

# Zrozumieć UML

Podejście praktyczne  
na przykładach z projektów  
rozporządzeń do ustawy PGiK

dr inż. Agnieszka Chojka  
Katedra Geodezji Szczegółowej  
Uniwersytet Warmińsko-Mazurski w Olsztynie

Wisła – Malinka, 7-9.09.2011r.



# Plan warsztatów

1. Wprowadzenie
2. Podstawowe pojęcia i definicje
3. Podstawy języka UML
4. Przykłady schematów aplikacyjnych UML z rozporządzeń



Zrozumieć UML – podejście praktyczne

# W P R O W A D Z E N I E

# INSPIRE

- *Ustawa o infrastrukturze informacji przestrzennej z dnia 4 marca 2010 r. jest transpozycją dyrektywy 2007/2/WE Parlamentu Europejskiego i Rady z dnia 14 marca 2007 r. ustanawiającą infrastrukturę informacji przestrzennej we Wspólnocie Europejskiej (INSPIRE).*
- **Transpozycja** – przystosowanie przepisów dyrektywy do prawa krajowego,
  - pociąga za sobą konieczność nowelizacji wielu ustaw, w tym ustawy *o prawie geodezyjnym i kartograficznym z dnia 17 maja 1989 r.*
  - oraz konieczność zmiany odpowiednich rozporządzeń do powyższej ustawy.

# Transpozycja

- Część rozporządzeń do ustawy *prawo geodezyjne i kartograficzne*
  - nie traci aktualności (np. uprawnienia zawodowe),
  - część wymaga zmian np. *rozporządzenie w sprawie geodezyjnej ewidencji sieci uzbrojenia terenu, czy rozporządzenie w sprawie bazy danych obiektów topograficznych,*
  - część wymaga opracowania od nowa.

# Rozporządzenia

- Integralną częścią aktualnie opracowywanych projektów rozporządzeń są:
  - specyfikacje **modeli pojęciowych**, dla danych zakresów przedmiotowych, w postaci:
    - **schematów aplikacyjnych UML**,
    - **katalogów obiektów**,
    - **schematów aplikacyjnych GML**.



Schematy aplikacyjne są przygotowywane zgodnie z normami ISO serii 19100 w dziedzinie informacji geograficznej, co ma zapewnić osiągnięcie interoperacyjności.



Zrozumieć UML – podejście praktyczne

**PODSTAWOWE  
POJĘCIA I DEFINICJE**

# Infrastruktura danych przestrzennych

- **Perspektywa danych** (ang. *data-centric view*):
  - schematy aplikacyjne,
  - metadane.
- **Perspektywa usług** (ang. *service-centric view*):
  - interoperacyjność,
  - architektura zorientowana na usługi.



# Perspektywa danych

## ■ **Podjęcie oparte na modelu**

(ang. *Model Driven Approach*):

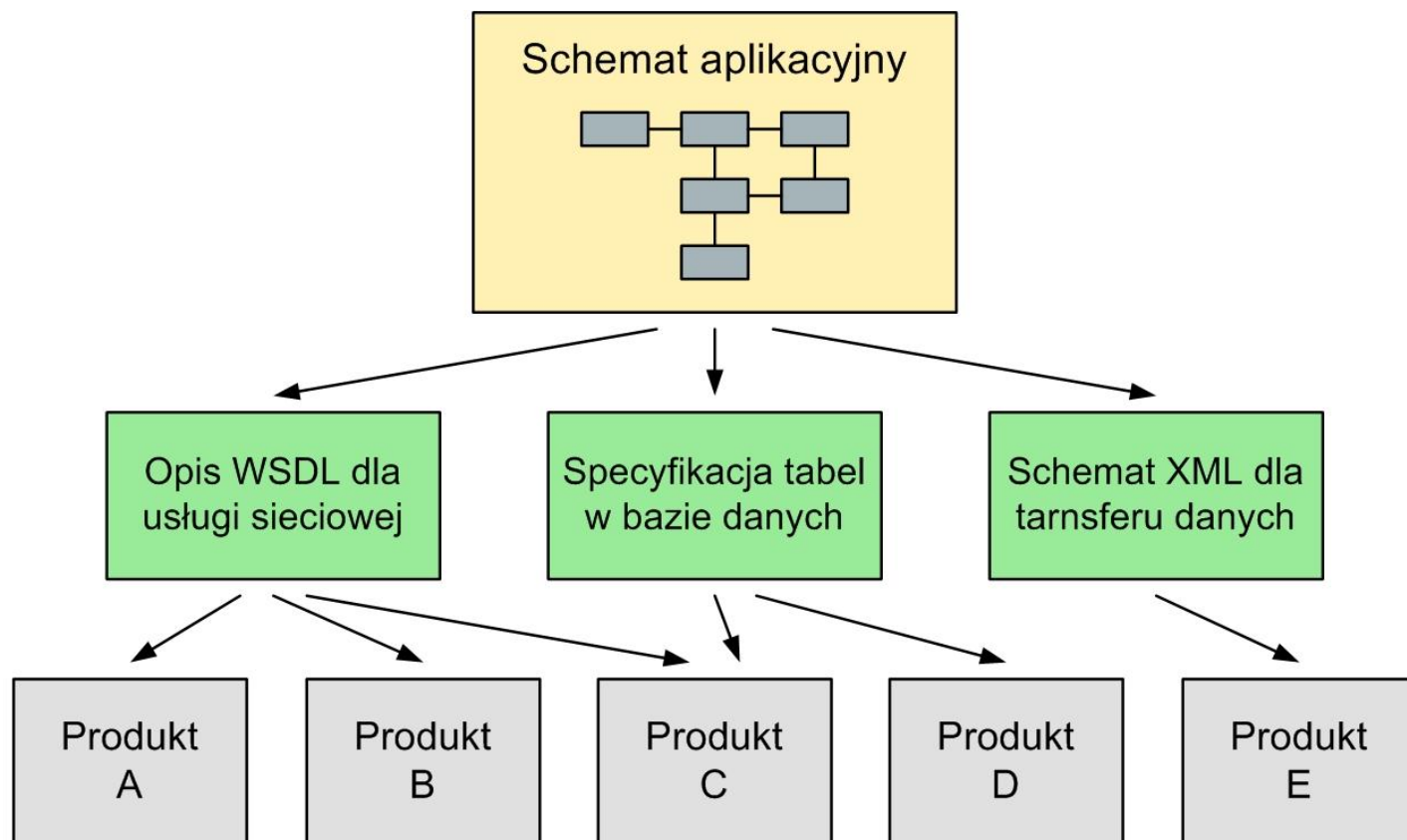
- **niezależny od implementacji schemat aplikacyjny** zostaje odwzorowany na **różne specyfikacje** (wykorzystujące różne technologie, np. usługi sieciowe, relacyjne bazy danych, XML),
- **specyfikacje** mogą zostać zaimplementowane (wdrożone) na **różnych platformach** sprzętowo-programowych.

# Podójście oparte na modelu

niezależny od implementacji opis semantyki i logiczna struktura danych

specyfikacje dla rójnych technik

implementacje



Podójście oparte na modelu

Źródło: prCEN/TR 15449, 2006.

# Podjęcie oparte na modelu

## ■ **Koncepcja MDA**

(ang. *Model Driven Architecture*):

- opracowana przez OMG  
(ang. *Object Management Group*),
- **Cel:** rozwiązywanie problemów związanych z integracją systemów informatycznych pochodzących od różnych dostawców oraz działających na różnych platformach informatycznych (wykorzystujących różne technologie, np. różne systemy operacyjne, różne standardy sieciowe, różne języki programowania).

# MDA – modele systemu

- **CIM** (ang. *Computation Independent Model*)
  - **specyfikacja wymagań**,
- **PIM** (ang. *Platform Independent Model*)
  - **model pojęciowy** (niezależny od platformy),
- **PSM** (ang. *Platform Specific Model*)
  - **model logiczny** (zależny od wybranej platformy),
- **Implementacja** (ang. *Implementation Model*)
  - **model fizyczny** (np. program, struktura bazy danych).

# Technologia MDA

- Kluczową rolę odgrywa modelowanie systemu w języku **UML** (ang. *Unified Modeling Language*).
  - Język modelowania służący do opisu świata obiektów w analizie obiektowej oraz programowaniu obiektowym.
- Zalecany przez **normy ISO serii 19100** język schematu pojęciowego (ang. *conceptual schema language*).
  - Środek formalny modelowania informacji geograficznej,
  - Umożliwia zapis informacji geograficznej (struktur danych przestrzennych) w sposób niezależny od platform sprzętowo-programowych, zapewniając tym samym interoperacyjność między różnymi systemami geoinformacyjnymi.

# Interoperacyjność

- **Cel norm ISO serii 19100!!!**
- **Zdolność do współdziałania** – m.in. zdolność do komunikowania, a co za tym idzie dokonywania transferu danych między systemami.
- **Jak osiągnąć?**
  - **Metodologia modelowania informacji geograficznej za pomocą schematów pojęciowych:**
    - podstawa dla spójnych realizacji GIS w odmiennych środowiskach sprzętowo-programowych,
    - zapewnia współdziałanie różnych realizacji GIS.

# Interoperacyjność różnych systemów

- Semantyka zawartości i struktur logicznych danych przestrzennych – tzw. **schemat aplikacyjny** (ang. *application schema*).
- Niezależna od platformy sprzętowo-programowej struktura danych, która może reprezentować dane zgodnie ze schematem aplikacyjnym.
  - Publikowanie danych dla poszczególnych tematów danych przestrzennych, zgodnie ze specyfikacjami danych przestrzennych (w UML) oraz przepisami wykonawczymi do dyrektywy INSPIRE.

# Terminologia

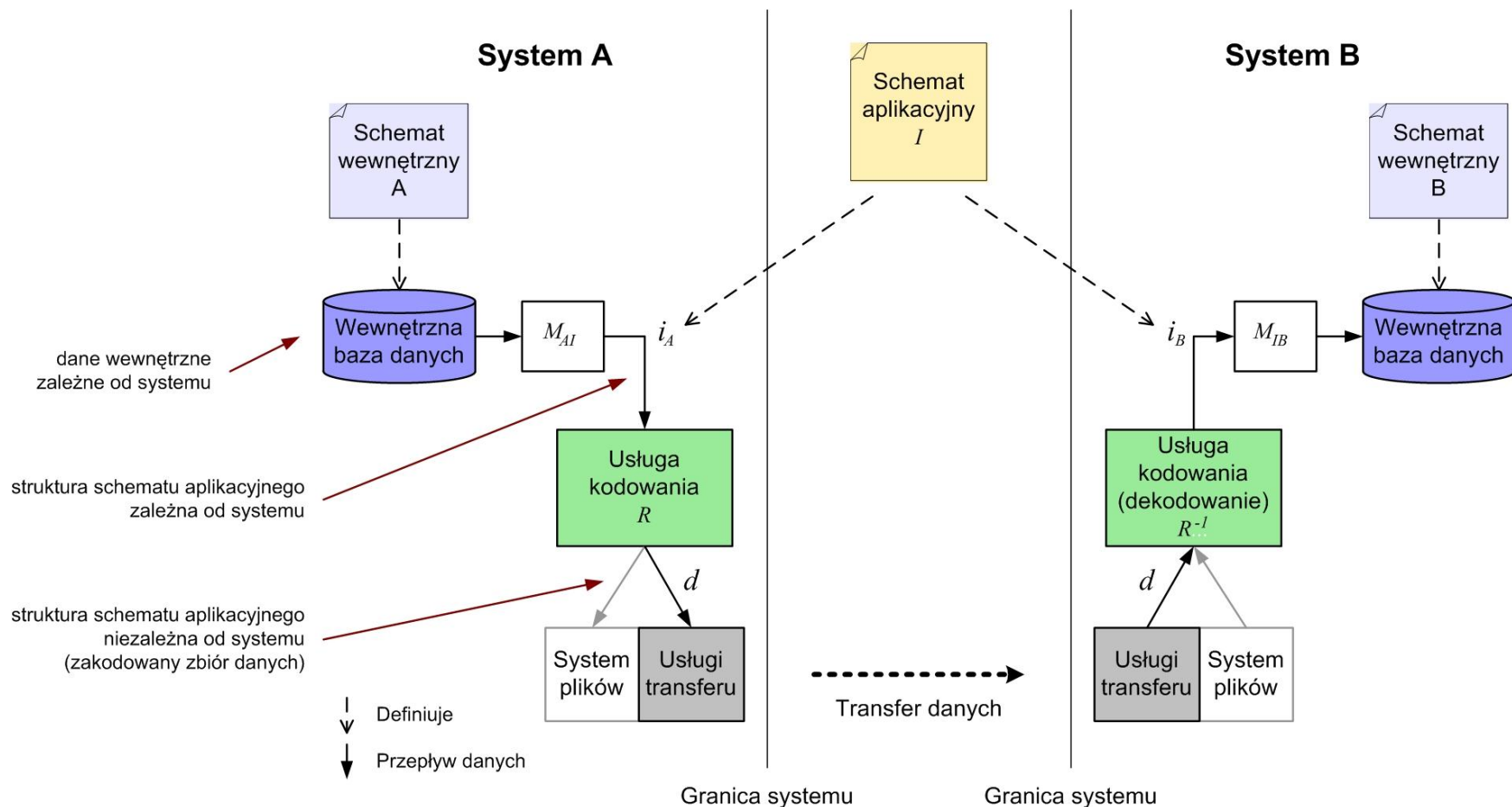
- **Model pojęciowy** (ang. *conceptual model*)
  - model definiujący pojęcia z pewnej przestrzeni rozważań (przedmiotu zainteresowań).
- **Schemat pojęciowy** (ang. *conceptual schema*)
  - formalny opis modelu pojęciowego w określonym języku schematu pojęciowego.
- **Schemat aplikacyjny** (ang. *application schema*)
  - schemat pojęciowy dla danych wykorzystywanych przez jedną lub więcej aplikacji, dla określonego zakresu przedmiotowego.



# Schemat aplikacyjny

- Podstawa pomyślnej wymiany danych:
  - definiuje możliwą zawartość oraz strukturę danych,
  - dostępny dla obu uczestników procesu wymiany danych (nadawcy i odbiorcy).
- Do wymiany danych:
  - zapisany w języku schematu pojęciowego UML, zgodnie z normami ISO/TS 19103 i ISO 19109 (reguły budowy schematów aplikacyjnych).

# Interoperacyjna wymiana danych



Ogólna idea wymiany danych między dwoma systemami

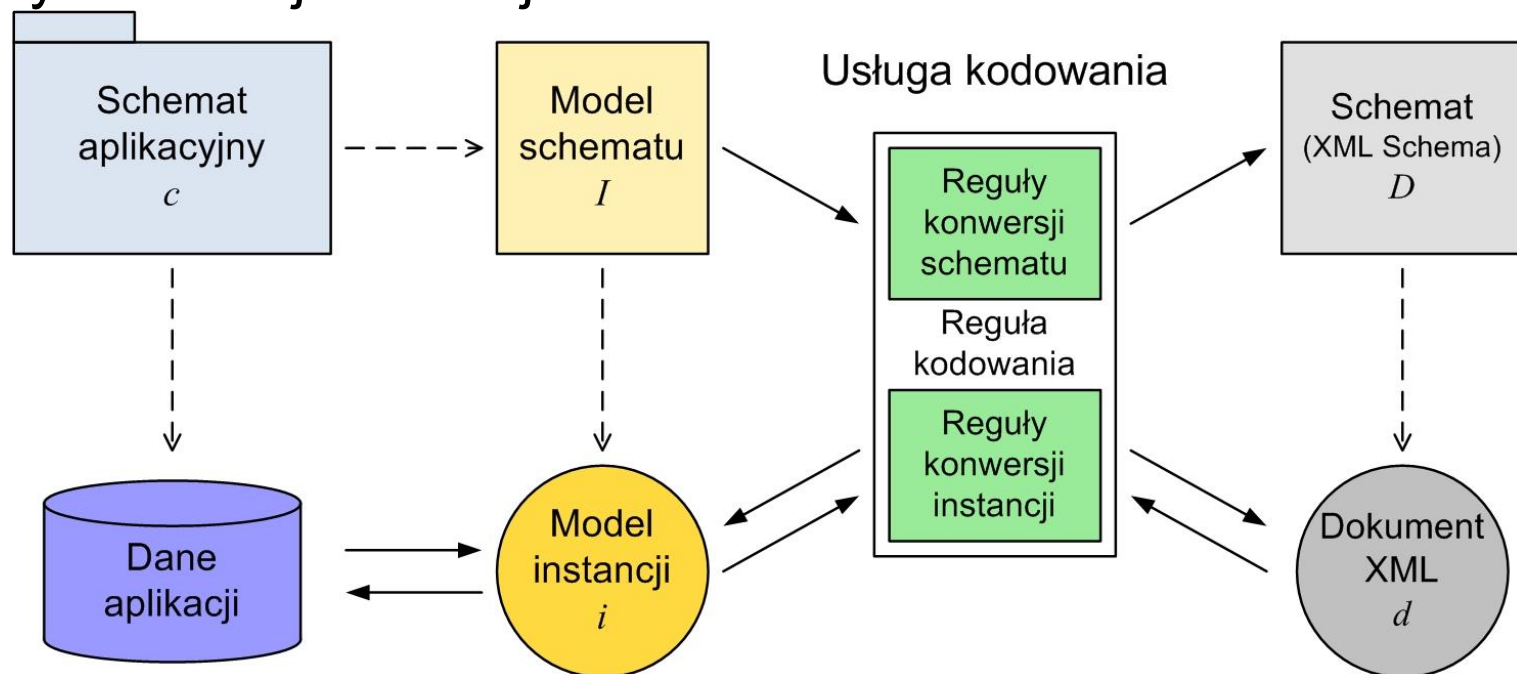
Źródło: prCEN/TR 15449, 2006; ISO/DIS 19118:2005

# Reguły kodowania

- Normy ISO serii 19100 określają dwie reguły kodowania oparte na języku **XML** (ang. *Extensible Markup Language*):
  - **ISO 19136 (GML), Załącznik E**
    - reguła kodowania oparta na XML dla schematów aplikacyjnych zgodnych z normą ISO 19109, które mogą być przedstawione za pomocą ograniczonego profilu UML, pozwalającego na konwersję do języka XML Schema,
  - **ISO/TS 19139**
    - reguła kodowania oparta na XML dla schematów pojęciowych określających typy, które opisują zasoby geograficzne, np. metadane według ISO 19115 i katalogi obiektów według ISO 19110.

# Reguła kodowania

- Powinna określać:
  - reguły konwersji schematu,
  - reguły konwersji instancji.



Reguła kodowania oparta na XML  
Źródło: ISO/DIS 19118:2005

# Katalog obiektów (ang. *feature catalogue*)

- Katalog zawierający **definicje** oraz **opisy**
  - typów obiektów,
  - atrybutów obiektów,
  - powiązań obiektów,  
występujących, co najmniej  
w jednym schemacie aplikacyjnym.

ISO 19110:2010 *Geographic information – Methodology for feature cataloguing*

# Podsumowanie

- **Model pojęciowy** – definiuje pojęcia z pewnej przestrzeni rozważań.

- **Schemat aplikacyjny** = model pojęciowy + UML + dziedzina.

- **Katalog obiektów** – definicje i opisy typów obiektów, atrybutów obiektów oraz powiązań obiektów.

TECHNICAL SPECIFICATION ISO/TS 19103

Final edition 2005-07-15

Geographic information — Conceptual schema language

Information géographique — Schéma de langage conceptuel

INTERNATIONAL STANDARD

ISO 19109

Final edition 2005-08-15

Geographic information — Rules for application schema

Information géographique — Règles de schéma d'application



ISO

INTERNATIONAL STANDARD

ISO 19110

Final edition 2005-02-15

Geographic information — Methodology for feature cataloguing

Information géographique — Méthodologie de catalogage des entités



Reference number ISO 19115:2005(E)

© ISO 2005

# Podsumowanie

- **UML** – język modelowania informacji geograficznej, służy do opisu świata obiektów rzeczywistych.
- **GML** – język implementowania, język znaczników geograficznych – format wymiany danych przestrzennych pomiędzy różnymi systemami geoinformacyjnymi.

TECHNICAL  
SPECIFICATION

ISO/TS  
19103

First edition  
2005-07-15

Geographic information — Conceptual  
schema language

Information géographique — Schéma de langage conceptuel

INTERNATIONAL  
STANDARD

ISO  
19136

First edition  
2007-09-01

Geographic information — Geography  
Markup Language (GML)

Information géographique — Langage de balisage en géographie  
(GML)



Reference number  
ISO 19136:2007(E)

© ISO 2007



Zrozumieć UML – podejście praktyczne

**PODSTAWY  
JĘZYKA UML**



# Schemat aplikacyjny

- Powinien być zapisany w języku schematu pojęciowego UML według zasad określonych w normach ISO/TS 19103 i ISO 19109.
- Składa się z pojęć określonych przez aplikację wyrażonych jako klasy i powiązania.
- Niektóre z klas mogą być zaimportowane ze schematów znormalizowanych z innych standardów.

# ISO 19109

- Ogólne reguły budowy i dokumentowania schematów aplikacyjnych:
  - zasady modelowania pojęciowego obiektów oraz ich właściwości,
  - reguły definiowania schematu aplikacyjnego za pomocą języka schematu pojęciowego,
  - wyrażanie pojęć z modelu pojęciowego w postaci typów danych w schemacie aplikacyjnym,
  - zasady integracji schematu aplikacyjnego ze znormalizowanymi schematami pojęciowymi informacji geograficznej.

# Istota ISO 19109

- Definicja obiektu geograficznego (ang. *geographic feature*), który reprezentuje dowolny element ze świata rzeczywistego, np. budynek, drzewo, obraz satelitarny.
- Integracja obiektu z modelem informacji geograficznej w jednorodny (homogeniczny) sposób:
  - *General Feature Model* (GFM) – abstrakcyjny obiekt z atrybutami (właściwości obiektu) i operacjami (zachowanie obiektu).

# ISO 19109 – GFM

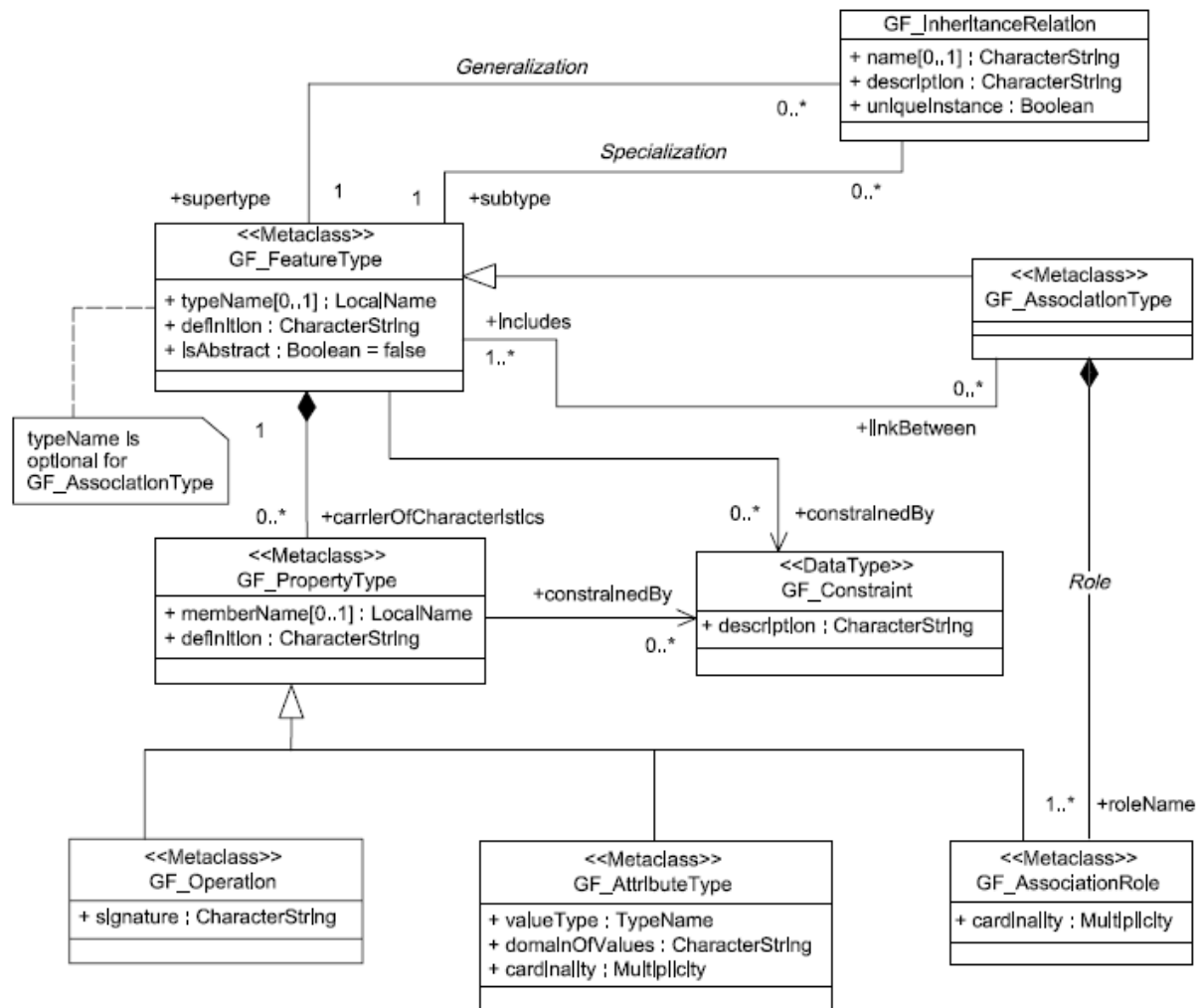


Figure 5 — Extract from the General Feature Model

# ISO/TS 19103

- **Profil UML** w dziedzinie informacji geograficznej (geoinformatyki/geomatyki) dostosowany do norm ISO serii 19100 (przyjęta konwencja nazywania i modelowania pozostaje niezmienna dla całej serii norm).
  - m.in. zasady definiowania klas, atrybutów, typów danych, operacji, związków i stereotypów.
  - modele normatywne wykorzystują diagramy klas i diagramy pakietów. Inne diagramy UML mogą być stosowane informacyjnie.

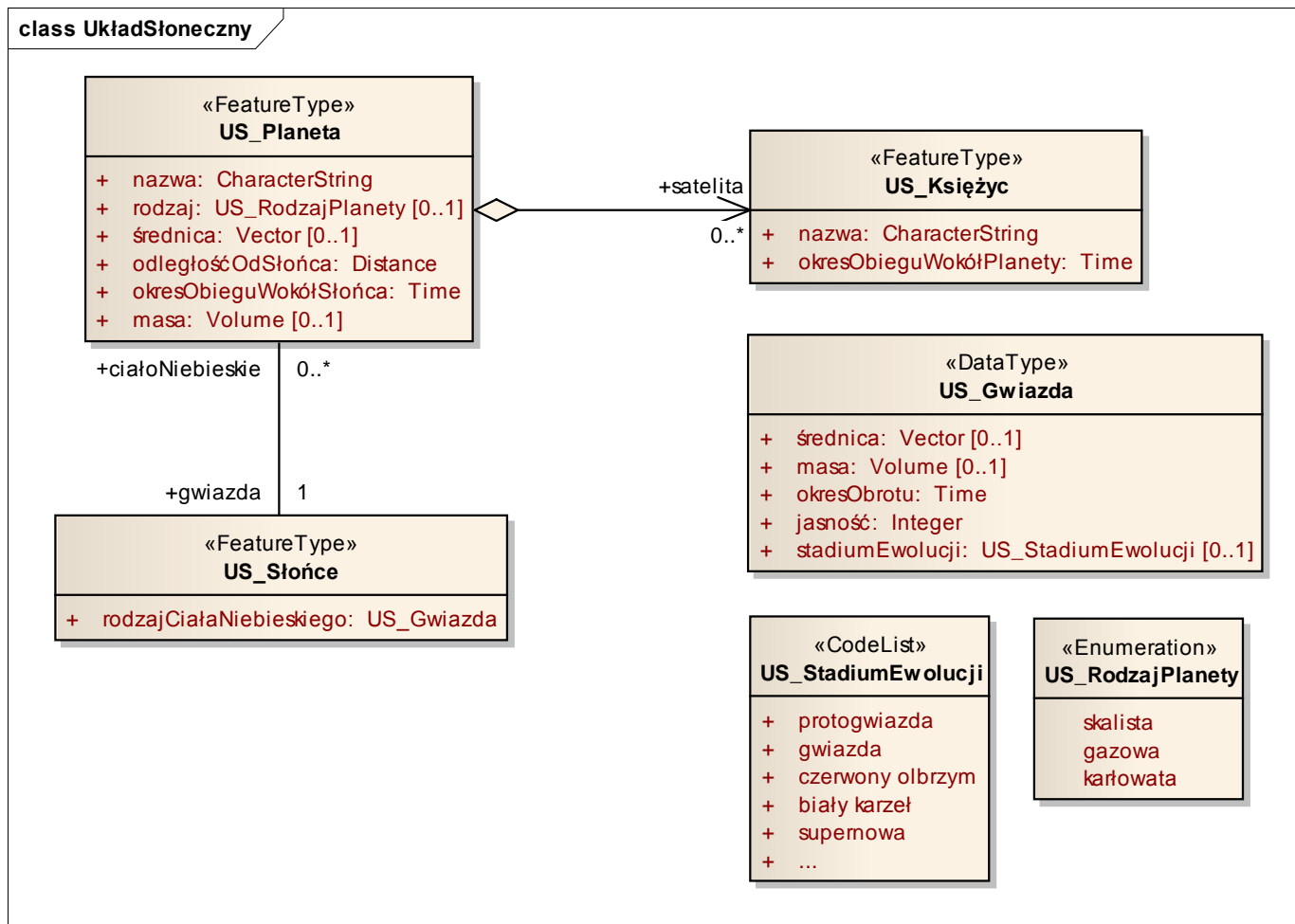
# UML

- ang. *Unified Modeling Language*
- pol. *zunifikowany (ujednolicony) język modelowania*
  
- Język formalny służący do opisu świata obiektów w analizie obiektowej oraz programowaniu obiektowym.

# Diagram klas w UML

- Opis:
  - zbioru danych,
  - zależności pomiędzy nimi.
  
- Podstawowe elementy:
  - oznaczenia klas,
  - związków pomiędzy nimi,
  - oznaczenia pomocnicze  
(np. stereotypy, metki, ograniczenia itp.).

# Diagram klas – przykład

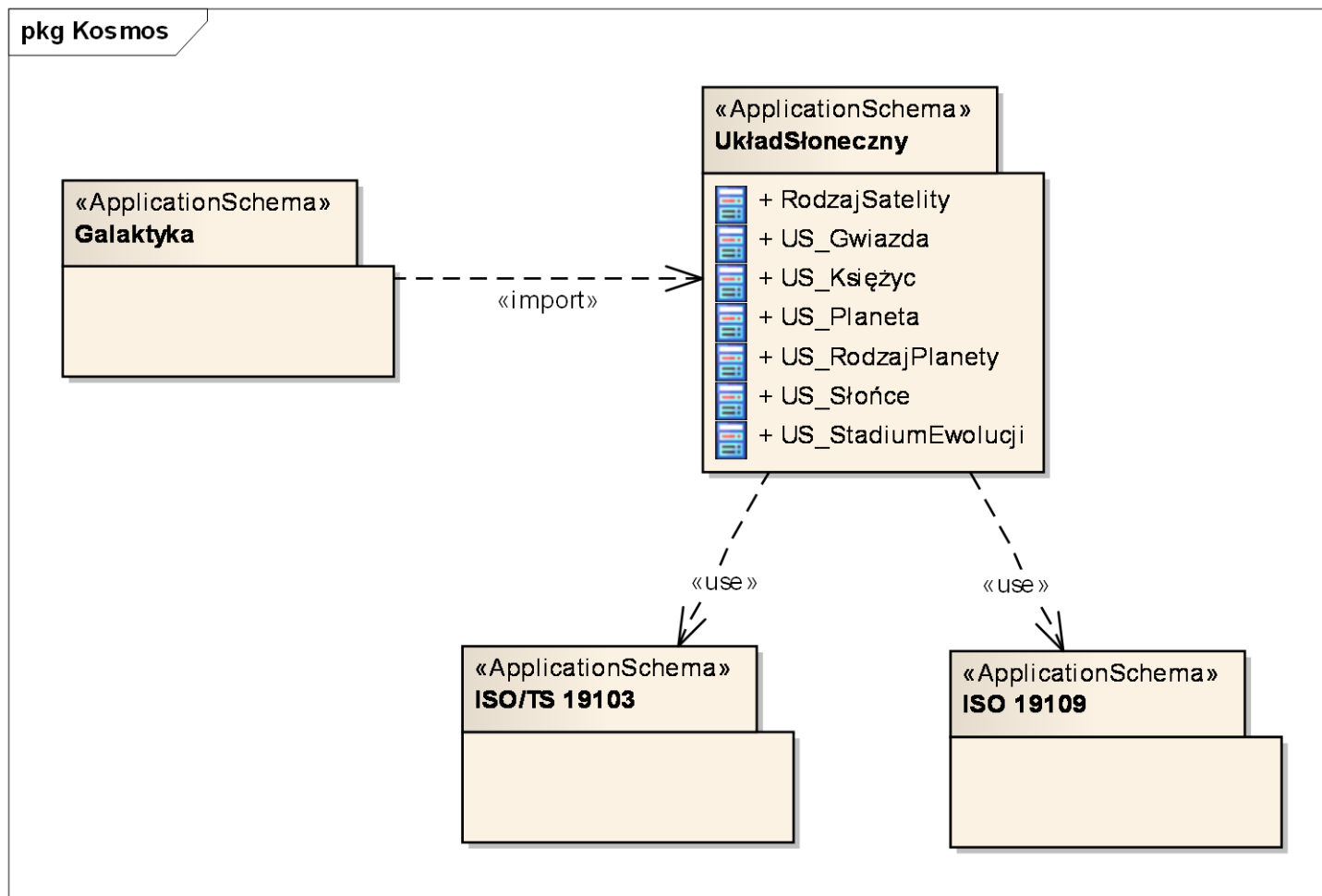




# Diagram pakietów w UML

- Zawiera m.in. klasy i diagramy klas.
- Porządkuje strukturę zależności w systemie, w który można wyróżnić bardzo wiele klas, przypadków użycia itp.
- Przedstawia podział systemu z logicznego punktu widzenia.

# Diagram pakietów – przykład



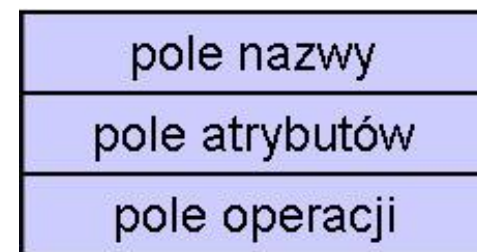
# Klasa

- Opis zbioru obiektów, które mają takie same atrybuty, operacje, związki i znaczenie (wzorzec dla tworzonego obiektu).
- Według ISO/TS 19103 każda klasa powinna być zdefiniowana wewnątrz określonego pakietu.

- **Symbol graficzny:**

- Prostokąt podzielony na trzy pola:

- **pole nazwy** – nazwa klasy, stereotyp, lista wartości etykietowanych,
- **pole atrybutów** – lista atrybutów klasy,
- **pole operacji** – lista operacji (metod) zdefiniowanych dla klasy.



# Nazwa

- Każda klasa musi mieć przypisaną nazwę, która wyróżnia ją spośród innych klas.
- Napis, pisany pogrubioną czcionką.
- Na ogół w formie krótkiego rzeczownika lub wyrażenia rzeczownikowego (mianownik, I. poj.), pochodzącego ze słownictwa modelowanego systemu.
- Każdy wyraz w nazwie zaczyna się wielką literą.
- Jeżeli nazwa klasy jest wieloczłonowa, każdy kolejny człon zaczyna się wielką literą, a człony są ze sobą „sklejone” (brak spacji, podkreślenia), np. Budynek, PunktAdresowy.

# Nazwa

- Zgodnie z ISO/TS 19103, nazwa klasy powinna być poprzedzona przedrostkiem składającym się z dwóch wielkich liter, który identyfikuje pakiet lub normę, z której taka klasa pochodzi, np. **GM** – geometria, **TP** – topologia.

AD	Abstract address	ISO 19133	LI	Lineage	ISO 19115
CC	Changing coordinates	ISO 19111	LR	Linear reference	ISO 19133
CI	Citation	ISO 19115	MD	Metadata	ISO 19115
CV	Coverages	ISO 19123	NT	Network position	ISO 19133
DQ	Data quality	ISO 19115	PF	Feature Portrayal	ISO 19117
DS	Dataset	ISO 19115	PS	Positioning Services	ISO 19116
EX	Extenet	ISO 19115	RS	Reference System	ISO 19111, ISO 19115
FC	Feature Catalogue	ISO 19110	SC	Spatial coordinates	ISO 19111
FD	Feature ID	ISO 19133	SD	Sensor and Data Models	ISO 19130
FE	Feature	ISO 19109	SI	Spatial Identification	ISO 19112
FT	Feature Topology	ISO 19107	SV	Services	ISO 19119
GF	Gebneral feature model	ISO 19109	TM	Temporal	ISO 19108
GM	Geometry model	ISO 19107	TP	Topology	ISO 19107
GR	Graph	ISO 19107	TS	Simple Topology	ISO 19107

# Atrybut

- Nazwana właściwość klasy.
- Klasa może mieć dowolną liczbę atrybutów albo nie mieć ich wcale.
- Nazwa atrybutu może być dowolnym tekstem (jak nazwa klasy), na ogół w formie rzeczownika lub wyrażenia opisującego właściwość danej klasy (mianownik, l. poj.).
  - Każdy wyraz w nazwie (poza pierwszym) zaczyna się wielką literą, kolejne wyrazy są ze sobą „sklejone”, np. nazwisko, dataUrodzenia.
- Musi mieć przypisany typ danych, wybrany spośród obowiązujących typów podstawowych.

# Operacja

- Abstrakcja czegoś, co można zrobić z każdym obiektem danej klasy.
- Klasa może mieć dowolną liczbę operacji albo nie mieć ich wcale.
- Nazwa operacji może być dowolnym tekstem, na ogół w formie czasownika lub wyrażenia opisującego pewne zachowanie danej klasy.
  - Każdy wyraz w nazwie (poza pierwszym) zaczyna się wielką literą, kolejne wyrazy są ze sobą „sklejone”, np. usuń, zatwierdźOperację.

# Widoczność atrybutów i/lub operacji

- Dodatkowo określana na diagramach UML:
  - + oznacza, że atrybut (operacja) jest publiczny (ang. *public*), czyli dostępny spoza klasy,
  - – oznacza, że atrybut (operacja) jest prywatny (ang. *private*), czyli niedostępny spoza swojej klasy,
  - # oznacza, że atrybut (operacja) jest chroniony (ang. *protected*), czyli dostępny dla klas pochodnych (dziedziczących) od danej klasy.
  
  - / oznacza, że wartość danego atrybutu jest wyliczana (ang. *derived*) na podstawie wartości innych atrybutów.



# Typy danych

- ISO/TS 19103 grupuje podstawowe typy danych w 3 kategorie:
  - **Elementarne** (ang. *primitives types*),
  - **Implementacyjne** (ang. *implementation types*),
  - **Pochodne** (ang. *derived types*).

# Elementarne typy danych

- Podstawowe typy dla reprezentacji wartości:

- Integer,

- Decimal,

- Real,

- Vector,

- CharacterString,

- Date,

- Time,

- DateTime,

- Boolean,

- Logical,

- Probability,

- Multiplicity.

# Implementacyjne typy danych

- Szablony typów dla przedstawiania wielokrotnych wystąpień innych typów:
  - **zbiorowe** (ang. *collection*):
    - Set, Bag, Sequence, Dictionary,
  - **wyliczeniowe** (ang. *enumeration*):
    - Enumeration, CodeList,
  - **reprezentacyjne** (ang. *representation*):
    - Record (RecordType), GenericName.

# Pochodne typy danych

- Typy miar i jednostki miary:
  - Area,
  - Length (Distance),
  - Angle,
  - Scale,
  - Mtime,
  - Volume,
  - Velocity.

# Związki

- **Zależność** (ang. *dependency*)
- **Powiązanie** (ang. *association*)
- **Uogólnienie** (ang. *generalization*)

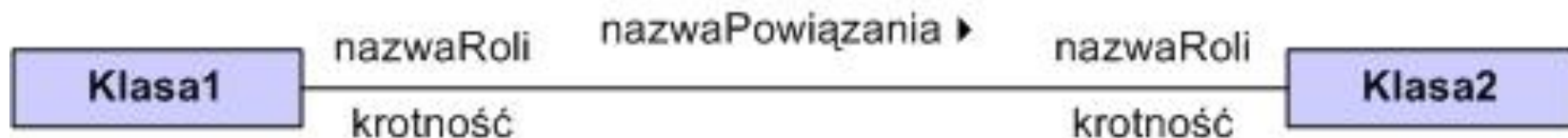
# Zależność

- Związek użycia między dwoma elementami.
- Zmiany dokonane w specyfikacji jednego elementu (np. klasy *Zamówienie*) mogą mieć wpływ na inny element, który używa (korzysta) tego pierwszego (np. na klasę *ZamówienieSpecjalne*), ale niekoniecznie na odwrót.
- **Graficznie** – linia przerywana z grotem skierowanym na element, od którego coś zależy:



# Powiązanie (asocjacja)

- Związek strukturalny równorzędnych partnerów, tzn. żadna klasa nie jest ważniejsza.
- Wskazuje, iż obiekty jednego elementu są połączone z obiektami innego.
- Dla każdej pary klas/obiektów można zdefiniować dowolnie wiele powiązań, z których każde reprezentuje inną zależność.
- **Graficznie** – linia ciągła łącząca klasy:



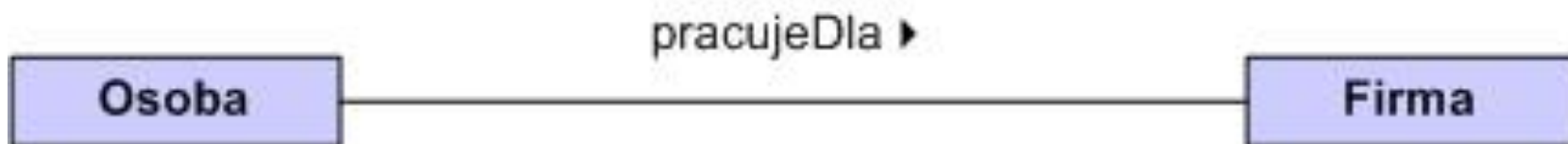
# Powiązanie

- Można opisać za pomocą:
  - nazwy,
  - ról,
  - krotności.



# Nazwa powiązania

- Powiązanie może mieć przypisaną nazwę, która określa istotę danego związku.
- Aby uniknąć niejednoznaczności, można podać kierunek odczytu (trójkątny znacznik przy nazwie).
- Nie jest konieczna, gdy określone są role.
- Zwykle jest to czasownik w czasie teraźniejszym, w 3os. l. poj.



# Rola

- Klasa biorąca udział w powiązaniu odgrywa w nim określoną rolę.
- „Oblicze”, które klasa przy jednym końcu powiązania prezentuje klasie przy drugim końcu.
- Dana klasa może odgrywać tę samą albo inną rolę w różnych powiązaniach.



# Krotność (liczność)

- Oznacza ile obiektów może być połączonych przez jeden egzemplarz powiązania.
- Przykłady możliwych krotność dla powiązania w UML:

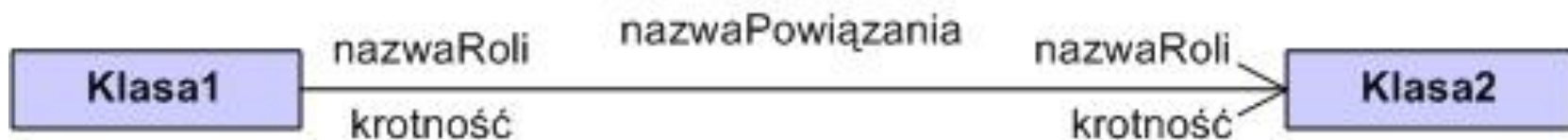
Krotność	Znaczenie	Np.
–	1 lub nie określono krotności	–
N	dokładnie N	5
*	dowolnie wiele	–
m..n	od m do n	1..5
m..*	od m do dowolnie wiele	2..*

# Powiązania w UML

- **Nawigacja** (ang. *navigation*)
- **Agregacja zwykła** (ang. *aggregation*)
- **Agregacja całkowita, kompozycja** (ang. *composition*)

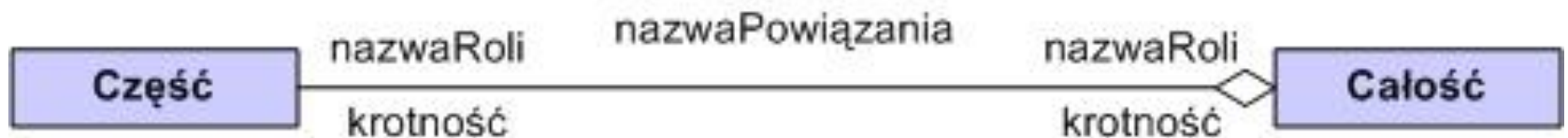
# Nawigacja

- Powiązanie jednokierunkowe, tzn. tylko jedna z klas biorących udział w powiązaniu „wie” o drugiej klasie.
- **Graficznie** – strzałka dołączona do zakończenia linii powiązania i wskazująca na daną klasę. Brak strzałki oznacza, że powiązaniem można „poruszać się” w obu kierunkach (obie klasy wiedzą o sobie).



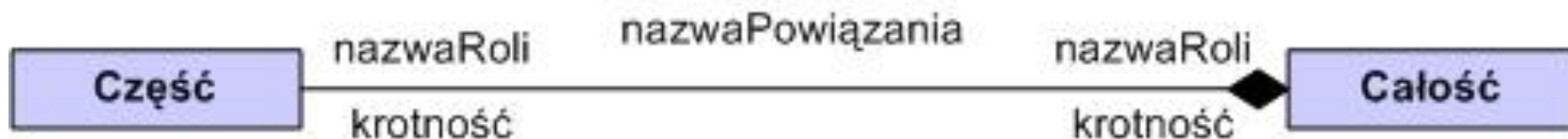
# Agregacja zwykła

- Związek „**posiadania**” między klasami, klasa-całość składa się z zestawu innych klas, które stanowią jej składniki (klasy-części); całość „ma/posiada” części.
- Klasy-części mogą być składnikami innych agregacji, zaś ich czas życia nie jest ograniczony do czasu życia klasy-całości.
- **Graficznie** – pusty romb umieszczonego na końcu linii powiązania od strony klasy-całości:



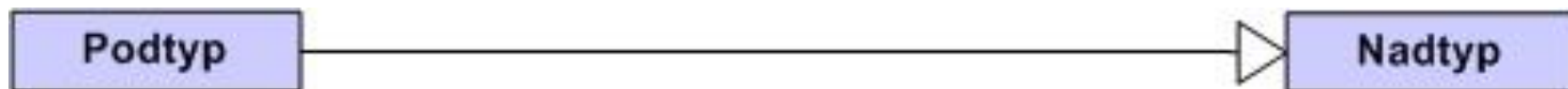
# Agregacja całkowita (kompozycja)

- Związek „**zawierania**” między klasami, gdzie klasa-całość zawiera inne klasy, które stanowią jej składniki (klasy-części); całość „zawiera” części.
- Klasy-części nie mogą być składnikami innych klas-całości, zaś ich czas życia jest ograniczony do czasu życia klasy-całości.
- **Graficznie** – wypełniony romb umieszczony na końcu linii powiązania od strony klasy-całości:



# Uogólnienie (dziedziczenie)

- Związek między elementem ogólnym (generalizacją, nadklasą, przodkiem) a jego pewnym specyficznym rodzajem (specjalizacją, podklasą, potomkiem).
- Potomek dziedziczy wszystkie właściwości przodka:
  - atrybuty,
  - powiązania,
  - operacje,
  - ograniczenia.
- Najczęściej potomek ma jeszcze własne cechy, poza odziedziczonymi.
- **Graficznie** – linia ciągła zakończona niewypełnionym trójkątem wskazującym przodka:





# Mechanizmy rozszerzenia w UML

- Pozwalają na rozszerzenie semantyki (słownictwa) UML:
  - umożliwiają dostosowanie UML do potrzeb konkretnego zadania i pozwalają na przystosowanie go do nowych technologii.
  - pozwalają na uszczegółowienie modelu, zdefiniowanie nowych elementów, przystosowanie notacji do specyficznej dziedziny lub preferencji użytkownika.
- Są to:
  - **stereotypy** (ang. *stereotypes*),
  - **metki** (ang. *tagged values*),
  - **ograniczenia** (ang. *constraints*).

# Stereotyp

- Wyrażenie umożliwiające klasyfikację elementów modelu.
- Wspólna, nazwana własność obiektów, klas, powiązań, atrybutów, ograniczeń.
- Dla każdego rodzaju elementu UML istnieje lista dozwolonych stereotypów.
- ISO/TS 19103 definiuje 11 stereotypów dla informacji geograficznej (9 to standardowe stereotypy UML), np.:
  - «**Enumeration**»,
  - «**CodeList**»,
  - «**DataType**».

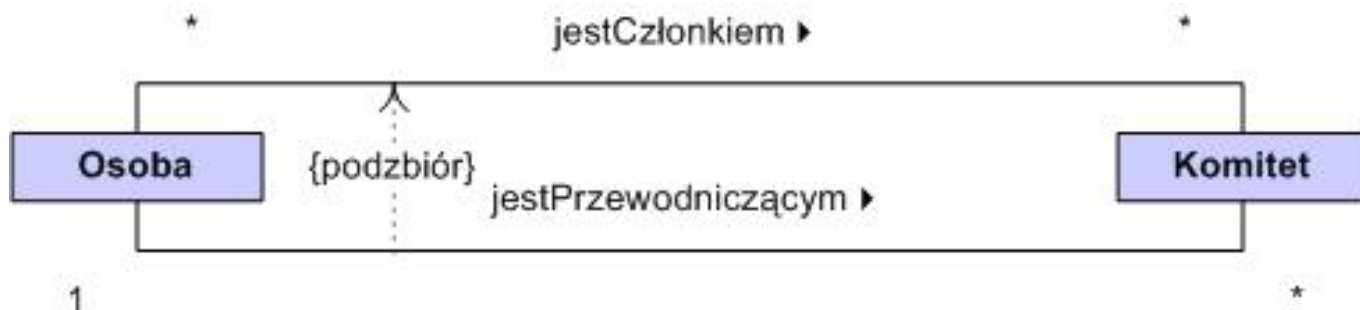
# Metka (wartość etykietowana)

- Pozwala dołączyć do elementu dodatkowe właściwości.
- Para „*klucz = wartość*”.
- Listę wartości etykietowanych oddzielonych przecinkami umieszcza się w nawiasach klamrowych {}:

```
{ autor = 'Marcin Woicki', dataWydania = '2000' }
```

# Ograniczenie

- Określa restrykcje nakładane na elementy modelu.
- Może stanowić wyrażenia języka naturalnego, języka formalnego (np. OCL w UML), może przyjmować postać formuły matematycznej lub fragmentu kodu.
- OCL (ang. *Object Constraint Language*) – formalny język zapisu ograniczeń w UML.





Zrozumieć UML – podejście praktyczne

**PRZYKŁADY SCHEMATÓW  
APLIKACYJNYCH UML  
Z ROZPORZĄDZEŃ**

# Etapy budowy schematu aplikacyjnego UML

- Identyfikacja dziedziny problemowej oraz przegląd wymagań.
- Opracowanie modelu pojęciowego – identyfikacja:
  - typów obiektów,
  - ich właściwości,
  - związków między nimi,
  - ograniczeń.
- Zapisanie modelu w języku formalnym = schemat aplikacyjny UML.
- Integracja opracowanego schematu aplikacyjnego ze schematami znormalizowanymi, zdefiniowanymi w normach ISO serii 19100 (np. geometria i topologia, jakość, opis położenia).

# Oznaczenia na diagramach

## ■ Stereotypy:

- «**ApplicationSchema**» (pakiet) – schemat aplikacyjny,
- «**FeatureType**» (klasa) – typ obiektu przestrzennego,
- «**CodeList**» (klasa) – lista predefiniowanych wartości, którą można rozszerzyć,
- «**Enumeration**» (klasa) – stała lista predefiniowanych wartości (nie można jej rozszerzać),
- «**DataType**» (klasa) – definicja strukturalnego typu danych (posiada własne atrybuty),
- «**Union**» (klasa) – strukturalny typ danych, dla którego dokładnie jeden z atrybutów musi wystąpić.

# Stereotyp «voidable»

- Przypisany do atrybutu obiektu lub roli w powiązaniu, jeżeli:
  - pewna właściwość (cecha) obiektu przestrzennego nie jest prezentowana w zbiorze danych przestrzennych,
  - ale może być obecna lub mieć zastosowanie w świecie rzeczywistym.



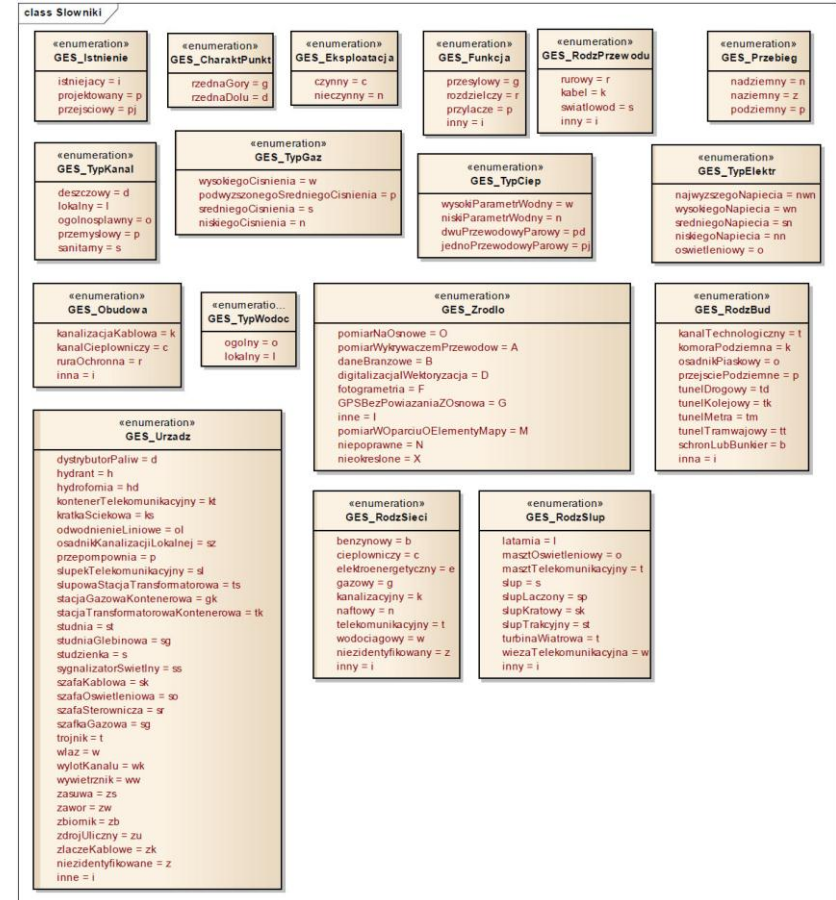
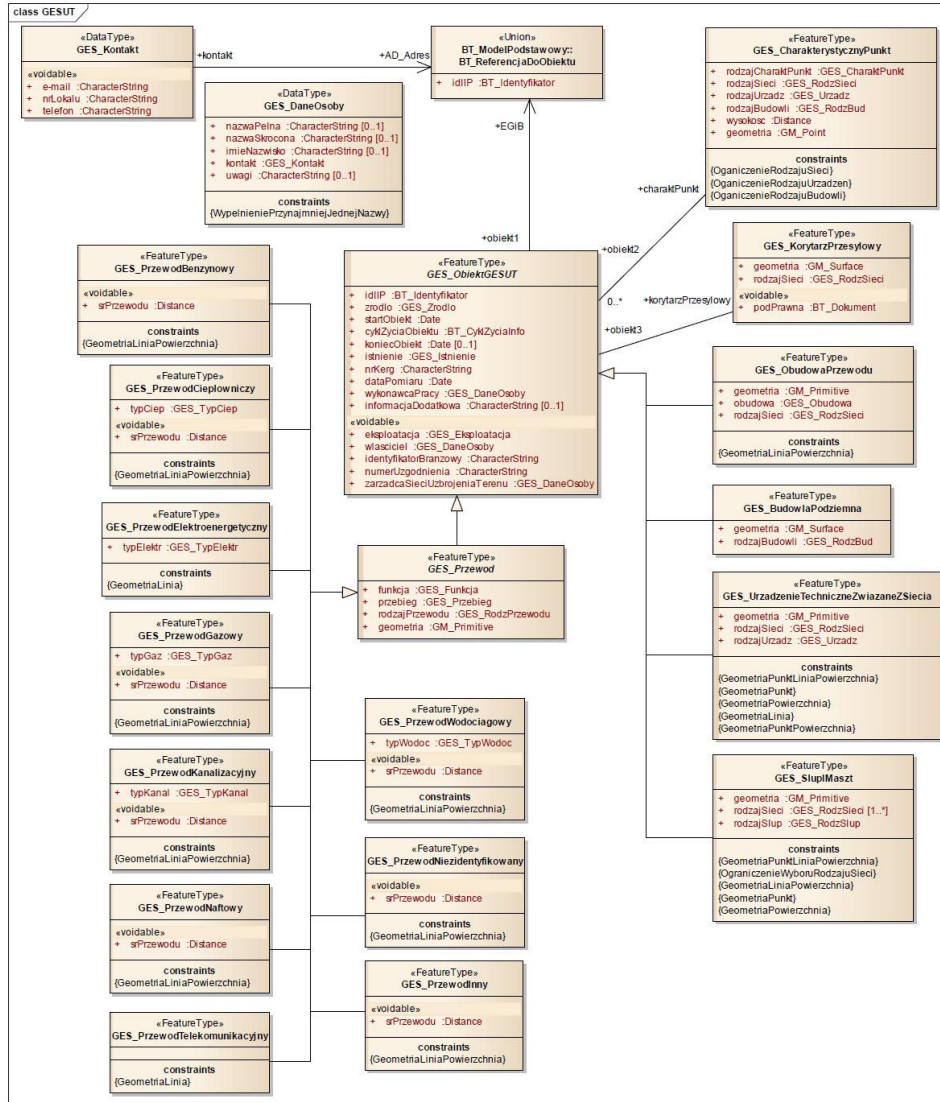
# Stereotyp «voidable»

- Atrybut specjalny może przyjmować następujące wartości:
  - *inapplicable* – (*nie stosuje się*) brak wartości, nie ma zastosowania,
  - *missing* – (*brak danych*) prawidłowa wartość nie jest łatwo dostępna, może nie istnieć,
  - *template* – (*tymczasowy brak danych*) wartość będzie dostępna później,
  - *unknown* – (*nieznany*) prawidłowa wartość nie jest znana, ale prawdopodobnie istnieje,
  - *withheld* – (*zastrzeżony*) wartość nie została ujawniona, jest zastrzeżona.

# Rozporządzenia

- Rozporządzenie MSWiA w sprawie **bazy danych geodezyjnej ewidencji sieci uzbrojenia terenu, bazy danych obiektów topograficznych oraz mapy zasadniczej**  
(projekt – 12.08.2011r.)

# Rozporządzenie GESUT i BDOT500



Schemat aplikacyjny UML

# Rozporządzenie GESUT i BDOT500

Klasa: GES PrzewodElektroenergetyczny	
<i>Nazwa:</i>	GES_SUPE
<i>Definicja:</i>	Przewód elektroenergetyczny.
<i>Klasa bazowa:</i>	GES_Przewod
<i>Stereotypy:</i>	«FeatureType»
<b>Atrybut:</b>	
<i>Nazwa:</i>	typElektr
<i>Nazwa (pełna):</i>	typ sieci elektroenergetycznej
<i>Dziedzina:</i>	GES_TypElektr
<i>Liczność:</i>	1
<i>Definicja:</i>	Typ sieci elektroenergetycznej.
<b>Relacja:</b>	
<i>Typ:</i>	Generalization
<i>Dziedzina:</i>	GES_Przewod
<b>Ograniczenie:</b>	
<i>Nazwa:</i>	GeometriaLinia
<i>Język naturalny:</i>	Geometria obiektu jest linią.
<i>OCL:</i>	inv: self.geometria.ocllsTypeOf(GM_Curve)=true



Katalog obiektów (fragment)

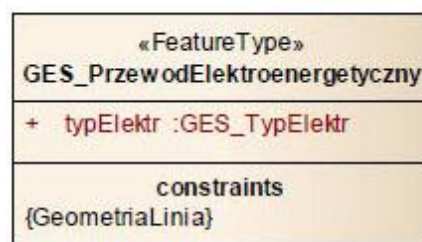


Diagram klas



# Rozporządzenie GESUT i BDOT500

```
- <element name="GES_PrzewodElektroenergetyczny" type="ges:GES_PrzewodElektroenergetycznyType"
  substitutionGroup="ges:GES_Przewod">
  - <annotation>
    <documentation>Przewód elektroenergetyczny.</documentation>
  </annotation>
</element>
- <complexType name="GES_PrzewodElektroenergetycznyType">
  - <complexContent>
    - <extension base="ges:GES_PrzewodType">
      - <sequence>
        - <element name="typElektr" type="ges:GES_TypElektrType">
          - <annotation>
            <documentation>Wartość ze słownika typu sieci elektroenergetycznej.</documentation>
          </annotation>
        </element>
      </sequence>
    </extension>
  </complexContent>
</complexType>
- <complexType name="GES_PrzewodElektroenergetycznyPropertyType">
  - <sequence minOccurs="0">
    <element ref="ges:GES_PrzewodElektroenergetyczny" />
  </sequence>
  <attributeGroup ref="gml:AssociationAttributeGroup" />
  <attributeGroup ref="gml:OwnershipAttributeGroup" />
</complexType>
```

Schemat aplikacyjny GML (fragment)



# Rozporządzenie GESUT i BDOT500

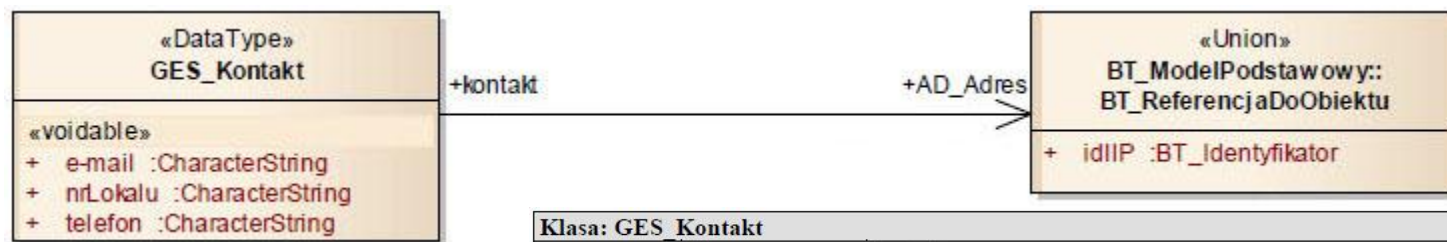


Diagram klas

Klasa: GES Kontakt	
<i>Nazwa:</i>	kontakt
<i>Definicja:</i>	Adres i dane kontaktowe.
<i>Stereotypy:</i>	«DataType»
<b>Atrybut:</b>	
<i>Nazwa:</i>	e-mail
<i>Nazwa (pełna):</i>	e-mail
<i>Dziedzina:</i>	CharacterString
<i>Liczność:</i>	1
<i>Definicja:</i>	Adres e-mail.
<i>Stereotypy:</i>	«voidable»
<b>Atrybut:</b>	
<i>Nazwa:</i>	nrLokalu
<i>Nazwa (pełna):</i>	numer lokalu
<i>Dziedzina:</i>	CharacterString
<i>Liczność:</i>	1
<i>Definicja:</i>	Numer lokalu.
<i>Stereotypy:</i>	«voidable»
<b>Atrybut:</b>	
<i>Nazwa:</i>	telefon
<i>Nazwa (pełna):</i>	telefon
<i>Dziedzina:</i>	CharacterString
<i>Liczność:</i>	1
<i>Definicja:</i>	Telefon.
<i>Stereotypy:</i>	«voidable»
<b>Relacja:</b>	
<i>Typ:</i>	Association
<i>Rola:</i>	AD_Adres
<i>Dziedzina:</i>	BT_ReferencjaDoObiektu
<i>Liczność:</i>	1
<i>Definicja:</i>	Referencja do EMUiA w zakresie wszystkich atrybutów pozyskanych z klasy AD_Adres.

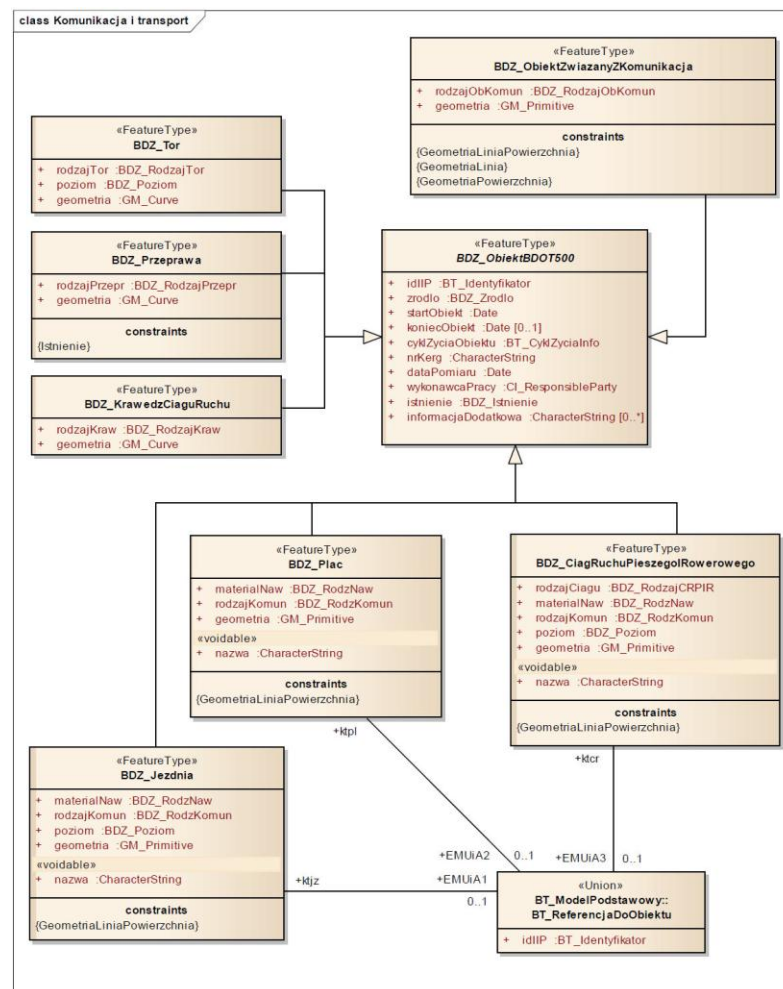
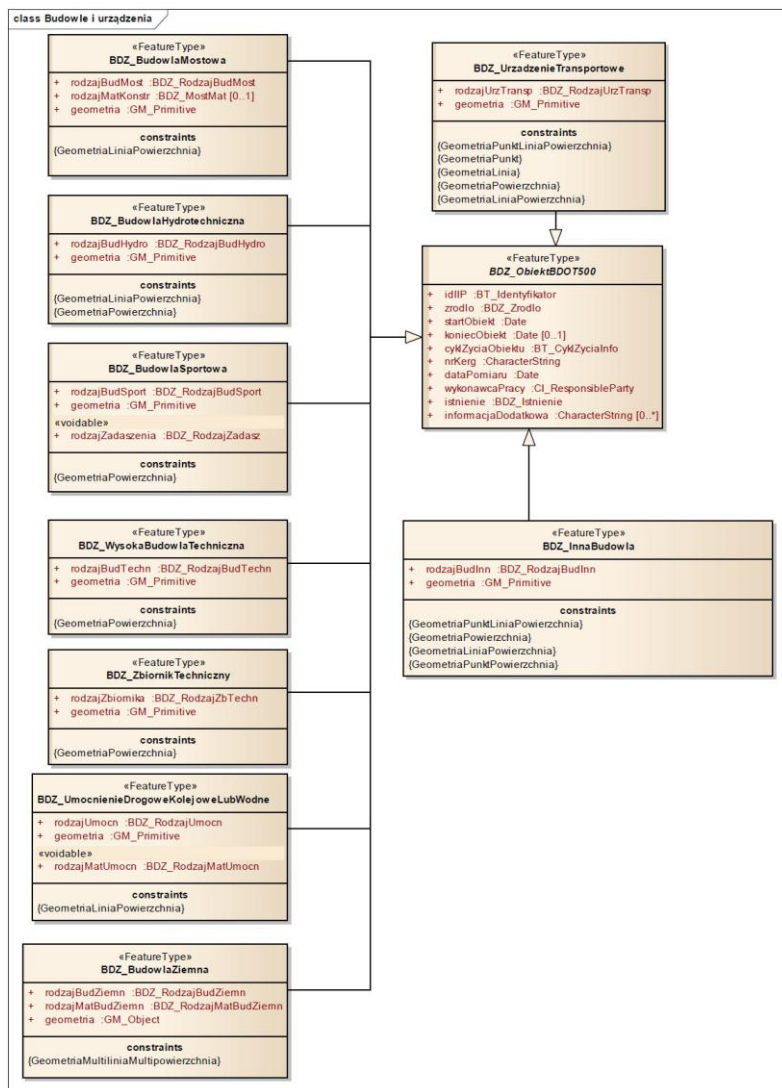
Katalog obiektów (fragment)

# Rozporządzenie GESUT i BDOT500

```
<element name="GES_Kontakt" type="ges:GES_KontaktType" substitutionGroup="gml:AbstractObject" />
- <complexType name="GES_KontaktType">
- <sequence>
- <element name="e-mail">
- <annotation>
  <documentation>E-mail.</documentation>
</annotation>
- <complexType>
- <simpleContent>
- <extension base="string">
  <attribute ref="gco:nilReason" />
</extension>
</simpleContent>
</complexType>
</element>
- <element name="nrLokalu">
- <annotation>
  <documentation>Numer lokalu.</documentation>
</annotation>
- <complexType>
- <simpleContent>
- <extension base="string">
  <attribute ref="gco:nilReason" />
</extension>
</simpleContent>
</complexType>
</element>
- <element name="telefon">
- <annotation>
  <documentation>Telefon.</documentation>
</annotation>
- <complexType>
- <simpleContent>
- <extension base="string">
  <attribute ref="gco:nilReason" />
</extension>
</simpleContent>
</complexType>
</element>
<!-- Roles -->
<element name="AD_Adres" type="bt:BT_ReferencjaDoObiektuPropertyType" />
</sequence>
</complexType>
- <complexType name="GES_KontaktPropertyType">
- <sequence minOccurs="0">
  <element ref="ges:GES_Kontakt" />
</sequence>
<attributeGroup ref="gml:OwnershipAttributeGroup" />
</complexType>
```

Schemat aplikacyjny GML (fragment)

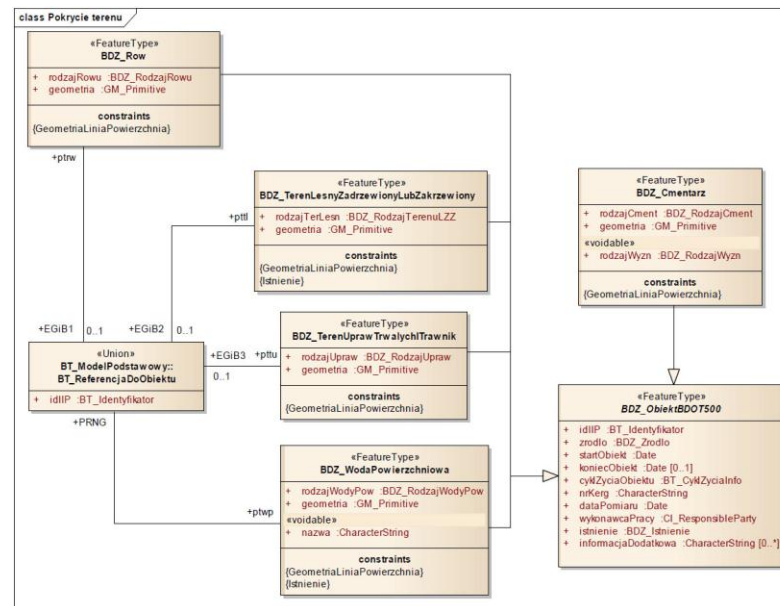
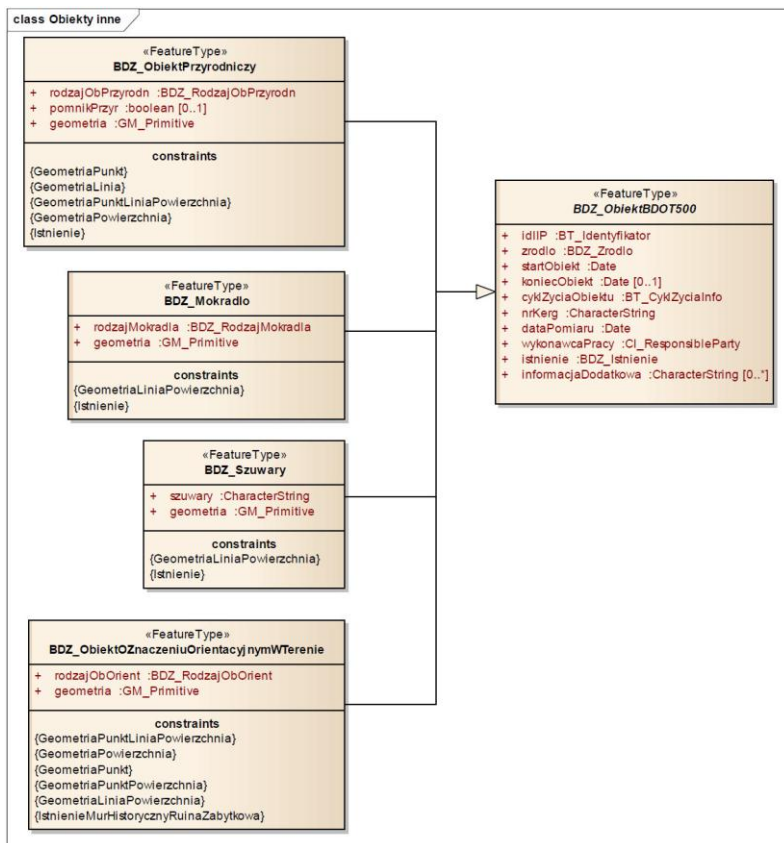
# Rozporządzenie GESUT i BDOT500



Schemat aplikacyjny UML (fragment)

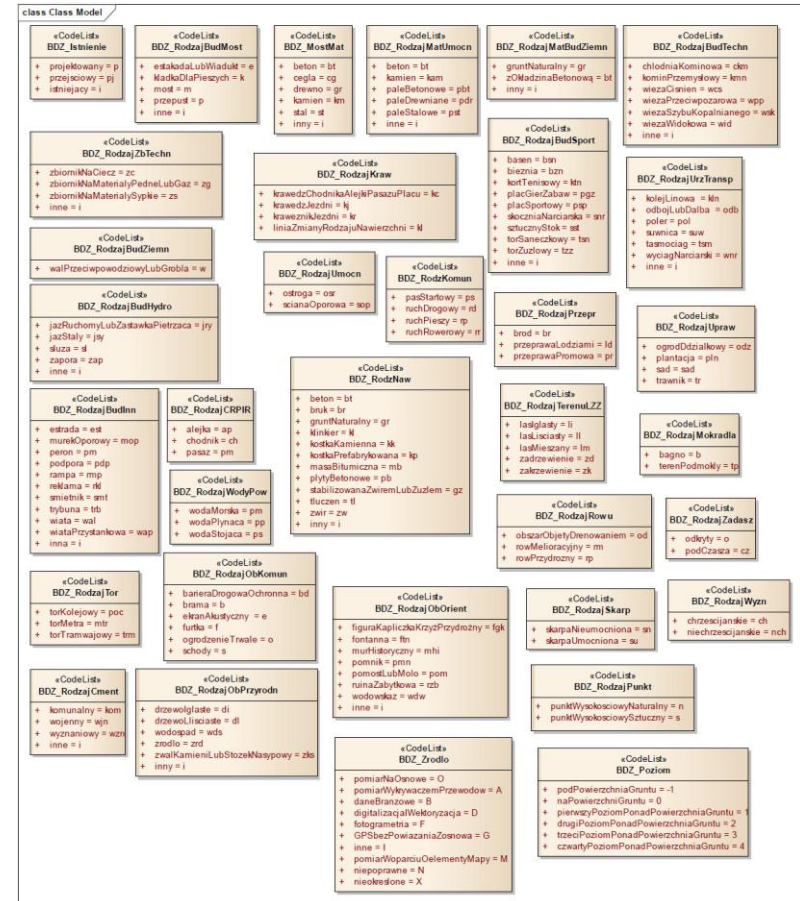
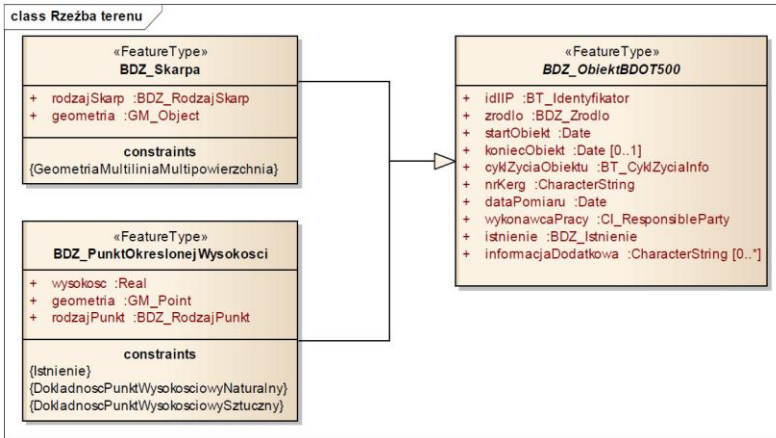


# Rozporządzenie GESUT i BDOT500



Schemat aplikacyjny UML (fragment)

# Rozporządzenie GESUT i BDOT500



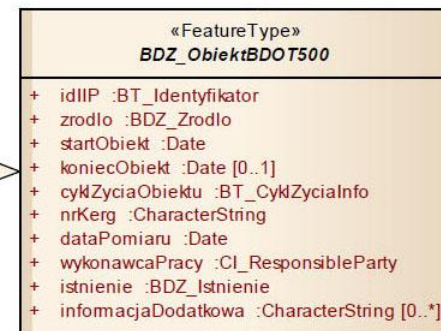
Schemat aplikacyjny UML (fragment)

# Rozporządzenie GESUT i BDOT500

<b>Klasa:BDZ_BudowlaMostowa</b>	
<i>Nazwa:</i>	BDZ_BUBM
<i>Definicja:</i>	Rodzaje przepraw w postaci budowli inżynierskich, których konstrukcja pozwala pokonywać przeszkody wodne i lądowe, np. mosty, estakady, wiadukty, kładki oraz przepusty.
<i>Klasa bazowa:</i>	BDZ_ObjektBDOT500
<i>Stereotypy:</i>	«FeatureType»
<b>Atrybut:</b>	
<i>Nazwa:</i>	rodzajBudMost
<i>Nazwa (pełna):</i>	rodzaj budowli mostowych
<i>Dziedzina:</i>	BDZ_RodzajBudMost
<i>Liczność:</i>	1
<i>Definicja:</i>	Rodzaj budowli mostowych.
<b>Atrybut:</b>	
<i>Nazwa:</i>	rodzajMatKonstr
<i>Nazwa (pełna):</i>	rodzaj materiału konstrukcyjnego
<i>Dziedzina:</i>	BDZ_MostMat
<i>Liczność:</i>	0..1
<i>Definicja:</i>	Rodzaj materiału konstrukcyjnego, z którego wykonane są budowle mostowe.
<b>Atrybut:</b>	
<i>Nazwa:</i>	geometria
<i>Nazwa (pełna):</i>	geometria
<i>Dziedzina:</i>	GM_Primitive
<i>Liczność:</i>	1
<i>Definicja:</i>	Geometria obiektu.
<b>Relacja:</b>	
<i>Typ:</i>	Generalization
<i>Dziedzina:</i>	BDZ_ObjektBDOT500
<b>Ograniczenie:</b>	
<i>Nazwa:</i>	GeometriaLiniaPowierzchnia
<i>Język naturalny:</i>	Geometria budowli mostowej jest linią, powierzchnią.
<i>OCL:</i>	inv: self.geometria.ocIsTypeOf(GM_Curve)=true or self.geometria.ocIsTypeOf(GM_Surface)=true



Diagram klas



Katalog obiektów (fragment)

# Rozporządzenie GESUT i BDOT500

```
- <element name="BDZ_BUBM" type="bdz:BDZ_BUBMType" substitutionGroup="bdz:BDZ_ObiektBDOT500">
  - <annotation>
    <documentation>Rodzaje przepraw w postaci budowli inżynierskich, których konstrukcja pozwala pokonywać
    przeszkody wodne i lądowe, np. mosty, estakady, wiadukty, kładki oraz przepusty.</documentation>
  </annotation>
</element>
- <complexType name="BDZ_BUBMType">
  - <complexContent>
    - <extension base="bdz:BDZ_ObiektBDOT500Type">
      - <sequence>
        - <element name="rodzajBudMost" type="bdz:BDZ_RodzajBudMostType">
          - <annotation>
            <documentation>Rodzaj budowli mostowych.</documentation>
          </annotation>
        </element>
        - <element name="rodzajMatKonstr" type="bdz:BDZ_MostMatType" minOccurs="0">
          - <annotation>
            <documentation>Rodzaj materiału konstrukcyjnego z którego wykonane są budowle
            mostowe.</documentation>
          </annotation>
        </element>
        - <element name="geometria" type="gml:GeometricPrimitivePropertyType">
          - <annotation>
            <documentation>Reprezentacją geometryczną budowli mostowej jest powierzchnia.
            Reprezentacją geometryczną przepustu węższego od 1,0 m jest linia, będącą jego osią
            geometryczną.</documentation>
          </annotation>
        </element>
      </sequence>
    </extension>
  </complexContent>
</complexType>
```

Schemat aplikacyjny GML (fragment)



# Rozporządzenie GESUT i BDOT500

```

- <simpleType name="BDZ_RodzajBudMostType">
  <union memberTypes="bdz:BDZ_RodzajBudMostEnumerationType bdz:BDZ_RodzajBudMostOtherType" />
</simpleType>
- <simpleType name="BDZ_RodzajBudMostEnumerationType">
  - <restriction base="string">
    <enumeration value="e" />
    <enumeration value="k" />
    <enumeration value="m" />
    <enumeration value="p" />
    <enumeration value="i" />
  </restriction>
</simpleType>
- <simpleType name="BDZ_RodzajBudMostOtherType">
  - <restriction base="string">
    <pattern value="other: \w{2,}" />
  </restriction>
</simpleType>

```

Schemat aplikacyjny GML (fragment)

«CodeList» BDZ_RodzajBudMost
+ estakadaLubWiadukt = e
+ kladkaDlaPieszych = k
+ most = m
+ przepust = p
+ inne = i

Diagram klas

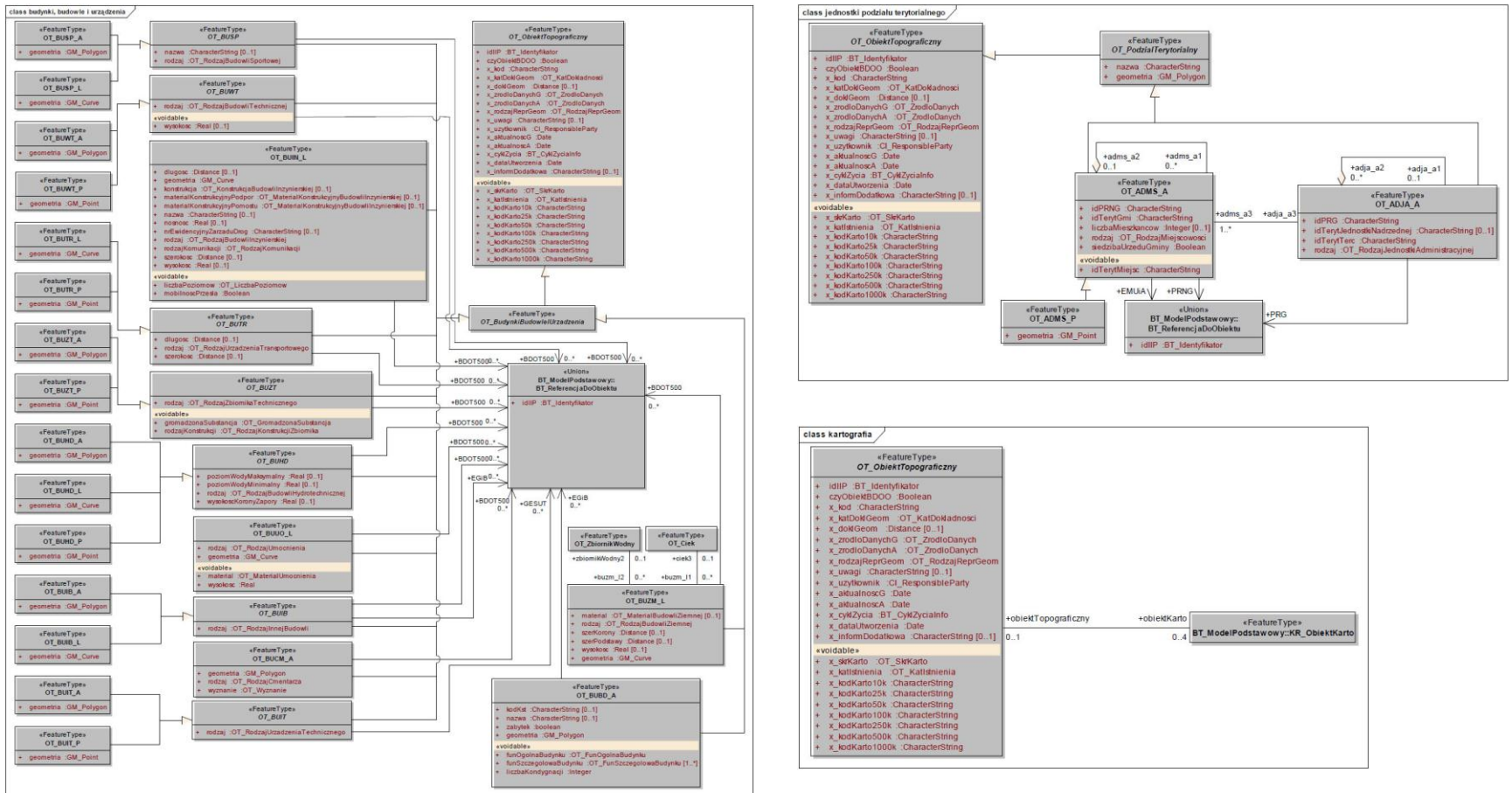
Klasa: BDZ_RodzajBudMost		
	<i>Nazwa:</i>	rodzaj budowli mostowej
	<i>Definicja:</i>	Słownik rodzajów budowli mostowych.
	<i>Stereotypy:</i>	«CodeList»
<b>Atrybut:</b>	<i>Nazwa:</i>	estakadaLubWiadukt
	<i>Nazwa (pełna):</i>	estakada lub wiadukt - e
	<i>Definicja:</i>	
<b>Atrybut:</b>	<i>Nazwa:</i>	inne
	<i>Nazwa (pełna):</i>	inne - i
	<i>Definicja:</i>	
<b>Atrybut:</b>	<i>Nazwa:</i>	kladkaDlaPieszych
	<i>Nazwa (pełna):</i>	kladka dla pieszych - k
	<i>Definicja:</i>	
<b>Atrybut:</b>	<i>Nazwa:</i>	most
	<i>Nazwa (pełna):</i>	most - m
	<i>Definicja:</i>	
<b>Atrybut:</b>	<i>Nazwa:</i>	przepust
	<i>Nazwa (pełna):</i>	przepust - p
	<i>Definicja:</i>	

Katalog obiektów (fragment)

# Rozporządzenia

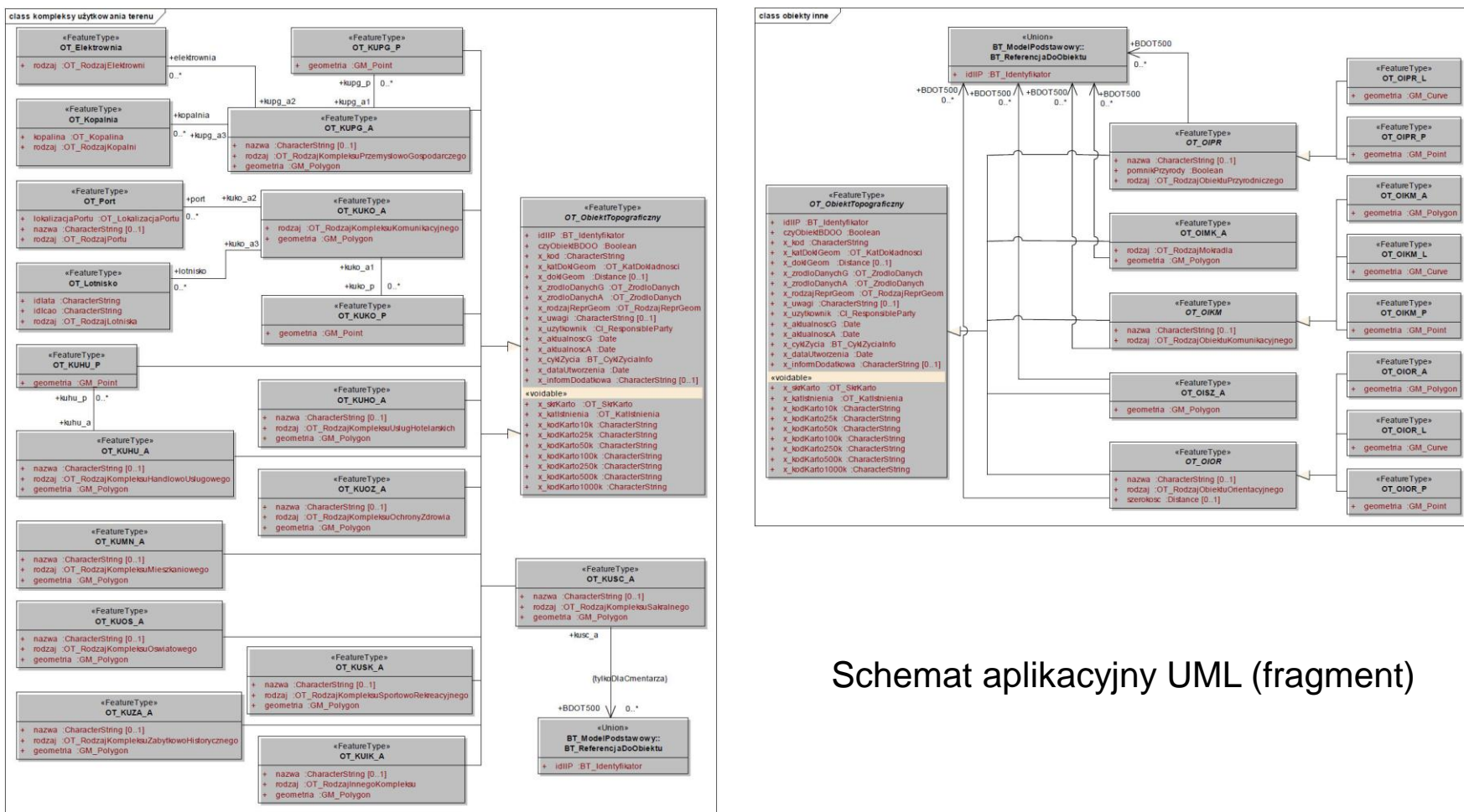
- Rozporządzenie MSWiA w sprawie **bazy danych obiektów topograficznych oraz bazy danych obiektów ogólnogeograficznych, a także standardowych opracowań kartograficznych** (projekt – 25.08.2011r.)

# Rozporządzenie BDOT10k i BDOO



Schemat aplikacyjny UML (fragment)

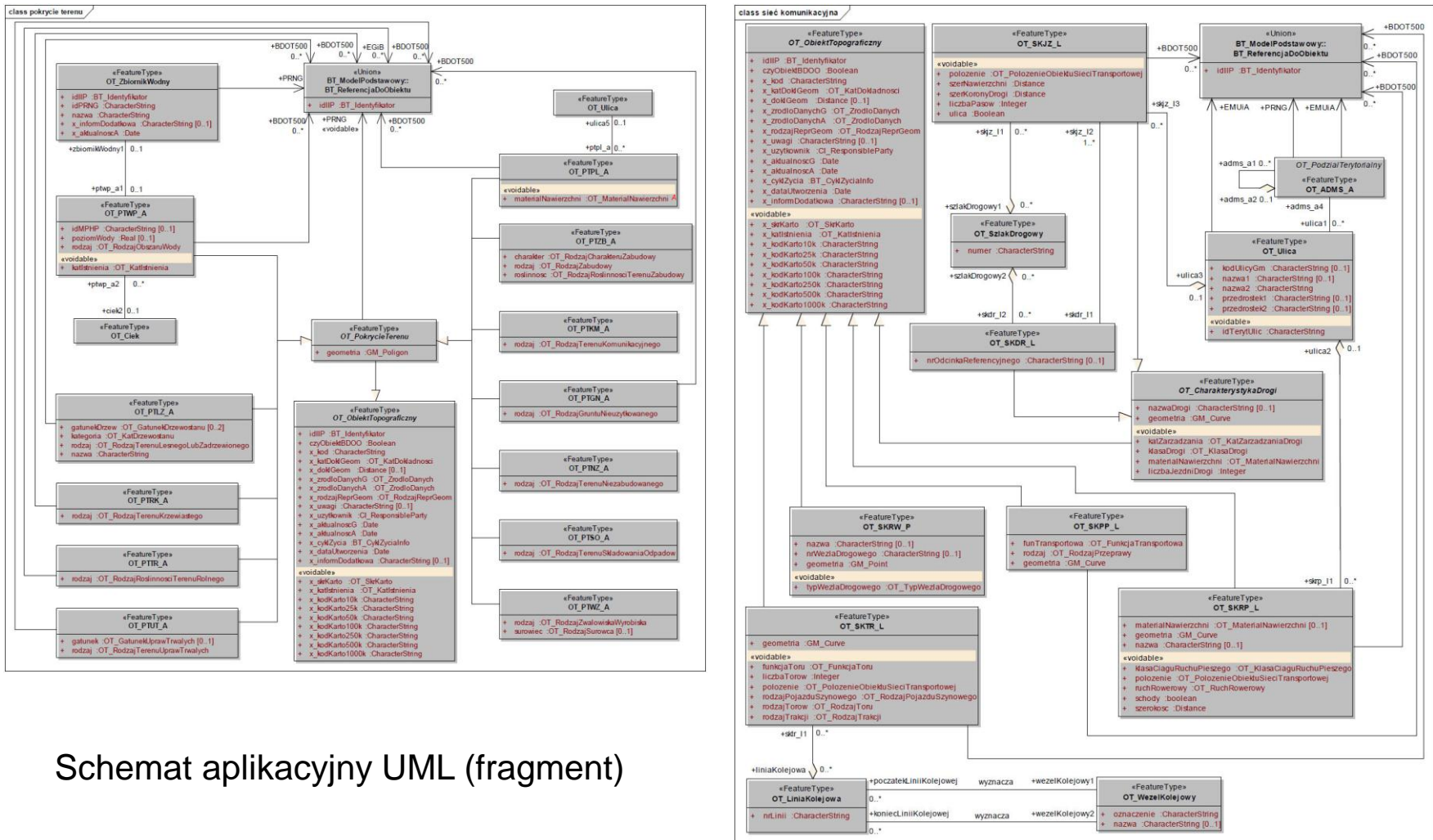
# Rozporządzenie BDOT10k i BDOO



Schemat aplikacyjny UML (fragment)

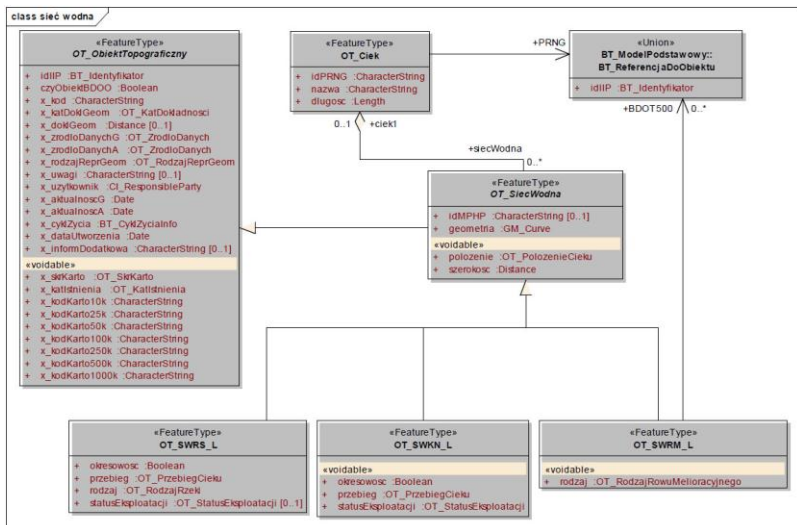
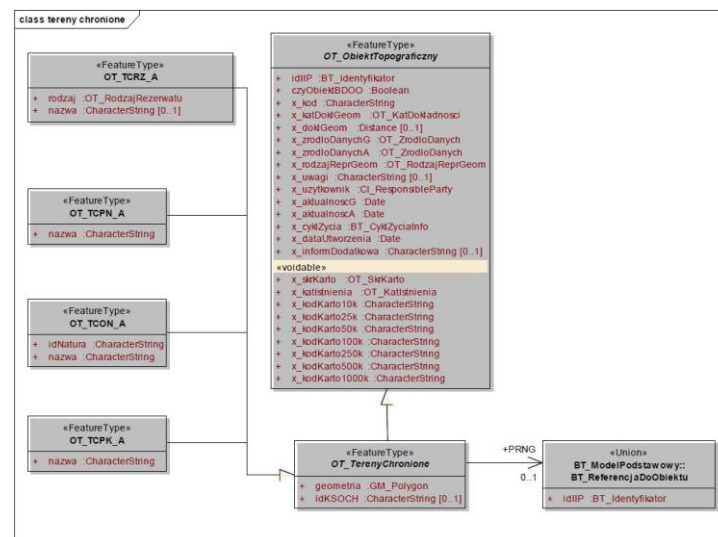
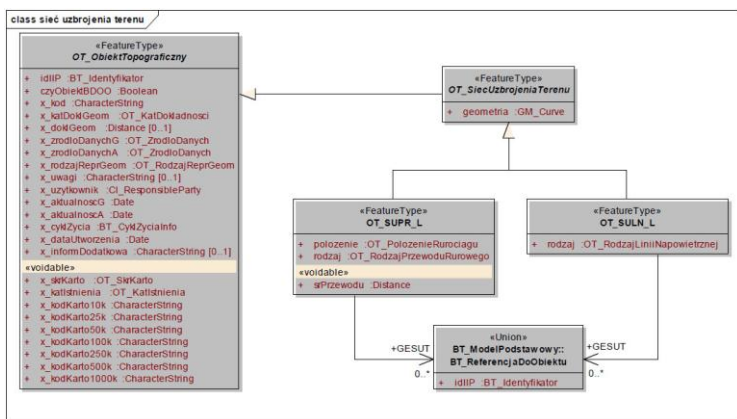


# Rozporządzenie BDOT10k i BDOO



Schemat aplikacyjny UML (fragment)

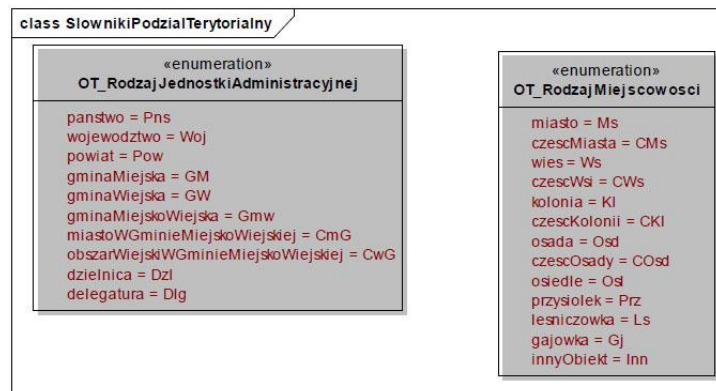
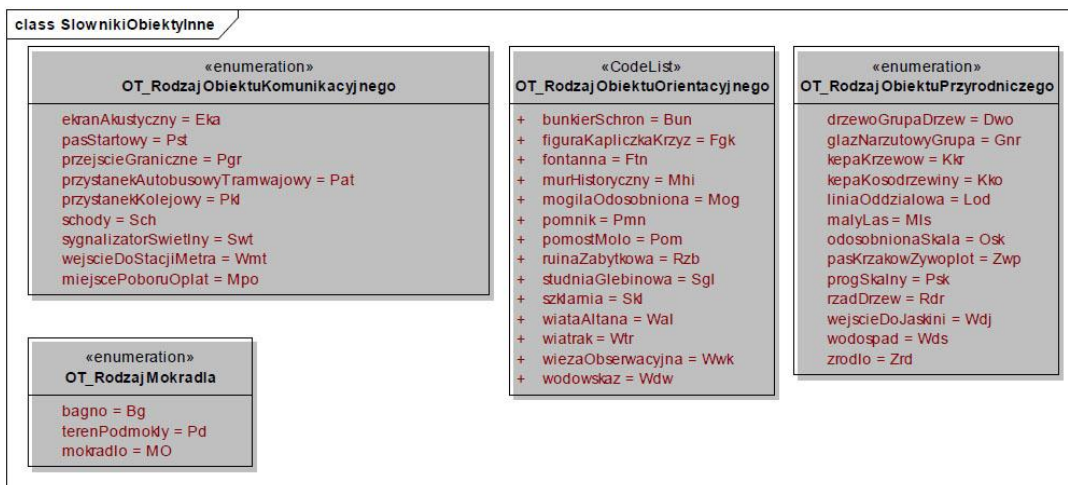
# Rozporządzenie BDOT10k i BDOO



Schemat aplikacyjny UML (fragment)

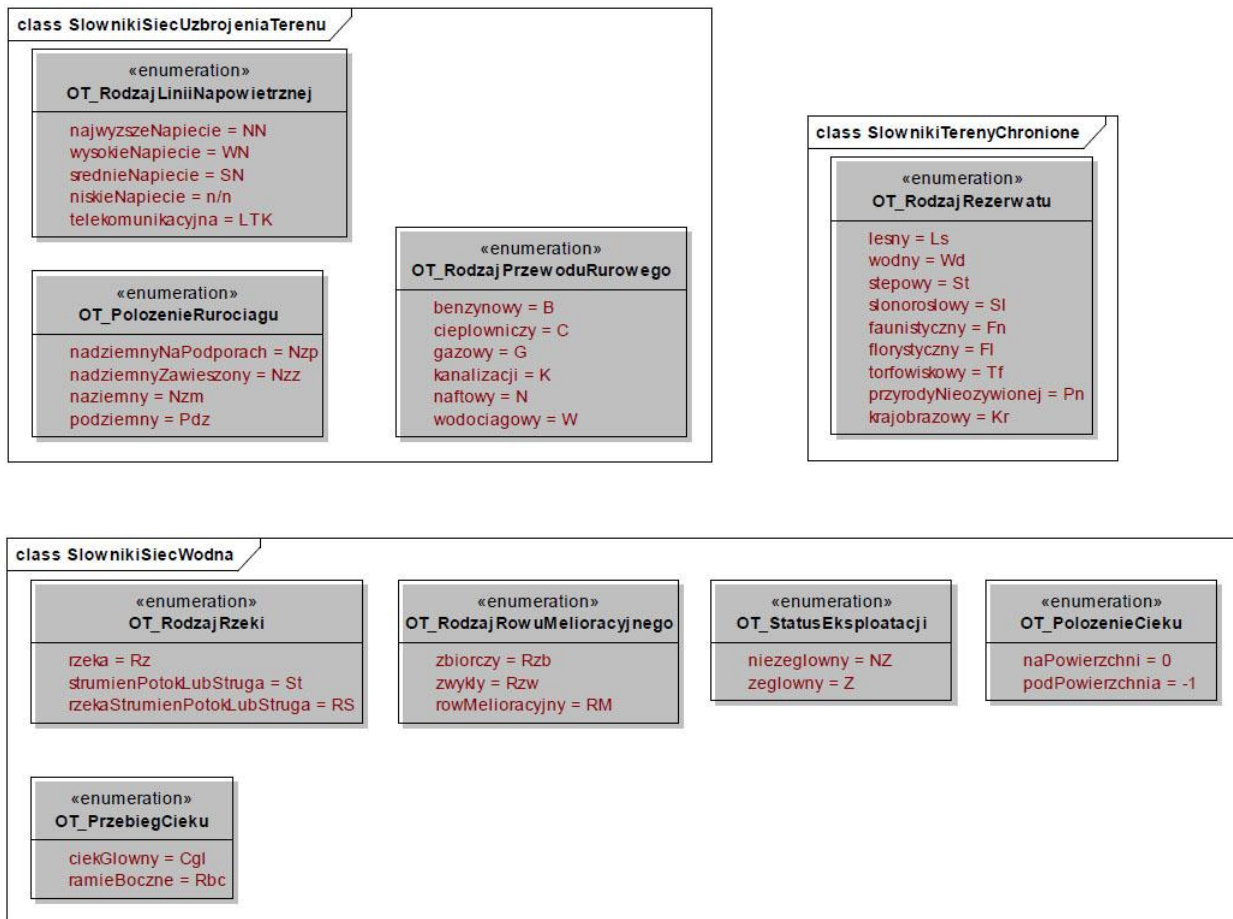


# Rozporządzenie BDOT10k i BDOO



Schemat aplikacyjny UML (fragment)

# Rozporządzenie BDOT10k i BDOO



Schemat aplikacyjny UML (fragment)

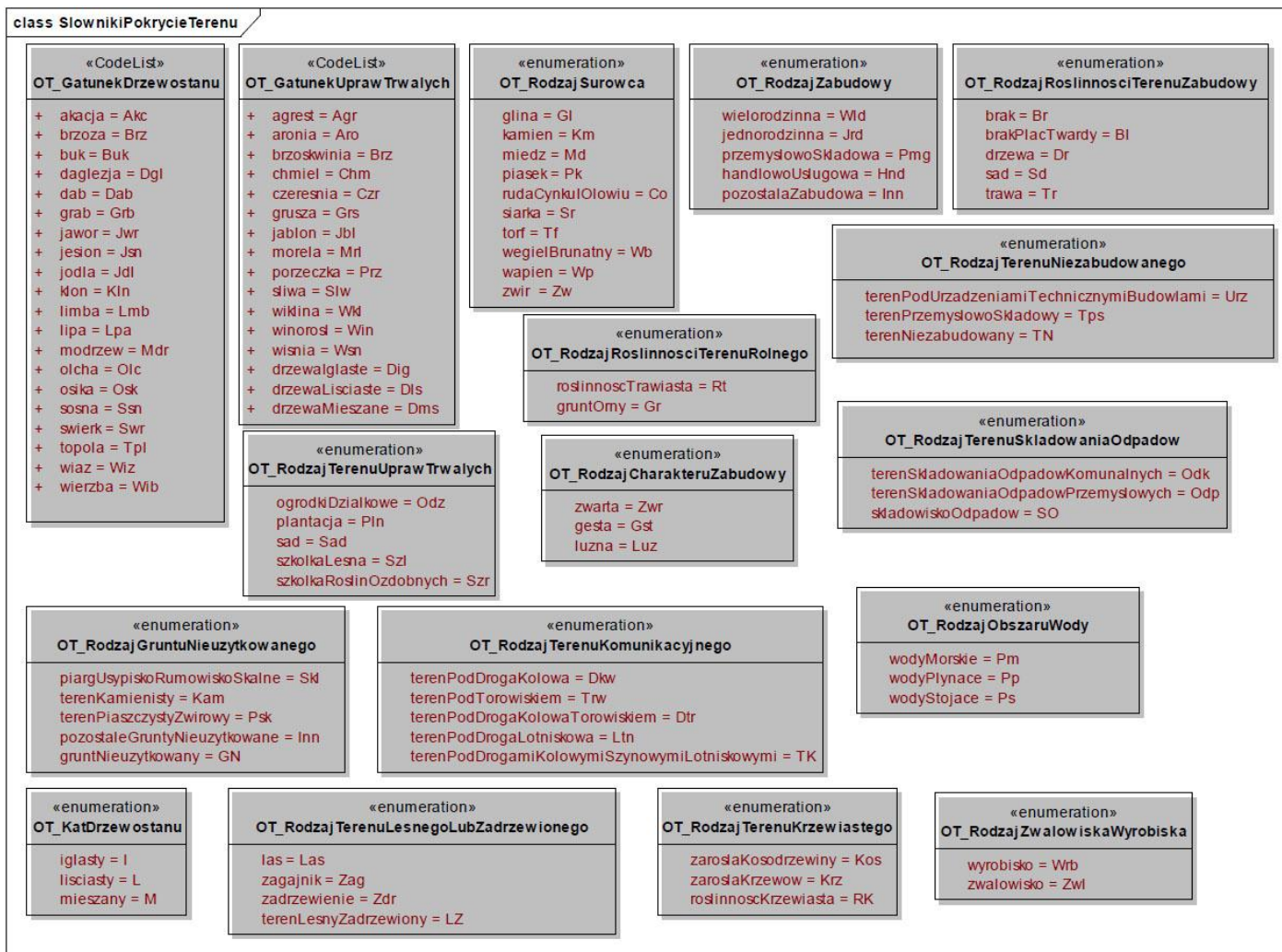


# Rozporządzenie BDOT10k i BDOO



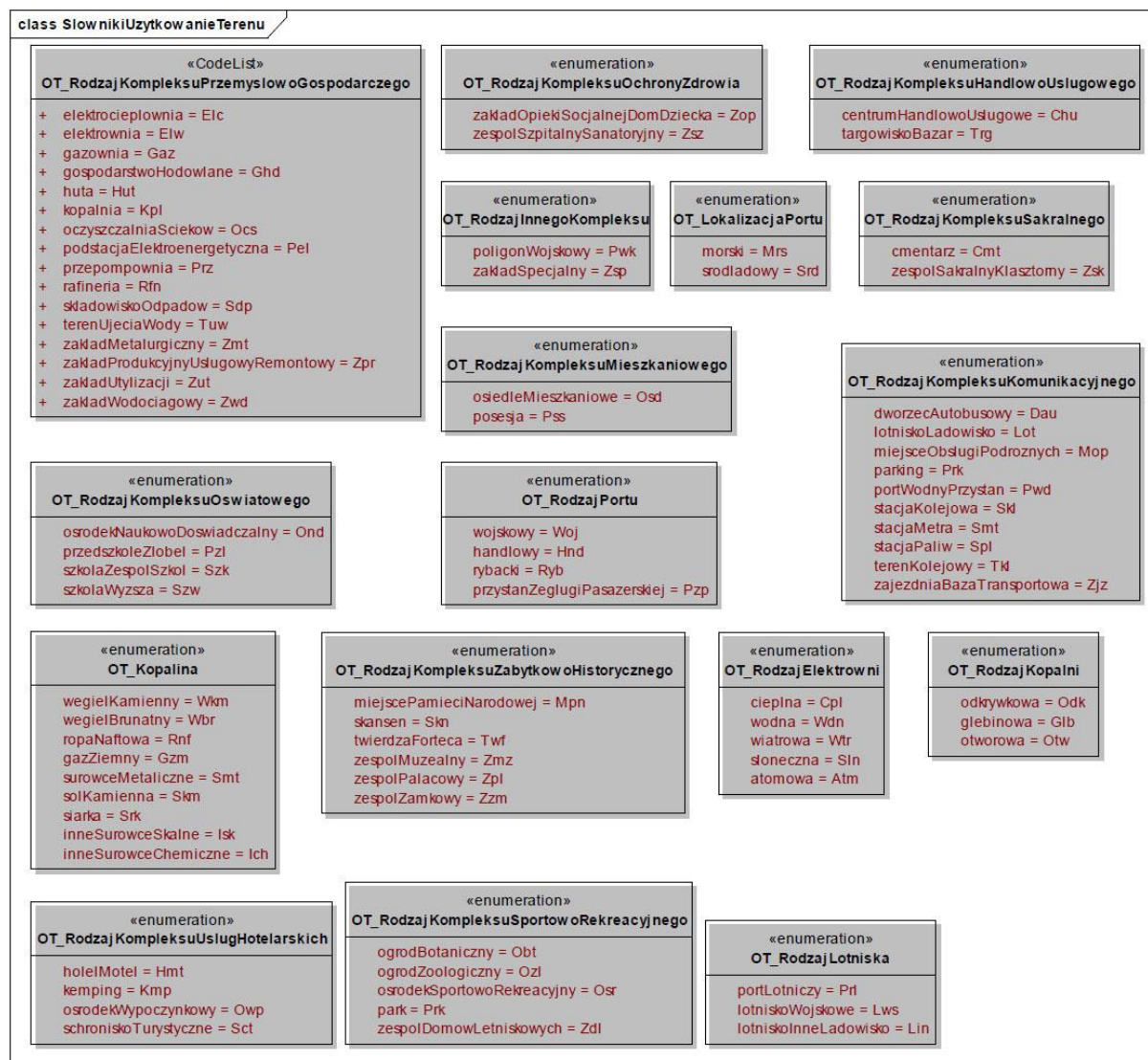
Schemat aplikacyjny UML (fragment)

# Rozporządzenie BDOT10k i BDOO



Schemat aplikacyjny UML (fragment)

# Rozporządzenie BDOT10k i BDOO

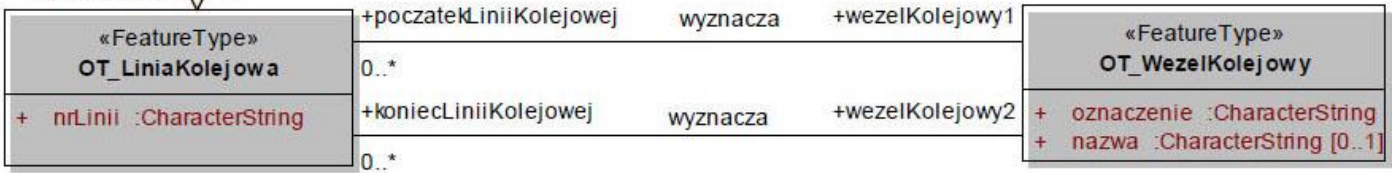
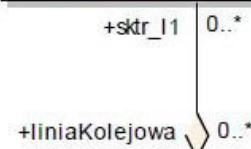
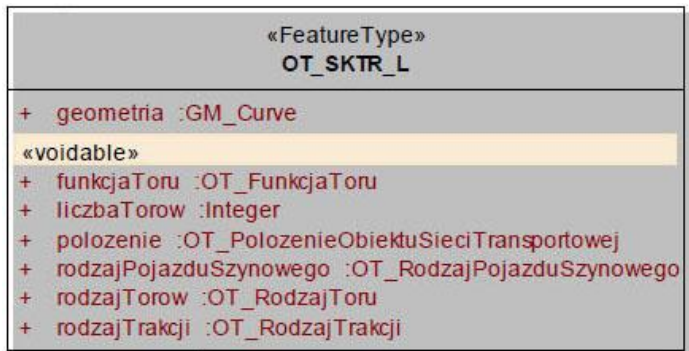


Schemat aplikacyjny UML (fragment)



# Rozporządzenie BDOT10k i BDOO

Katalog obiektów (fragment)



<b>Klasa: OT_WezelKolejowy</b>		
<i>Nazwa:</i>	wykaz węzłów kolejowych	
<i>Definicja:</i>	Typ danych wyliczeniowych: WYKAZ.	
<i>Stereotypy:</i>	«FeatureType»	
<b>Atrybut:</b>	<i>Nazwa:</i>	oznaczenie
	<i>Nazwa (pełna):</i>	oznaczenie
	<i>Dziedzina:</i>	CharacterString
	<i>Liczność:</i>	1
	<i>Definicja:</i>	Oznaczenie węzła kolejowego.
<b>Atrybut:</b>	<i>Nazwa:</i>	nazwa
	<i>Nazwa (pełna):</i>	nazwa
	<i>Dziedzina:</i>	CharacterString
	<i>Liczność:</i>	0..1
	<i>Definicja:</i>	Nazwa węzła kolejowego.
<b>Relacja:</b>	<i>Typ:</i>	Association
	<i>Rola:</i>	poczatekLiniiKolejowej
	<i>Dziedzina:</i>	OT_LiniaKolejowa
	<i>Liczność:</i>	0..*
	<i>Definicja:</i>	Obiekty klasy OT_LiniaKolejowa są powiązane z obiektami klasy OT_WezelKolejowy.
<b>Relacja:</b>	<i>Typ:</i>	Association
	<i>Rola:</i>	koniecLiniiKolejowej
	<i>Dziedzina:</i>	OT_LiniaKolejowa
	<i>Liczność:</i>	0..*
	<i>Definicja:</i>	Obiekty klasy OT_LiniaKolejowa są powiązane z obiektami klasy OT_WezelKolejowy.
<b>Ograniczenie:</b>	<i>Nazwa:</i>	Atrybut nazwa.
	<i>Język naturalny:</i>	Pole atrybutu nazwa przechowuje maksymalnie 255 znaków.
	<i>OC:</i>	inv: self.nazwa.size()<=255

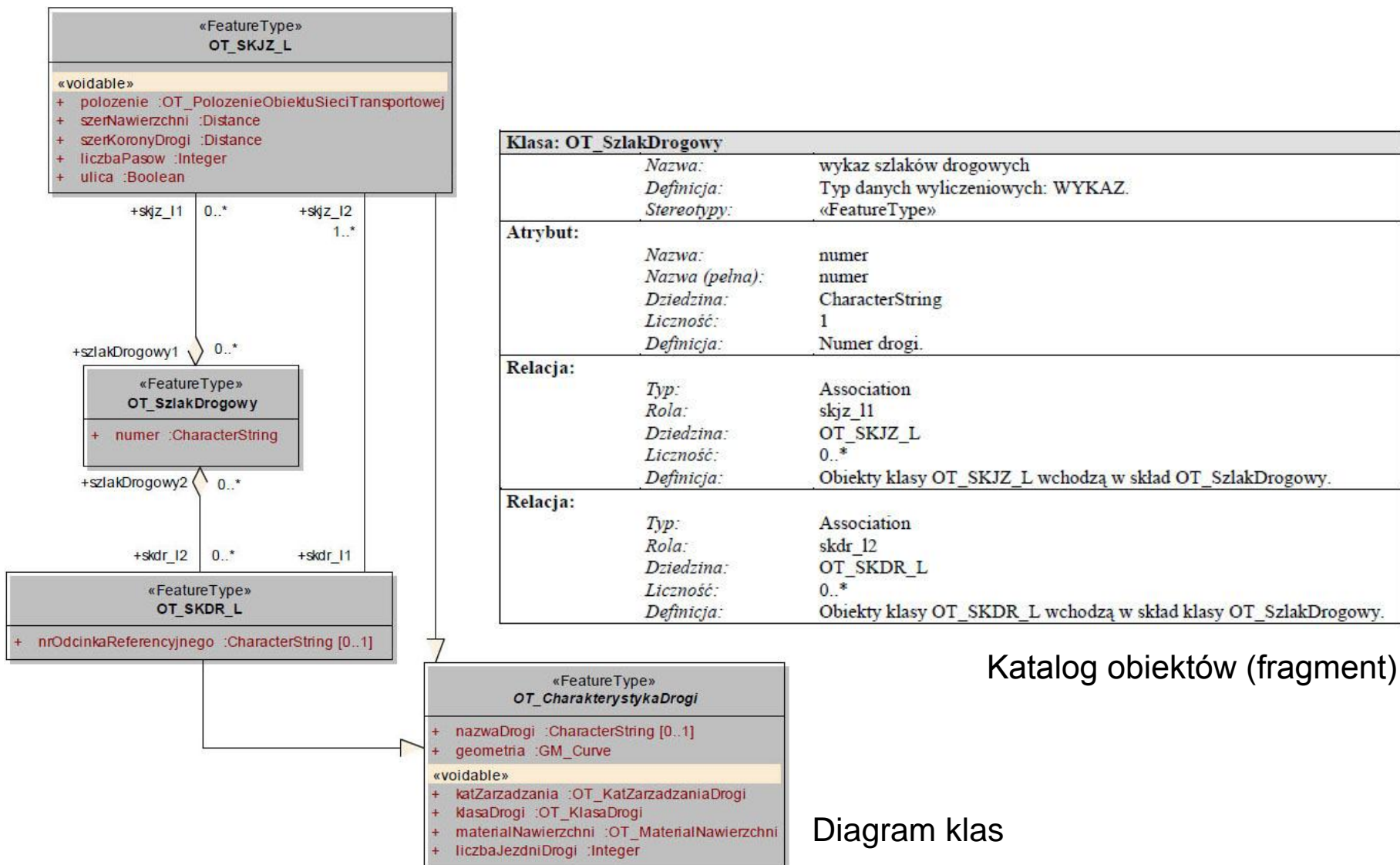
Diagram klas

# Rozporządzenie BDOT10k i BDOO

```
<element name="OT_WezelKolejowy" type="ot:OT_WezelKolejowyType" substitutionGroup="gml:AbstractFeature" />
- <complexType name="OT_WezelKolejowyType">
- <complexContent>
- <extension base="gml:AbstractFeatureType">
- <sequence>
  <element name="oznaczenie" type="string" />
  <element name="nazwa" type="string" minOccurs="0" />
  <!-- Roles -->
- <element name="początekLiniiKolejowej" type="ot:OT_LiniaKolejowaPropertyType" minOccurs="0"
  maxOccurs="unbounded">
- <annotation>
- <appinfo>
  <gml:reversePropertyName>ot:wezelKolejowy1</gml:reversePropertyName>
  </appinfo>
</annotation>
</element>
- <element name="koniecLiniiKolejowej" type="ot:OT_LiniaKolejowaPropertyType" minOccurs="0"
  maxOccurs="unbounded">
- <annotation>
- <appinfo>
  <gml:reversePropertyName>ot:wezelKolejowy2</gml:reversePropertyName>
  </appinfo>
</annotation>
</element>
</sequence>
</extension>
</complexContent>
</complexType>
- <complexType name="OT_WezelKolejowyPropertyType">
- <sequence minOccurs="0">
  <element ref="ot:OT_WezelKolejowy" />
</sequence>
<attributeGroup ref="gml:AssociationAttributeGroup" />
<attributeGroup ref="gml:OwnershipAttributeGroup" />
</complexType>
```

Schemat aplikacyjny GML (fragment)

# Rozporządzenie BDOT10k i BDOO



Klasa: OT_SzlakDrogowy		
Nazwa:	wykaz szlaków drogowych	
Definicja:	Typ danych wyliczeniowych: WYKAZ.	
Stereotypy:	«FeatureType»	
Atrybut:		
Nazwa:	numer	
Nazwa (pełna):	numer	
Dziedzina:	CharacterString	
Liczność:	1	
Definicja:	Numer drogi.	
Relacja:		
Typ:	Association	
Rola:	skjz_11	
Dziedzina:	OT_SKJZ_L	
Liczność:	0..*	
Definicja:	Obiekty klasy OT_SKJZ_L wchodzą w skład OT_SzlakDrogowy.	
Relacja:		
Typ:	Association	
Rola:	skdr_12	
Dziedzina:	OT_SKDR_L	
Liczność:	0..*	
Definicja:	Obiekty klasy OT_SKDR_L wchodzą w skład klasy OT_SzlakDrogowy.	

Katalog obiektów (fragment)

Diagram klas

# Rozporządzenie BDOT10k i BDOO

```
<element name="OT_SzlakDrogowy" type="ot:OT_SzlakDrogowyType" substitutionGroup="gml:AbstractFeature" />
- <complexType name="OT_SzlakDrogowyType">
- <complexContent>
- <extension base="gml:AbstractFeatureType">
- <sequence>
  <element name="numer" type="string" />
  <!-- Roles -->
- <element name="skjz_I1" type="ot:OT_SKJZ_LPropertyType" minOccurs="0" maxOccurs="unbounded">
- <annotation>
- <appinfo>
  <gml:reversePropertyName>ot:szlakDrogowy1</gml:reversePropertyName>
  </appinfo>
</annotation>
</element>
- <element name="skdr_I2" type="ot:OT_SKDR_LPropertyType" minOccurs="0" maxOccurs="unbounded">
- <annotation>
- <appinfo>
  <gml:reversePropertyName>ot:szlakDrogowy2</gml:reversePropertyName>
  </appinfo>
</annotation>
</element>
</sequence>
</extension>
</complexContent>
</complexType>
- <complexType name="OT_SzlakDrogowyPropertyType">
- <sequence minOccurs="0">
  <element ref="ot:OT_SzlakDrogowy" />
</sequence>
<attributeGroup ref="gml:AssociationAttributeGroup" />
<attributeGroup ref="gml:OwnershipAttributeGroup" />
</complexType>
```

Schemat aplikacyjny GML (fragment)



# Rozporządzenie BDOT10k i BDOO

```

- <simpleType name="OT_RodzajKomunikacjiType">
- <restriction base="string">
- <enumeration value="dr">
- <annotation>
- <appinfo>
  <gml:description>drogowa</gml:description>
  </appinfo>
</annotation>
</enumeration>
- <enumeration value="kl">
- <annotation>
- <appinfo>
  <gml:description>kolejowa</gml:description>
  </appinfo>
</annotation>
</enumeration>
- <enumeration value="tr">
- <annotation>
- <appinfo>
  <gml:description>tramwajowa</gml:description>
  </appinfo>
</annotation>
</enumeration>
</restriction>
</simpleType>

```

Schemat aplikacyjny GML  
(fragment)

Diagram klas



Katalog obiektów (fragment)

Klasa: OT_RodzajKomunikacji		
	<i>Nazwa:</i>	rodzaj komunikacji
	<i>Definicja:</i>	Rodzaj komunikacji na moście i wiadukcie.
	<i>Stereotypy:</i>	«enumeration»
<b>Atrybut:</b>	<i>Nazwa:</i>	drogowa
	<i>Nazwa (pełna):</i>	drogowa - dr
	<i>Definicja:</i>	Drogowa.
<b>Atrybut:</b>	<i>Nazwa:</i>	kolejowa
	<i>Nazwa (pełna):</i>	kolejowa - kl
	<i>Definicja:</i>	Kolejowa.
<b>Atrybut:</b>	<i>Nazwa:</i>	tramwajowa
	<i>Nazwa (pełna):</i>	tramwajowa - tr
	<i>Definicja:</i>	Tramwajowa.

# Podsumowanie

- Publikowanie danych dla poszczególnych tematów danych przestrzennych
  - zgodnie ze specyfikacjami danych przestrzennych oraz przepisami wykonawczymi do dyrektywy INSPIRE to jeden z warunków osiągnięcia interoperacyjności.
- Specyfikacje danych przestrzennych są zapisywane w języku UML i określają ramy dla zharmonizowanych zbiorów dotyczących każdego z tematów INSPIRE.



konieczna znajomość UML

# Podsumowanie

- **Opracowywanie** standardów danych przestrzennych
- **Tworzenie** zbiorów/baz danych przestrzennych
- **Korzystanie** ze standardów danych przestrzennych



**konieczna znajomość UML**

UML to uniwersalny środek formalny

- zalecany przez normy ISO serii 19100 do budowy schematów aplikacyjnych
- wykorzystywany do opracowywania specyfikacji danych INSPIRE

# Materiały uzupełniające

- *Booch G., Rumbaugh J., Jacobson I. 2002. UML. Przewodnik użytkownika. WNT, Warszawa.*
- *MDA Guide Version 1.0, 2003. OMG. <http://www.omg.org/mda>*
- *prCEN/TR 15449, 2006. Geographic information – Standards, specifications, technical reports and guidelines, required to implement Spatial Data Infrastructure.*
- *ISO/TS 19103:2005 Geographic information – Conceptual schema language*
- *ISO 19109:2009 Geographic information – Rules for application schema*
- *ISO/DIS 19118:2005 Geographic information – Encoding*
- *ISO 19136:2007 Geographic information – Geography Markup Language (GML)*
- *ISO/TS 19139:2007 Geographic information – Metadata – XML schema implementation*
- *Object Management Group (OMG), UML. <http://www.uml.org>*
- *Portal UML. Projekty systemów informatycznych z UML. <http://www.uml.com.pl>*
- *Schmuller J. 2003. UML dla każdego. Helion, Gliwice.*





# Dziękuję za uwagę!!!

- **dr inż. Agnieszka Chojka**  
agnieszka.chojka@uwm.edu.pl

Katedra Geodezji Szczegółowej  
Wydział Geodezji i Gospodarki Przestrzennej  
Uniwersytet Warmińsko-Mazurski w Olsztynie