POLSKIE TOWARZYSTWO INFORMACJI PRZESTRZENNEJ
ROCZNIKI GEOMATYKI 2014 ❍ Tom XII ❍ Zeszyt 2(64): 197–204

ZŁOŻONOŚĆ SCHEMATÓW APLIKACYJNYCH UML I GML

COMPLEXITY OF UML AND GML APPLICATION SCHEMAS

Agnieszka Chojka

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

Słowa kluczowe: GML, UML, schemat aplikacyjny, złożoność
Keywords: GML, UML, application schema, complexity

Wprowadzenie

Implementacja dyrektywy INSPIRE w Polsce oraz budowa Krajowej Infrastruktury Informacji Przestrzennej, spowodowały znaczny wzrost zainteresowania udostępnianiem danych przestrzennych i związanych z nimi usług, zwłaszcza przez organy publiczne i interesariuszy prywatnych. Zaowocowało to wieloma inicjatywami mającymi na celu harmonizację różnych zbiorów danych przestrzennych, a więc zapewnienie ich spójności logicznej i semantycznej.

Proces harmonizacji wymaga, albo opracowania nowych struktur danych, albo dostosowania już istniejących struktur danych przestrzennych do wytycznych i zaleceń INSPIRE. Struktury danych zapisywane są w postaci schematów aplikacyjnych UML i GML (XML Schema). Błędne lub zbyt złożone zapisy struktur danych, mają bezpośredni wpływ na możliwość generowania plików GML z konkretnymi danymi (obiektami), a tym samym mogą być przyczyną różnych problemów i anomalii na etapie produkcji danych.

Przedmiotem badań jest dokonanie pomiaru złożoności schematów aplikacyjnych UML i GML, opracowanych w Głównym Urzędzie Geodezji i Kartografii (GUGiK) w zakresie prac związanych z implementacją dyrektywy INSPIRE w Polsce. Zakłada się także dokonanie analizy istniejących miar złożoności struktur zapisanych w językach UML i XML Schema, zbadanie możliwości wykorzystania różnych narzędzi do zmierzenia złożoności struktur zapisanych w obu językach, a także zaproponowanie nowych, bardziej optymalnych metryk dla tych potrzeb.

Schematy aplikacyjne UML i GML

Wprowadzenie w życie postanowień dyrektywy INSPIRE w Polsce – uchwalenie ustawy *o infrastrukturze informacji przestrzennej*, spowodowało między innymi konieczność nowelizacji ustawy *prawo geodezyjne i kartograficzne* oraz zmiany związanych z nią rozporządzeń. Integralną częścią tych rozporządzeń są schematy aplikacyjne UML oraz schematy aplikacyjne GML.

Niewątpliwą zaletą schematów opracowanych w GUGiK jest przede wszystkim to, że definiują one spójną i jednorodną w skali całego kraju, strukturę informacyjną baz danych, właściwych dla danego rozporządzenia. Co więcej, do ich opracowania wykorzystano normy ISO serii 19100 w dziedzinie informacji geograficznej, aby w przyszłości zapewnić interoperacyjność w zakresie danych przestrzennych.

Jednakże, podczas tworzenia tych schematów napotkano wiele problemów technicznych, związanych z przekształceniem UML na GML. Po opublikowaniu rozporządzeń, również wykonawcy zgłosili wiele uwag dotyczących schematów aplikacyjnych UML i GML, w tym między innymi wskazali wady, błędy oraz pewne nieprawidłowości w zapisach tych schematów. Jedną z przyczyn zaistniałej sytuacji jest niejednoznaczność transformacji UML-GML (Chojka, 2013). Inną przyczyną może być zbyt duża złożoność opracowanych schematów aplikacyjnych UML i GML, co z kolei może mieć istotny wpływ na możliwość generowania plików GML z konkretnymi danymi (obiektami), ale również na możliwość przetworzenia i obsługi takich plików przez oprogramowanie GIS.

W związku z powyższym, zdaniem autorki warto obliczyć złożoność schematów aplikacyjnych UML i GML, zawartych w rozporządzeniach opracowanych przez GUGiK, aby na tej podstawie zaproponować ich optymalizację i podnieść jakość, zarówno samych schematów aplikacyjnych, jak również baz danych tworzonych na ich podstawie.

Miary złożoności

W informatyce zastosowanie mają metryki oprogramowania. Są to miary pewnych własności oprogramowania lub jego specyfikacji. Miara złożoności strukturalnej jest jedną z najistotniejszych miar, pozwalających oszacować nie tylko jakość samego oprogramowania jako produktu końcowego, ale również śledzić złożoność poszczególnych komponentów składowych systemu, we wszystkich fazach procesu jego tworzenia. Jednym z takich komponentów jest model informacyjny systemu, na który składają się opisy struktur danych na przykład w postaci diagramów klas UML.

W literaturze przedmiotu można znaleźć wiele publikacji dotyczących kwestii pomiaru złożoności, zarówno diagramów klas UML (np. Genero, Piattini, Calero, 2005; Kang, Xu, Lu, Chu, 2004; Kim, Boldyreff, 2002), jak i struktur zapisanych w języku XML Schema (np. Lämmel, Kitsis, Remy, 2005; Manso, Genero, Piattini, 2003; McDowell, Schmidt, Yue, 2005). Zwykle takie miary stanowią wypadkową różnych metryk, na przykład w przypadku modeli UML będą to metryki charakteryzujące pojedyncze klasy lub też związki między nimi (Kang, Xu, Lu, Chu, 2004).

Poniżej dokonano krótkiego przeglądu, najciekawszych zdaniem autorki, metryk złożoności, które można wykorzystać do pomiaru entropii schematów aplikacyjnych UML i GML, opracowanych w GUGiK.

Złożoność UML

Kompleksowego zestawienia różnych metryk złożoności diagramów klas UML dokonano w publikacji *A Survey of Metrics for UML Class Diagrams* (Genero, Piattini, Calero, 2005). Jej autorzy uwzględnili nie tylko metryki zaprojektowane na potrzeby badania entropii diagramów klas, ale również przedstawili inne miary złożoności, które przetestowali na diagramach UML. Dodatkowo w artykule omówiono kwestie które należy rozważyć podczas opracowywania nowych metryk złożoności.

Interesującego porównania typowych metryk diagramów klas UML dokonali autorzy pracy *A Comparison of Metrics for UML Class Diagrams* (Yi, Wu, Gan, 2004). W badaniach wykorzystano 6 różnych metryk, które następnie przetestowano na 27 diagramach klas UML, pochodzących z bankowego systemu informacji.

Wśród typowych metryk złożoności diagramów klas można wyróżnić dwie grupy metryk (Manso, Genero, Piattini, 2003): metryki wielkości (*size metrics*) oraz metryki złożoności strukturalnej (*structural complexity metrics*). Do metryk koncentrujących się na rozmiarze diagramów klas należą:

- NC (*Number of Classes*) – całkowita liczba klas,
- NA (*Number of Attributes*) – całkowita liczba atrybutów,
- NM (*Number of Methods*) – całkowita liczba metod.

Metryki złożoności strukturalnej reprezentują następujące miary:

- NAssoc (*Number of Associations*) – całkowita liczba powiązań (asocjacji),
- NAgg (*Number of Aggregations*) – całkowita liczba związków agregacji,
- NDep (*Number of Dependencies*) – całkowita liczba związków zależności,
- NGen (*Number of Generalisations*) – całkowita liczba związków generalizacji,
- NGenH (*Number of Generalization hierarchies*) – całkowita liczba hierarchii dziedziczenia między klasami,
- MaxDIT (*Maximum DIT*) – wartość DIT jest liczona dla każdej klasy, jest to ścieżka od danej klasy do „korzenia" w ramach hierarchii dziedziczenia,
- MaxHAgg (*Maximum HAgg*) – wartość HAgg jest liczona dla każdej klasy, jest to ścieżka od danej klasy do „liścia" w ramach hierarchii agregacji.

Do tej grupy metryk można także zaliczyć metryki (Vargas, Nugroho, Chaudron, Visser, 2012):

- AscNoRole (*Associations Without Role*) – liczba powiązań bez nazwanych ról,
- LoneClass (*Lonely Classes*) – liczba klas, które nie są w żaden sposób powiązane z innymi klasami (dana klasa na diagramie nie jest w związku z inną klasą oraz nie posiada atrybutu typu inna klasa).

Złożoność XML Schema

Najciekawsze i wyczerpujące podejście do kwestii badania złożoności schematów zapisanych w języku XML Schema, przedstawia pozycja *Analysis of XML schema usage* (Lämmel, Kitsis, Remy, 2005), w której zdefiniowano kompletny zbiór metryk dla XML Schema. Innym interesującym opracowaniem w tym zakresie jest publikacja *Analysis and Metrics of*

XML Schema (McDowell, Schmidt, Yue, 2005), w której autorzy zdefiniowali aż 11 różnych metryk pomiaru jakości i złożoności struktur XML Schema.

Metryki złożoności XML Schema można podzielić na 3 kategorie: *XML-agnostic*, *XSD-agnostic* oraz *XSD-aware* (Lämmel, Kitsis, Remy, 2005). Metryki *XML-agnostic* nie uwzględniają żadnych informacji powiązanych z XML. W tej kategorii znalazły się następujące metryki:

- KB – wielkość wszystkich plików XSD, które należą do jednego schematu, mierzona w kilobajtach (KB),
- LOC (*Lines of Code*) – całkowita ilość linii kodu danego schematu.

W grupie metryk *XSD-agnostic*, które dotyczą zależności związanych z XML, można wyróżnić metryki:

- #NODE – liczba wszystkich węzłów XML (atrybuty i elementy),
- #ANN – liczba węzłów "annotation" w XML.

Trzecie grupa metryk, *XSD-aware*, koncentruje się na strukturze (najważniejszych blokach konstrukcyjnych) plików XSD. W tej kategorii znalazły się:

- $\#EL_g$ – liczba deklaracji elementów globalnych,
- $\#CT_g$ – liczba definicji globalnych typów złożonych (*complex-types*),
- $\#ST_g$ – liczba definicji globalnych typów prostych (*simple-types*),
- $\#MG_g$ – liczba definicji grup modeli globalnych (*model-group*),
- $\#AG_g$ – liczba deklaracji grup atrybutów globalnych (*attribute-group*),
- $\#AT_g$ – liczba deklaracji atrybutów globalnych,
- #GLOBAL – suma wszystkich powyższych elementów globalnych.

Warto w tym miejscu wspomnieć o jeszcze jednej, nieco bardziej wyszukanej metryce złożoności struktur zapisanych w XML Schema – metryce C(XSD) (Basci, Misra, 2009). Uwzględnia ona strukturę schematów XML, w przeciwieństwie do wyżej wymienionych metryk, które ograniczają się jedynie do zliczenia poszczególnych komponentów składowych schematów. Metryka C(XSD) kładzie szczególny nacisk na wykorzystanie struktur rekurencyjnych, które mogą być przyczyną złożoności schematów (Tamayo, Granell, Huerta, 2011). Wartość złożoności lub waga złożoności schematu stanowi sumę wag wyliczanych dla każdego komponentu schematu, według wzoru (Basci, Misra, 2009):

$$C(XSD) = C(V_g) + C(G_g) + C(T_g),$$

gdzie:

$C(V_g)$ – całkowita wartość (suma) złożoności wszystkich elementów i atrybutów globalnych, które mogą być załączone lub zaimportowane z zewnętrznych schematów XSD lub zadeklarowane/zdefiniowane w danym pliku XSD,

$C(G_g)$ – całkowita wartość (suma) złożoności elementów i atrybutów globalnych, które mogą być zadeklarowane/zdefiniowane w danym pliku XSD i nie posiadają żadnych powiązań (referencji) z innymi elementami w danym schemacie XSD,

$C(T_g)$ – całkowita wartość (suma) złożoności definicji/deklaracji globalnych typów złożonych i prostych (wbudowanych i zdefiniowanych przez użytkownika) nie posiadających żadnych powiązań z innymi elementami w danym schemacie XSD.

Narzędzia programowe

Badanie złożoności dokumentów, zapisanych w językach UML i XML Schema, nie jest zagadnieniem nowym. Złożoność oprogramowania nurtuje projektantów systemów informatycznych już od dawna, ponieważ ma ona istotny wpływ na samą realizację systemu, ale również na późniejsze utrzymanie takiego rozwiązania. Zdaniem DeMarco, inżyniera oprogramowania, nie można kontrolować tego, czego nie da się zmierzyć (DeMarco, 1986).

Istnieje wiele narzędzi, które usprawniają liczenie entropii oprogramowania. Ciekawe zestawienie takich aplikacji przedstawili autorzy artykułu *Comparing Software Metrics Tools* (Lincke, Lundberg, Löwe, 2008). W literaturze przedmiotu można także znaleźć rozwiązania dedykowane modelom UML oraz strukturom zapisanym w języku XML.

Przykładem oprogramowania, które pozwala oszacować jakość modeli UML, jest aplikacja *SDMetrics* (SDMetrics, 2014). Mierzy ona takie właściwości struktur zapisanych w UML jak: ich rozmiar, złożoność, powiązania. Sprawdza także reguły projektowe, na przykład wzajemne zależności między elementami modelu, czy stosowaną konwencję nazewnictwa.

W publikacji *Developing Software Metrics Applicable to UML Models* (Kim, Boldyreff, 2002) autorzy zaproponowali alternatywne narzędzie *UML Metrics Producer*, oparte na oprogramowaniu *Rational Rose* i pozwalające liczyć różne metryki dla diagramów UML.

Autorzy pracy *Analysis and Metrics of XML Schema* (McDowell, Schmidt, Yue, 2005), wykorzystując platformę open-source *Castor* (Castor, 2014) opracowali własny analizator, pozwalający obliczać złożoność dokumentów XML Schema za pomocą różnych metryk.

Zdaniem autorki, do obliczania złożoności schematów aplikacyjnych UML i GML, możliwe jest również wykorzystanie funkcjonalności dostępnych w narzędziach GIS. Zarówno struktury danych zapisane w UML jak i XML Schema można przedstawić za pomocą grafów (np. Kang, Xu, Lu, Chu, 2004) i wówczas przeprowadzić na nich różne analizy sieciowe.

Analiza złożoności schematów aplikacyjnych

Analizie poddano tylko te schematy aplikacyjne z rozporządzeń, które zostały udostępnione na stronie GUGiK (*http://www.gugik.gov.pl/prawo/schematy-aplikacyjne*) w postaci plików EAP (UML) oraz plików XSD (GML). Pominięto schematy dotyczące „Modelu Podstawowego", ponieważ w każdym z rozporządzeń schemat ten jest nieco inaczej zdefiniowany. Przy pomiarze złożoności poszczególnych schematów aplikacyjnych nie uwzględniono klas pochodzących z innych schematów (np. z „Modelu Podstawowego"), ale za to uwzględniono referencje do tych klas.

W tabelach zestawiono wyniki przeprowadzonej analizy złożoności wybranych schematów aplikacyjnych UML (tab. 1) oraz GML (tab. 2) przy wykorzystaniu przykładowych metryk opisanych w rozdziale „Miary złożoności". Do obliczenia złożoności nie zastosowano żadnego narzędzia przeznaczonego do tego celu. Analizę diagramów UML przeprowadzono „ręcznie" w aplikacji *Enterprise Architect*, zaś analizę plików XSD w programie *Notepad++*.

W obu tabelach zaznaczono maksymalne i minimalne wartości dla poszczególnych metryk. Uzyskane wyniki pozwalają jednoznacznie stwierdzić, że najbardziej złożonym schematem aplikacyjnym UML spośród przebadanych schematów, uwzględniając wartości wszystkich metryk, jest schemat opisujący strukturę bazy danych Ewidencji Gruntów i Budynków (EGiB). Najmniej skomplikowaną strukturą danych charakteryzuje się baza danych dla Mapy Zasadniczej (MZ).

Tabela 1. Złożoność schematów aplikacyjnych UML obliczona za pomocą wybranych metryk

SA UML	Metryki UML				
	NC	NA	NAssoc	NGen	LoneClass
EGiB	71	699	78	30	38
RCiWN	20	158	10	6	13
PRG	10	71	4	5	3
EMUiA	15	83	9	1	10
BDOT	60	244	3	27	32
GESUT	36	182	4	17	17
MZ	1	0	7	0	0
SytWys	12	45	17	0	6
Osnowa	29	180	9	11	10

Tabela 2. Złożoność schematów aplikacyjnych GML obliczona za pomocą wybranych metryk

SA GML	Metryki XML Schema				
	KB	LOC	#NODE	#CTg	#STg
EGiB	154	5053	317	76	34
RCiWN	35,1	1152	71	14	13
PRG	22,8	572	41	16	9
EMUiA	18,7	489	70	16	15
BDOT	44,9	1154	116	56	96
GESUT	29,9	795	90	40	16
MZ	1,86	28	7	2	0
SytWys	14,5	374	41	12	18
Osnowa	31,3	746	165	38	24

Taki sam wniosek nasuwa się po przebadaniu złożoności schematów aplikacyjnych GML. Nie jest to wynik zaskakujący, ponieważ schemat aplikacyjny GML jest ściśle związany ze schematem aplikacyjnym UML – stanowi jego „tłumaczenie". Przekształcenie to oparte jest na zbiorze reguł kodowania określonych w normie ISO 19136 (ISO/TC 211, 19136:2007).

Kolejnym etapem badań powinno być zbadanie złożoności próbek z danymi – plików GML zawierających już konkretne obiekty i sprawdzenie wpływu złożoności schematów aplikacyjnych na jakość (w tym złożoność) samych danych.

Podsumowanie i wnioski

Na podstawie analizy złożoności kilku wybranych schematów aplikacyjnych UML i GML widać wyraźnie, że niektóre schematy są bardzo złożone. Z jednej strony wynika to z obszerności samego zakresu tematycznego, którego dany schemat dotyczy, z drugiej zaś strony może być to skutek złego (nieefektywnego) zaprojektowania takiej struktury.

Przetestowane metryki złożoności nie oddają w pełni charakteru schematów aplikacyjnych UML i GML, opracowanych w GUGiK. Nie uwzględniają na przykład takich właściwości jak: użycie stereotypu „voidable" (UML), wartości „nilReason" (GML), klas abstrakcyjnych (UML, GML), różnych rodzajów geometrii (UML, GML), ograniczeń nakładanych na atrybuty (UML), wzajemnych zależności między poszczególnymi schematami aplikacyjnymi (UML, GML). Właściwości te mają istotny wpływ na złożoność struktur zapisanych w UML i XML Schema, a także na złożoność samych danych. Dlatego też, dalsze prace w tym temacie zakładają opracowanie autorskich metryk złożoności, dostosowanych do specyfiki schematów aplikacyjnych UML i GML zawartych w rozporządzeniach.

Zdaniem autorki możliwe jest również zapisanie modeli UML w formacie XMI (ang. *XML Metadata Interchange*), czyli w języku XML i wówczas na potrzeby obliczania złożoności diagramów klas UML można wykorzystać zarówno metryki, jak i narzędzia dedykowane strukturom zapisanym w XML.

W toku dalszych prac badawczych przewiduje się również przetestowanie funkcjonalności narzędzi GIS do obliczania entropii struktur zapisanych w językach UML i XML Schema oraz ewentualnie opracowanie własnego narzędzia przeznaczonego do tego celu.

Literatura

Basci D., Misra S., 2009: Measuring and evaluating a design complexity metric for XML schema documents. *Journal of Information Science and Engineering*, 25(5): 1405-1425.

Castor, 2014: The Castor Project. http://castor.codehaus.org/index.html

Chojka A., 2013: Niejednoznaczność transformacji UML-GML. *Roczniki Geomatyki*, t. 11, z. 1(58): 21-33, PTIP, Warszawa.

DeMarco T., 1986: Controlling Software Projects: Management, Measurement, and Estimates. Prentice Hall PTR Upper Saddle River, NJ, USA.

Genero M., Piattini M., Calero C., 2005: A Survey of Metrics for UML Class Diagrams. *Journal of Object Technology*, vol. 4, no. 9: 59-92, ETH Zurich.

ISO/TC 211 (Geographic Information/Geomatics), ISO 19136:2007, Geographic information – Geography Markup Language (GML). Norma PN-EN ISO 19136:2009, Informacja geograficzna – Język znaczników geograficznych GML.

Kang D., Xu B., Lu J., Chu W. C., 2004: A Complexity Measure for Ontology Based on UML. Proceedings of the 10th IEEE International Workshop on Future Trends of Distributed Computing Systems: 222-228.

Kim H., Boldyreff C., 2002: Developing Software Metrics Applicable to UML Models. 6th ECOOP Workshop on Quantitative Approaches in Object-Oriented Software Engineering.

Lämmel R., Kitsis S., Remy D., 2005: Analysis of XML schema usage. Proceedings of XML Conference 2005: 1-35, Atlanta, Georgia.

Lincke R., Lundberg J., Löwe W., 2008: Comparing Software Metrics Tools. Proceedings of the International Symposium on Software Testing and Analysis: 131-141, ACM, New York, USA.

Manso M.E., Genero M., Piattini M., 2003: No-Redundant Metrics for UML Class Diagram Structural Complexity. Advanced Information Systems Engineering, Lecture Notes in Computer Science Vol. 2681: 127-142, Springer, Berlin, Heidelberg.

McDowell A., Schmidt Ch., Yue K-B., 2005: Analysis and Metrics of XML Schema. International Conference on Software Engineering Research and Practice: 538-544.

SDMetrics, 2014: The Software Design Metrics tool for the UML. http://www.sdmetrics.com/

Tamayo A., Granell C., Huerta J., 2011: Analysing Complexity of XML Schemas in Geospatial Web Services. COM.Geo '11 Proceedings of the 2nd International Conference on Computing for Geospatial Research & Applications, Article No. 17, ACM New York, NY, USA.

Vargas R.T., Nugroho A., Chaudron M., Visser J., 2012: The Use of UML Class Diagrams and Its Effect on Code Change-proneness. Proceedings of the Experiences and Empirical Studies in Software Modelling Workshop 2012, Innsbruck, Austria.

Yi T., Wu F., Gan Ch., 2004: A Comparison of Metrics for UML Class Diagrams. *ACM SIGSOFT Software Engineering Notes*, Vol. 29 Issue 5: 1-6, New York, USA.

Streszczenie

Implementacja dyrektywy INSPIRE w Polsce oraz budowa Krajowej Infrastruktury Informacji Przestrzennej, spowodowały znaczny wzrost zainteresowania udostępnianiem danych przestrzennych i związanych z nimi usług, zwłaszcza przez organy publiczne i interesariuszy prywatnych. Zaowocowało to wieloma inicjatywami mającymi na celu harmonizację różnych zbiorów danych przestrzennych, a więc zapewnienie ich spójności logicznej i semantycznej.

Proces harmonizacji wymaga, albo opracowania nowych struktur danych, albo dostosowania już istniejących struktur danych przestrzennych do wytycznych i zaleceń INSPIRE. Struktury danych zapisywane są w postaci schematów aplikacyjnych UML i GML. Błędne lub zbyt złożone zapisy struktur danych mają bezpośredni wpływ na możliwość generowania plików GML z konkretnymi

danymi (obiektami), a tym samym mogą być przyczyną różnych problemów i anomalii na etapie produkcji danych.
Przedmiotem badań jest dokonanie pomiaru złożoności schematów aplikacyjnych UML i GML, opracowanych w Głównym Urzędzie Geodezji i Kartografii, w zakresie prac związanych z implementacją dyrektywy INSPIRE w Polsce. Zakłada się także dokonanie analizy istniejących miar złożoności struktur zapisanych w języku XML Schema oraz zbadanie możliwości wykorzystania różnych narzędzi do zmierzenia złożoności struktur zapisanych w języku UML i GML (XML Schema).

Abstract

Implementation of the INSPIRE Directive in Poland and construction of the National Spatial Data Infrastructure have caused a significant increase of interest in making spatial data and services available, particularly among public administration and private institutions. This entailed a series of initiatives that aim to harmonise different spatial data sets, so to ensure their logical and semantic coherence.
The process of harmonisation requires either working out new data structures or adjusting existing spatial data structures to the INSPIRE guidelines and recommendations. Data structures are described with the use of UML and GML application schemas. Incorrect or too complex data structures have direct influence on the ability to generate GML data sets with concrete data (objects), and thereby can cause various problems and anomalies at the data production stage.
The principal subject of this research is to measure complexity of UML and GML application schemas prepared in the Head Office of Geodesy and Cartography in Poland within the INSPIRE Directive implementation works. It is also assumed to analyse existing complexity measures of XML Schemas and to examine a possibility to use various tools to measure complexity of data structures expressed in UML and GML (XML Schema).

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