym znaczeniu jest to pojęciowy obraz danych, przypisany do pojedynczego użytkownika lub do grupy użytkowników, który pomija dane dla nich nieistotne.
- Istotne jest zapewnienie przezroczystości (ang. *transparency*), aby użytkownik miał wrażenie, że operuje na danych zapamiętanych, a nie wirtualnych.

Etap 13 – definiowanie perspektyw

W etapie definiowania perspektyw należy:

- wyszczególnić grupy użytkowników bazy danych,
- wyszczególnić perspektywy dla każdej grupy użytkowników,
- wyszczególnić transakcje dla każdej perspektywy;
- określić dane uczestniczące w transakcji – dane te można opisać podając nazwę relacji i atrybut, gdzie dana jest przechowywana oraz dopuszczalne operacje wykonywane na tej danej.

Opisy można przedstawić w formie tabeli o nagłówku:

Nazwa relacji, Atrybut, Zapis, Odczyt, Modyfikacja
zaznaczając symbolem + w odpowiedniej kolumnie dla odpowiedniego atrybutu
możliwość wykonania danej operacji (zapis, odczyt, modyfikacja).

Definicja perspektywy – przykład

PER/001 Dane pracownika

Użytkownik: Administrator

Transakcje: TRA/001, TRA/002, TRA/005

TRA/001 – Wprowadzenie danych pracownika

Tabela transakcji: Wprowadzenie danych pracownika

Nazwa relacji	Atrybut	Zapis	Odczyt	Modyfikacja
Osoby	<i>Id</i>	+		
Osoby	<i>Nazwisko</i>	+		
Osoby	<i>Imię</i>	+		
Osoby	<i>DataUrodzenia</i>	+		
Dzieci	<i>ImięDziecka</i>	+		
Dzieci	<i>DataUrDziecka</i>	+		

Podsumowanie

Przedstawiony sposób projektowania relacyjnej bazy danych wg autorskiej metodyki strukturalnej umożliwia opracowanie schematu bazy danych, w której wszystkie schematy relacji są w 3PN.

Poprawny projekt bazy danych powinien być:

- **kompletny** – wszystkie składowe projektu są zdefiniowane w sposób odpowiedni i wyczerpujący,
- **niesprzeczny** – każdy element jest zdefiniowany jednoznacznie, nie ma definicji sprzecznych ze sobą,
- **spójny** – informacje zawarte w kolejnych etapach projektu (definicjach, diagramach) są semantycznie zgodne,
- **zgodny z regułami składniowymi notacji** – opis projektu zależy od przyjętej notacji, z którą musi być zgodny.

Literatura

- Mazur Hanna, Mazur Zygmunt, *Projektowanie relacyjnych baz danych*. Oficyna Wydaw. PWroc, Wrocław, 2004.
- Mazur Hanna, Mazur Zygmunt, *Metodyka strukturalna projektowania relacyjnych baz danych*, Politechnika Wroc, Wrocław, 2020.



„ZPR PWr – Zintegrowany Program Rozwoju Politechniki Wrocławskiej”

