



Politechnika Wroclawska

Wydział Informatyki i Telekomunikacji

# ROZPRAWA DOKTORSKA

## **Metody modelowania i translacji modeli baz danych dla metamodelu asocjacyjnego**

Marcin Jodłowiec

Promotor:

dr hab. inż. Marek Krótkiewicz, prof. PWr

Wrocław 2022

*„Granice mego języka oznaczają granice mego świata.”*

— Ludwig Wittgenstein, *Traktat logiczno-filozoficzny*

## **Podziękowania**

*Pragnę podziękować mojemu promotorowi, dr. hab. inż. Markowi Krótkiewiczowi, za mnogość przekazanej wiedzy oraz cenne wsparcie, które istotnie wpłynęło na kształt niniejszej rozprawy.*

*Chciałbym również podziękować Panu Profesorowi Ngoc Thanh Nguyenowi za wszechstronną pomoc, a przede wszystkim za cenne wskazówki metodologiczne.*

*Dziękuję również kolegom dr. inż. Krystianowi Wojtkiewiczowi, dr. inż. Piotrowi Zabawie oraz pozostałym Koleżankom i Kolegom z Katedry Informatyki Stosowanej za konstruktywne komentarze, merytoryczne dyskusje oraz motywowanie mnie do napisania tej pracy.*

## Streszczenie

Niniejsza rozprawa doktorska dotyczy konceptualnego modelowania danych. W szczególności, podjęto zagadnienie opracowania wzorców modelowania oraz metod translacji modeli danych dla metamodelu asocjacyjnego. W ramach badań przyjęto założenie, iż kluczowym kryterium jakości jest zachowanie możliwie najwyższej zgodności semantycznej, zarówno w aspekcie tworzenia modeli, jak również translacji do innych metamodeli. Zgodność semantyczna jest tutaj rozumiana jako odzwierciedlenie pierwotnych konstrukcji myślowych znajdujących się w umyśle eksperta, bądź wyrażonych w ramach semantyki metamodelu.

W pracy zaproponowano system pojęć Conceptual Layer of Metamodels, stanowiący specyfikację konceptualizacji dziedziny modelowania danych. W oparciu o ten system wyspecyfikowano, zaimplementowano i przeanalizowano szereg wzorców modelowania dla metamodelu asocjacyjnego. Opracowano również metodę translacji semantyki, która operuje na reprezentacjach semantycznych modeli i metamodeli, zbudowanych w oparciu o zaproponowany system pojęć. Istotnym elementem niniejszego opracowania jest również metoda ewaluacji translacji semantyki, dostarczająca rozwiązań do ilościowej oceny zmiany semantyki w procesie translacji. Opracowane metody pozwoliły dokonać ekstrakcji i opisu semantyki metamodeli, translacji wraz z ewaluacją jej jakości, przy założeniu metamodelu asocjacyjnego jako źródłowego. Jako metamodele docelowe przyjęto ugruntowane w literaturze i praktyce rozwiązania z zakresu modelowania konceptualnego danych: Enhanced Entity-Relationship Model, Object-Role Modeling oraz Unified Modeling Language. Osiągnięte w pracy rezultaty zostały poddane weryfikacji przy pomocy studium przypadku. Studium to zostało oparte o model wyrażony w metamodelu asocjacyjnym, będący fragmentem struktury systemu reprezentacji wiedzy i służący do budowy sieci semantycznych.

## **Abstract**

This dissertation deals with conceptual data modeling. In particular, the problem of developing modeling patterns and data model translation methods for the association-oriented metamodel has been addressed. The main assumption of the research was that the key quality criterion is to maintain the highest possible semantic consistency, both in terms of model creation as well as translation to other metamodels. Semantic consistency is understood here as a reflection of the original thought structures found in the mind of an expert or expressed within the semantics of a metamodel.

In the thesis concept system named Conceptual Layer of Metamodels has been proposed, which is a specification of the conceptualization of the field of data modeling. Based on this system, a number of modeling patterns for the association-oriented metamodel have been specified, implemented and analyzed. A method aiming the translation of model semantics has also been developed. This method operates on the semantic representations of models and metamodels. These representations are built on the basis of the proposed concept system. An important element of this study is also the evaluation method of semantics translation, providing solutions for quantifying the change of semantics in the translation process. The developed methods allowed for the extraction and description of metamodel semantics, translation and evaluation of its quality, assuming the association-oriented metamodel as the source metamodel. As target metamodels, solutions in the field of conceptual data modeling, well-established in literature and practice, were adopted: Enhanced Entity-Relationship Model, Object-Role Modeling and Unified Modeling Language. The results achieved in the work were verified by the use of a case study. The case study was based on a model expressed in an association-oriented metamodel, which is a fragment of the knowledge representation system's structure and is used to express semantic networks.

# Spis treści

<b>Wykaz akronimów</b> . . . . .	8
<b>1. Wprowadzenie</b> . . . . .	10
1.1. Obszar tematyczny pracy . . . . .	10
1.2. Motywacja podjęcia tematu pracy . . . . .	10
1.3. Założenia pracy . . . . .	11
1.4. Problemy badawcze i cel pracy . . . . .	12
1.5. Zakres pracy i opis zawartości . . . . .	14
<b>2. Wybrane zagadnienia z zakresu modelowania oraz metamodelowania danych</b> . .	16
2.1. Podejście Entity-Relationship . . . . .	20
2.2. Object-Role Modeling . . . . .	21
2.3. Unified Modeling Language . . . . .	23
2.4. SBVR . . . . .	25
2.5. Metamodel asocjacyjny . . . . .	28
2.6. Transformacje modeli danych . . . . .	33
2.7. Obecne nurty rozwoju modelowania konceptualnego . . . . .	35
2.8. Podsumowanie . . . . .	36
<b>3. System pojęć w dziedzinie modelowania danych</b> . . . . .	43
3.1. Podstawowe elementy systemu pojęciowego . . . . .	45
3.2. Abstrakcyjność pojęciowa i konkretność pojęć modelowania . . . . .	48
3.3. Definicja uogólnionego zbioru mechanizmów i właściwości modelowania . .	49
3.3.1. Pojęcia rdzenne . . . . .	49
3.3.2. Pojęcia związane z modelowaniem klasyfikatorów . . . . .	53
3.3.3. Pojęcia związane z modelowaniem cech . . . . .	54
3.3.4. Pojęcia związane z modelowaniem zbiorowości . . . . .	55
3.3.5. Pojęcia związane z modelowaniem bytów . . . . .	58
3.3.6. Pojęcia związane z kategoryzacją pojęć . . . . .	60
3.3.7. Pojęcia związane z modelowaniem powiązań . . . . .	64
3.4. Pojęcia związane z modelowaniem wzorców . . . . .	69
3.5. Podsumowanie . . . . .	72
<b>4. Wzorce w modelowaniu asocjacyjnym</b> . . . . .	73

4.1.	Wzorce abstrakcyjne . . . . .	74
4.1.1.	Lista . . . . .	75
4.1.2.	Słownik . . . . .	84
4.1.3.	Autoreferencja . . . . .	92
4.1.4.	Pęk ról . . . . .	104
4.1.5.	Sieć uogólniona . . . . .	115
4.2.	Wzorce dedykowane . . . . .	130
4.2.1.	Bikompozycyjny Tandem Asocjacja-Kolekcja . . . . .	130
4.2.2.	Abstrakcyjny Cel Roli . . . . .	133
4.2.3.	Abstrakcyjny Właściciel Roli . . . . .	138
4.2.4.	Kaskada asocjacji . . . . .	142
4.3.	Podsumowanie . . . . .	148
<b>5.</b>	<b>Translacja semantyki modeli asocjacyjnych . . . . .</b>	<b>149</b>
5.1.	Koncepcja translacji . . . . .	149
5.1.1.	Ekstrakcja semantyki metamodelu źródłowego . . . . .	151
5.1.2.	Budowa cząsteczek semantycznych metamodelu źródłowego . . . . .	154
5.1.3.	Utworzenie wzorców mapowania . . . . .	156
5.1.4.	Przekształcenia aplikacyjne . . . . .	156
5.2.	Ekstrakcja semantyki metamodelu asocjacyjnego . . . . .	160
5.2.1.	Pojęcia w ramach metamodelu . . . . .	160
5.2.2.	Atomy semantyczne metamodelu . . . . .	161
5.2.3.	Budowa cząsteczek semantycznych metamodelu asocjacyjnego . . . . .	164
5.3.	Translacja semantyki AOM na EER . . . . .	167
5.3.1.	Specyfikacja fragmentu semantyki metamodelu związków-encji . . . . .	167
5.3.2.	Cząsteczki semantyczne przyjętego metamodelu związków-encji . . . . .	170
5.3.3.	Wzorce mapowania . . . . .	172
5.4.	Translacja semantyki AOM na ORM . . . . .	184
5.4.1.	Specyfikacja fragmentu semantyki metamodelu modelowania obiekt-rola . . . . .	184
5.4.2.	Cząsteczki semantyczne przyjętego metamodelu modelowania obiekt-rola . . . . .	188
5.4.3.	Wzorce mapowania . . . . .	189
5.5.	Translacja semantyki AOM na UML . . . . .	198
5.5.1.	Specyfikacja fragmentu semantyki metamodelu zunifikowanego języka modelowania . . . . .	198
5.5.2.	Cząsteczki semantyczne przyjętego metamodelu zunifikowanego języka modelowania . . . . .	202
5.5.3.	Wzorce mapowania . . . . .	204
5.6.	Podsumowanie . . . . .	214

<b>6. Metoda ewaluacji translacji semantyki</b>	215
6.1. Miary translacji wzorców mapowania	215
6.1.1. Przełożone atomy semantyczne	217
6.1.2. Zmienione atomy semantyczne	217
6.1.3. Wymuszone atomy semantyczne	217
6.1.4. Utracone atomy semantyczne	218
6.1.5. Przełożone pojęcia	218
6.1.6. Zmienione pojęcia	219
6.1.7. Wymuszone pojęcia	219
6.1.8. Utracone pojęcia	219
6.2. Miary translacji semantyki metamodeli	220
6.2.1. Miary pierwszego poziomu	220
6.2.2. Miary semantyki wynikowej	222
6.3. Miary ewaluacji semantyki modeli	223
6.3.1. Miary pierwszego poziomu	225
6.3.2. Miary semantyki wynikowej modelu	227
6.4. Podsumowanie	228
<b>7. Ewaluacja translacji semantyki metamodeli i dyskusja otrzymanych wyników</b>	229
7.1. Miary pierwszego poziomu	229
7.2. Miary translacji semantyki metamodeli	236
7.3. Miary semantyki wynikowej	239
7.4. Podsumowanie	240
<b>8. Studium przypadku</b>	241
8.1. Przedstawienie modelu	241
8.2. Opis modelu w świetle wzorców modelowania asocjacyjnego	243
8.2.1. Lista	243
8.2.2. Słownik	244
8.2.3. Autoreferencja	244
8.2.4. BACT	245
8.2.5. Abstrakcyjny cel roli	246
8.2.6. Abstrakcyjny właściciel roli	246
8.2.7. Kaskada asocjacji	247
8.3. Ekstrakcja semantyki modelu	248
8.4. Translacje modelu	248
8.4.1. Translacja semantyki do EER	248
8.4.2. Translacja semantyki do ORM	258
8.4.3. Translacja semantyki do UML	263

8.5.	Ewaluacja translacji semantyki modelu . . . . .	268
8.5.1.	Miary pierwszego poziomu . . . . .	269
8.5.2.	Miary semantyki wynikowej . . . . .	271
8.6.	Podsumowanie . . . . .	272
<b>9.</b>	<b>Podsumowanie . . . . .</b>	<b>273</b>
	<b>Dodatki . . . . .</b>	<b>276</b>
<b>A.</b>	<b>Przyjęta konwencja użycia wyrażeń SBVR . . . . .</b>	<b>277</b>
	Wyrażenia . . . . .	277
	Słowa kluczowe i frazy określające sformułowania logiczne . . . . .	277
	Kwantyfikacje . . . . .	278
	Operacje logiczne . . . . .	278
	Operacje modalne . . . . .	278
	Pozostałe słowa kluczowe . . . . .	279
<b>B.</b>	<b>Opis notacji symbolicznej . . . . .</b>	<b>280</b>
	<b>Bibliografia . . . . .</b>	<b>283</b>
	<b>Spis tablic . . . . .</b>	<b>296</b>
	<b>Spis rysunków . . . . .</b>	<b>297</b>



# Wykaz akronimów

**AOM** Association-Oriented Metamodel.

**AFN** Association-Oriented Formal Notation.

**AML** Association-Oriented Modeling Language.

**AODB** Association-Oriented Database Metamodel.

**AQL** Association-Oriented Modeling Language.

**ARD** Abstract Role Destination.

**ARO** Abstract Role Owner.

**BACT** Bicompositive Association-Collection Tandem.

**BPMN** Business Process Model and Notation.

**CLoM** Conceptual Layer of Metamodels.

**E-R** Entity-Relationship Model.

**EER** Enhanced Entity-Relationship Model.

**MDA** Model-Driven Architecture.

**MDE** Model-Driven Engineering.

**MOF** Meta Object Facility.

**OCL** Object Constraint Language.

**OMG** Object Management Group.

**ORM** Object-Role Modeling.

**OWL** Web Ontology Language.

**RM** Relational Model.

**SBVR** Semantics of Business Vocabulary and Business Rules.

**SKB** Semantic Knowledge Base.

**CVRM<sup>SKB</sup>** Cyclic Value Ranges Module.

**DSM<sup>SKB</sup>** Dimension & Space Module.

**ESNM<sup>SKB</sup>** Extended Semantic Network Module.

**LM<sup>SKB</sup>** Linguistic Module.

**OCM<sup>SKB</sup>** Ontological Core Module.

**RM<sup>SKB</sup>** Relationship Module.

**SM<sup>SKB</sup>** Structural Module.

**SQL** Structured Query Language.

**UFO** Unified Foundational Ontology.

**UML** Unified Modeling Language.

**UPM** Unified Process Metamodel.

**XML** Extensible Markup Language.

**XMLS** Extensible Markup Language Schema.

# 1. Wprowadzenie

Niniejsza dysertacja dotyczy opracowania metod modelowania i translacji modeli baz danych dla metamodelu asocjacyjnego. Modelowanie danych rozumiane jest tutaj jako proces przenoszenia konstrukcji myślowych do postaci struktur wyrażonych w ramach konkretnego metamodelu. W pracy przedstawiono metody modelowania stanowiące opis sposobu przenoszenia konstrukcji myślowych, gdzie kluczową rolę pełnią wzorce zawierające specyfikację koncepcji myślowych modelującego. Fundamentalnym, z punktu widzenia przyjętego podejścia, jest opracowany system pojęciowy o charakterze świata otwartego, będący specyfikacją konceptualizacji dziedziny modelowania i metamodelowania. Translacja modeli rozumiana jest jako przekształcenie konstrukcji semantycznych wyrażonych w metamodelu źródłowym do konstrukcji semantycznych wyrażonych w innym metamodelu. W pracy podjęto zagadnienia dotyczące modeli baz danych w kontekście stricte konceptualnym, skupionym na warstwie semantycznej, tj. abstrahującej od kluczowych kwestii bazodanowych takich jak np. systemy zarządzania bazami danych. Kontekstem opracowanych metod jest metamodel asocjacyjny [88] charakteryzujący się wzajemnie jednoznacznym odwzorowaniem kategorii semantycznych metamodelu w ich konceptualne odpowiedzialności.

## 1.1. Obszar tematyczny pracy

Obszar tematyczny pracy dotyczy metamodelowania i modelowania danych. Praca obejmuje zagadnienia związane z semantyką metamodeli i modeli, pragmatyką modelowania oraz tematykę translacji modeli.

## 1.2. Motywacja podjęcia tematu pracy

Podstawową motywacją pracy jest chęć wypełnienia luki badawczej w zakresie możliwości specyfikacji semantyki metamodeli oraz modeli danych całkowicie abstrahujących od ich składni. Jest to kluczowe z punktu widzenia zdolności do wyrażania konstrukcji myślowych w sposób możliwie precyzyjny i jednoznaczny. Konstrukcje myślowe z założenia stanowią „czystą semantykę”, tzn. nie jesteśmy

w stanie określić dla nich żadnej składni w żadnym języku, gdyż znajdują się one w umyśle modelującego. Umysł modelującego jest całkowicie poza naszym postrzeganiem w zakresie budowy oraz reprezentacji syntaktycznej pojęć i związków między nimi. Istotą modeli jest przenoszenie semantyki, a składnia jest jedynie koniecznym nośnikiem posiadającym wiele ograniczeń. Dlatego też kluczowym dla podjęcia tematu było wyekstrahowanie „czystej” semantyki tak, aby ograniczenia semantyczne nie stanowiły bariery dla wyrażania koncepcji oraz translacji między metamodelami. Prowadzone dotychczas badania nie zawierają całościowej metody ewaluacji zgodności pierwotnej semantyki (istniejącej w modelu źródłowym) z jej odwzorowaniem w modelu docelowym. Ponadto, model asocjacyjny jako relatywnie nowe podejście do modelowania nie posiada bogatej literatury naukowej w zakresie pragmatyki modelowania, a w szczególności translacji z i do niego.

Istotność luki badawczej polega głównie na tym, że translacje opierające się głównie na aspektach składniowych modeli (w niebezpośrednim stopniu dotyczącym semantyki) posiadają niezwykle istotną wadę. Polega ona na tym, iż podczas translacji następuje mniejsze lub większe zaburzenie (przekłamanie) w zakresie przenoszenia semantyki. Jest to istotny problem, gdyż w efekcie tego zaburzenia model docelowy może wyrażać całkowicie lub przynajmniej w części co innego (semantycznie) niż model pierwotny. Należy przy tym pamiętać, iż translacja (podobnie jak w językach naturalnych) powinna przenosić znaczenie wyrażen lub konstrukcji składniowych, a nie „mechanicznie” przekształcać jedno wyrażenie w drugie. Ponadto, należy podkreślić, iż wzorce modelowania oraz translacji stanowią bardzo istotny mechanizm zapewniający rozwiązywanie typowych problemów w ustalony i optymalny sposób. Dlatego też, ważne jest, aby wzorce te powstały oraz ich fundamentem było znaczenie (semantyka), a nie same konstrukcje gramatyczne.

### **1.3. Założenia pracy**

Założenia pracy sformułowano jako logiczną sekwencję następujących obserwacji:

1. Istotą modelowania konceptualnego danych jest przeniesienie semantyki wycinka rzeczywistości znajdującej się w umyśle eksperta, na postać zapisaną w pewnym języku opartym o określony metamodel, w sposób jak najbardziej wierny.
2. Podczas dokonywania translacji, może nastąpić pewna zmiana semantyki. Jakość opracowanych modeli, rozumiana jako zgodność przenoszona przez nie semantyki z semantyką koncepcji zawartych w umyśle modelującego, może i powinna podlegać ewaluacji. Association-Oriented Metamodel (AOM) posiada własności

odnoszące się do semantyki, które wspierają jednoznaczność translacji przy jednocześnie relatywnie dużej pojemności semantycznej.

#### 1.4. Problemy badawcze i cel pracy

Poniżej zostały przedstawione kluczowe problemy badawcze podjęte w pracy.

**Problem specyfikacji konceptualizacji domeny modelowania danych** Opis semantyki modelu danych związany z konkretnym metamodelem jest obarczony zależnościami wynikającymi ze składni tego metamodelu. Problemem jest uzyskanie takiej możliwości do formułowania semantyki w sposób jak najbardziej zatomizowany. Rozwiązanie tego problemu pozwoli na precyzyjne dopasowywanie konstrukcji składniowych danego metamodelu do wymagań dotyczących modelu. Języki formalne lub półformalne typu Unified Modeling Language (UML) stanowią zbyt duże ograniczenie i wymuszają wyrażanie koncepcji za pomocą bardzo konkretnych konstrukcji składniowych. Z drugiej strony, użycie języka naturalnego lub półformalnych rozwiązań, takich jak Semantics of Business Vocabulary and Business Rules (SBVR) posiada również wadę polegającą na tym, że aby ich użyć, konieczne jest wyrażanie konstrukcji myślowych za pomocą terminów wymagających zdefiniowania w kontekście systemu pojęć osoby modelującej. Mówiąc wprost, pojęcia reprezentowane przez terminy używane przez osobę modelującą mogą się znacznie różnić od terminów i pojęć innych osób. Rodzi to konieczność unifikacji zarówno terminologii, jak również systemu pojęć, aby przenoszona przez konkretne zapisy semantyka była dla każdej osoby interpretującej taka sama.

**Problem wzorców dla metamodelu asocjacyjnego** AOM jest rozwiązaniem o ścisłej semantyce, tzn. został opracowany z dużą dbałością o jednoznaczność budowanych wyrażen. Aktualnie, nie istnieje żadne zwarte opracowanie, kompleksowo podejmujące temat modelowania asocjacyjnego w kontekście wzorców. Wzorce modelowania jako uznane, ogólne i optymalne metody rozwiązywania typowych problemów są bardzo istotnym aspektem domeny modelowania. Mają one też ścisły związek z pragmatyką oraz powiązaną z nią wydajnością i jakością modelowania. Pojęcie wydajności i jakości modelowania jest trudne do jednoznacznego zdefiniowania, a tym bardziej do zmierzenia, jednak intuicyjnie są dobrze rozumiane. Kompleksowe i dogłębne przeanalizowanie zagadnienia wzorców projektowych oraz dokonanie syntezy wyników badań w tym zakresie jest bardzo pożądane z punktu widzenia modelowania asocjacyjnego.

**Problem translacji semantyki modeli** Model konceptualny wyrażony za pomocą systemu pojęć podlega translacji, co wynika z kilku powodów. Najbardziej naturalnym powodem konieczności translacji jest jego implementacja, czyli wyrażenie modelu konceptualnego w sposób zapewniający jego fizyczną realizację. Problem implementacji nie jest podejmowany *sensu stricto*. Natomiast leży on w ogólnym obszarze problemów istotnych z punktu widzenia zagadnień podejmowanych w ramach niniejszej rozprawy. Translacja dotyczy przekształcania modeli pomiędzy różnymi metamodelami, co oczywiście niekoniecznie musi oznaczać, że model docelowy będzie modelem fizycznym. Jest to zagadnienie będące bliską analogią do tłumaczenia np. tekstu w języku naturalnym z jednego języka na inny. Kluczowym zagadnieniem jest to, aby semantyka modelu docelowego była możliwie jak najbardziej zbliżona do semantyki modelu źródłowego, gdyż tylko wówczas możemy mówić o precyzyjnej translacji, tzn. takiej, która nie zmienia lub zmienia w jak najmniejszym stopniu sens opisu pierwotnego. W szczególności, zagadnienie translacji dotyczy modeli baz danych, lub ogólniej – modeli danych, jak również modeli konceptualnych. Natomiast konkretnym problemem badawczym podjętym w niniejszej rozprawie jest zagadnienie translacji modeli asocjacyjnych do metamodeli: UML, Enhanced Entity-Relationship Model (EER), Object-Role Modeling (ORM).

Zawężenie to wynika z faktu wskazania użyteczności prowadzonych badań dla konkretnych rozwiązań, mimo iż podjęty problem badawczy ma charakter jak najbardziej ogólny.

**Problem miary zmiany semantyki** Niniejszy problem badawczy jest ściśle związany z zagadnieniem dotyczącym translacji modeli. Jak już zostało opisane, każda translacja niesie ze sobą ryzyko zmiany semantyki. Oprócz problemu precyzyjnego wyrażania modelu w innym niż jego pierwotny metamodel, istnieje uzasadniona potrzeba oceny jakości tej translacji. Miarą oceny jakości translacji może być poziom zmiany semantyki modelu podczas wykonywania tej operacji. W związku z powyższym, ważnym z punktu widzenia teoretycznego, ale i również praktycznego, jest opracowanie sposobu wyznaczania różnic semantycznych w ramach translacji metamodeli. Pozwoli to zarówno na ocenę jakości modelu docelowego, jak również na ocenę samej metody translacji. Ponadto, bardzo interesującym problemem badawczym jest ocena zgodności semantycznej metamodeli, którą można bardzo ogólnie wyrazić jako potencjalną zdolność wyrażania różnych koncepcji myślowych za pomocą dwóch różnych metamodeli. Oznacza to podjęcie zagadnienia zgodności semantycznej zarówno na poziomie modeli, jak i na poziomie metamodeli. Z oczywistych powodów, ocena zgodności semantycz-

nej metamodeli jest zagadnieniem zdecydowanie bardziej złożonym, a przez to trudnym do uchwycenia, ale tym samym bardziej interesującym i wartościowym z poznawczego punktu widzenia.

Celem pracy jest rozwiązanie powyższych problemów badawczych poprzez zaproponowanie rozwiązań będących wynikiem prowadzonych badań. Rozwiązania te mają postać następujących artefaktów:

- systemu pojęć w dziedzinie modelowania,
- wzorców modelowania asocjacyjnego,
- metod translacji semantyki modeli,
- metod ewaluacji zmiany semantyki podczas translacji.

## **1.5. Zakres pracy i opis zawartości**

W ramach pracy wyodrębniono dziewięć rozdziałów. Rozdział pierwszy stanowi wprowadzenie do pracy.

W rozdziale drugim przedstawiono przegląd literatury obejmujący wybrane zagadnienia z zakresu obszaru prowadzonych badań. W szczególności, zaprezentowano kluczową literaturę obejmującą standard SBVR oraz AOM. Przedstawiono też w sposób syntetyczny inne rozwiązania, takie jak UML, EER, ORM. Rozdział podsumowano porównaniem pojemności semantycznej wybranych metamodeli w postaci uproszczonego studium przypadku opartego o model struktury dedykowanej reprezentacji wiedzy. Model struktury stanowiący uproszczone studium przypadku został wybrany w sposób świadomy, uwzględniający specyfikę podejmowanych zagadnień.

W rozdziale trzecim opisano opracowany w ramach pracy system pojęć w dziedzinie modelowania danych, stanowiący półformalne, ontologiczne, uogólnione rozwiązanie w zakresie opisu budowy oraz złożoności semantycznej metamodeli danych. System ten abstrahuje od składni, zarówno konkretnej, jak również abstrakcyjnej konkretnych metamodeli danych. Dodatkowo, wprowadzono mechanizmy abstrakcji i konkretyzacji, pozwalający na używanie opracowanych terminów do opisu pojęć na styku dowolnych warstw modelowania i metamodelowania. System ma postać słownika terminologicznego pojęć oraz podzielony został na zagadnienia dotyczące modelowania klasyfikatorów, cech, zbiorowości, bytów, pojęć i powiązań. System ten oparto o SBVR.

Rozdział czwarty prezentuje autorskie ujęcie metodyki modelowania dostosowanej do AOM. Metodyka uwzględnia wykorzystanie wzorców modelowania, które

zostały wyspecyfikowane w sposób abstrahujący od metamodelu danych, przy użyciu systemu pojęć. Specyfikacja wzorców opiera się o atomizację semantyczną, w ramach której wyodrębniono *atomy semantyczne*<sup>1</sup> konieczne do zaistnienia instancji danego wzorca, jak również zbiory własności możliwych, składających się z atomów semantycznych, które potencjalnie mogą we wzorcu zaistnieć. Zbiór wzorców obejmuje zarówno wzorce ogólne, abstrakcyjne, jak również charakterystyczne dla właściwości modelowania asocjacyjnego. W pracy zaprezentowano implementacje tych wzorców dla AOM oraz przykłady zastosowań.

Rozdział piąty opisuje metodę translacji modeli asocjacyjnych do innych metamodeli. W oparciu o system pojęć, opracowano metodę opisu złożoności semantycznej metamodeli oraz modeli danych. Metoda polega na ekstrakcji semantyki pierwotnej metamodelu do postaci zbioru atomów semantycznych, a następnie tworzenie konstrukcji składających te atomy, tzw. *cząsteczek semantycznych*. Te z kolei wykorzystywane są do opisu wzorców mapowania pomiędzy cząsteczkami w dwóch metamodelach. W rozdziale przedstawiono opis koncepcji z uwzględnieniem przekształceń występujących w metodzie oraz implementację tej koncepcji dla AOM jako metamodelu źródłowego oraz EER, UML, jak również ORM jako metamodeli docelowych.

Rozdział szósty przedstawia autorską metodę ewaluacji translacji uwzględniającą zmiany semantyki. Metoda ma za zadanie stanowić zestaw miar służących do oceny możliwości przeniesienia semantyki pierwotnej metamodelu źródłowego do metamodelu docelowego oraz do określenia zgodności semantycznej modeli będących źródłem i wynikiem translacji.

W rozdziale siódmym przedstawiono ewaluację translacji semantyki metamodelu asocjacyjnego AOM do metamodeli docelowych: EER, UML oraz ORM. Ewaluacja została przeprowadzona według metody przedstawionej w ramach poprzedniego rozdziału.

W rozdziale ósmym przedstawiono studium przypadku, zarówno w aspekcie opisu instancji wzorców wykorzystanych podczas budowy modelu wyrażonego w ramach AOM, jak również translacji do wspomnianych powyżej modeli docelowych. Studium przypadku oparto o asocjacyjną strukturę reprezentacji rozszerzonych sieci semantycznych, wchodzącą w skład Semantycznej Bazy Wiedzy – Semantic Knowledge Base (SKB).

Rozdział dziewiąty stanowi konkluzywne podsumowanie pracy ze wskazaniem wkładu naukowego pracy.

---

<sup>1</sup> Przez *atom semantyczny* rozumie się półformalne wyrażenie semantyki o charakterze elementarnym, niepodzielnym. Atomy werbalizowane są w postaci SBVR i skonstruowane są przy użyciu systemu pojęć przedstawionego w pracy.



## 2. Wybrane zagadnienia z zakresu modelowania oraz metamodelowania danych

Modele danych definiują strukturę (składnię) oraz znaczenie (semantykę) danych. Stosowanie modeli danych w procesach wytwarzania oprogramowania pozwala na poprawę jakości tworzenia systemów informatycznych. Jakość ta rozumiana jest w kontekście spełnienia szeregu kryteriów oceny oprogramowania w zakresie spełniania przez niego założonych wymagań. Równie ważnym aspektem tworzenia modeli jest wspieranie w zakresie jednoznaczności wyrażania opisu sposobów rozwiązania problemów, usprawniający komunikację pomiędzy osobami zaangażowanymi w rozwój oprogramowania. Modele dają możliwość tworzenia spójnego widoku na dane, który pozwala na integrację wielu elementów systemu, bądź wielu systemów. Proces modelowania danych zakłada odwzorowywanie analizowanej rzeczywistości poprzez dokonywanie uproszczeń, tj. poprzez proces abstrakcji rozumiany zgodnie z definicją Guttaga [54]:

*Esencją abstrakcji jest zachowanie informacji, która w danym kontekście jest istotna oraz zapomnienie informacji, która w tym kontekście nie jest istotna.*<sup>1</sup>

Kolejnym istotnym aspektem modelowania jest wyrażanie analizowanych elementów w ramach przyjętej składni abstrakcyjnej wyrażającej semantykę bytów z modelowanego wycinka rzeczywistości. Pojęcie *modelu* jako takie definiowane jest na wiele różnych sposobów. Jedną z definicji mającą zastosowanie zarówno w zakresie inżynierii oprogramowania, w tym modelowania konceptualnego, definiuje model jako *uproszczony obraz złożonej rzeczywistości* [36]. Guizzardi [51] dodaje do tej definicji konieczność uwzględnienia pewnej *konceptualizacji*.

Niniejszy rozdział obejmuje tematykę przedstawienia rozwiązań z zakresu *modelowania danych*, stanowiących główny przedmiot zainteresowania niniejszej rozprawy. W celu odniesienia się do nich, w ramach rozważań autor używa pojęcia kryjącego się pod terminem *metamodel danych*. Metamodel danych<sup>2</sup> stanowi ramy i ograniczenia składniowe dotyczący budowy modeli oraz ich semantykę. Za Seide-

---

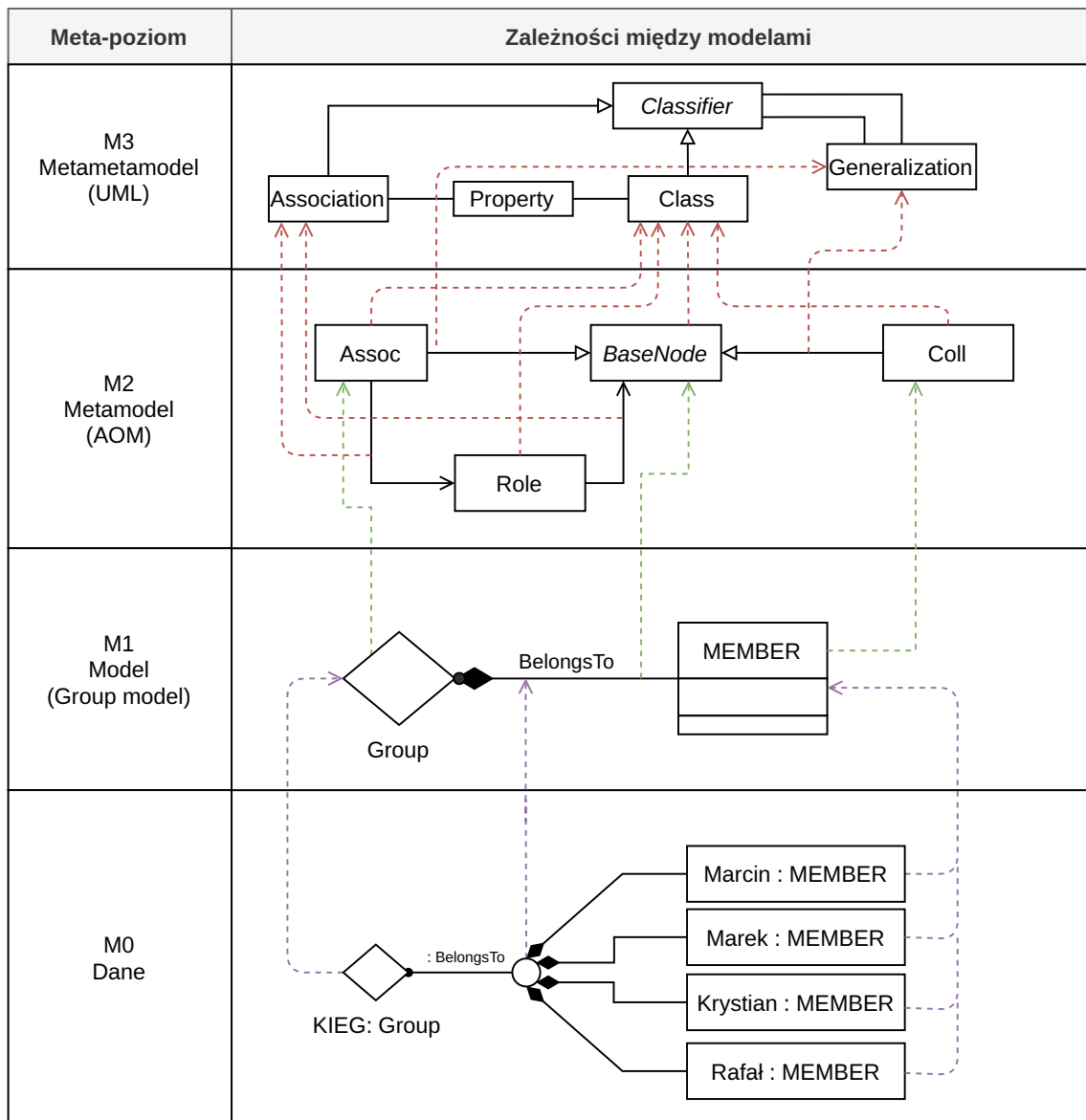
<sup>1</sup> Tekst oryginalny. *The essence of abstraction is preserving information that is relevant in a given context, and forgetting information that is irrelevant in that context.*

<sup>2</sup> czasem zapisywany w literaturze również jako meta-model lub meta model

witzem, metamodel tworzy deklaracje odnośnie tego, co może być wyrażone przez poprawne modele wyrażony w danym języku modelowania<sup>3</sup> [129]. W praktyce, oznacza to, iż metamodel traktuje się jako *model modelu* [66]. W aspekcie modelowania danych, metamodel danych oznacza abstrakcyjny model o określonej składni i semantyce, który służy do definiowania modeli danych. Należy jednak wspomnieć, iż zależność model – metamodel jest zależnością względną. W praktyce, modelowanie oparte jest o odrębne *meta-poziomy*, często oznaczane kolejnymi indeksami numerycznymi, połączonych relacją *jest modelem* (rys. 2.1). Klasyczna architektura modelowania oparta o metamodel zaproponowana przez Bézivina operuje na 4 poziomach abstrakcji [8]. Na diagramie 2.1 pokazano symboliczne warstwy w czterowarstwowej architekturze modelowania opartej o metamodel asocjacyjny. Warstwa M3 – meta-metamodel pokazuje kategorie modelowania zdefiniowane w ramach języka UML. Kategorie te zostały użyte do wyrażenia metamodelu asocjacyjnego będącego w warstwie metamodelu (M2). Modelowanie odbywa się w warstwie M1, w ramach której następuje konceptualizacja wycinka rzeczywistości i wyodrębnienie jego modelu, korzystając z pojęć definiowanych w ramach metamodelu. Warstwa M0 obejmuje część ekstensjonalną modelu, tj. dane, które są z tym modelem zgodne. Należy jednak zauważyć, iż przypisanie konkretnych numerów danym poziomom (1-3) jest tutaj umowne. Rozważać można dowolne *n*-warstwowe hierarchie metapoziomów, a sama hierarchia – w ogólnym ujęciu – zawieszona jest we względnej przestrzeni [103]. Wzajemne powiązania pomiędzy podstawowymi pojęciami w ramach opisu, czym jest metamodel i jakie jest jego miejsce w technikach modelowania w sposób czytelny zostały zobrazowane w [83] (rys. 2.2). Diagram ten przedstawia związki pomiędzy kluczowymi elementami modelowania. Autorzy pokazują, że w skład metody modelowania wchodzi technika obejmująca język modelowania i procedury składające się z szeregu kroków, mówiące o tym, jak danej techniki używać. Metamodel został ujęty jako zbiór obejmujący reguły składni abstrakcyjnej, semantykę oraz odwzorowanie reguł składniowych w semantykę. Jednocześnie, organizacja tych reguł przebiega według pewnego schematu znaczeniowego, wyrażonego w sposób formalny (np. w formie semantyki denotacyjnej, operacyjnej, algebraicznej [23]), bądź nieformalny – w formie języka naturalnego. Dodatkowo, można również rozważać sposoby wykorzystania danego metamodelu, tj. jego pragmatykę. Podsumowując, w kontekście metamodelu można mówić o trzech jego częściach: składniowej, semantycznej i pragmatycznej.

1. Część składniowa obejmuje zbiór pojęć pierwotnych oraz reguł ich wiązania.

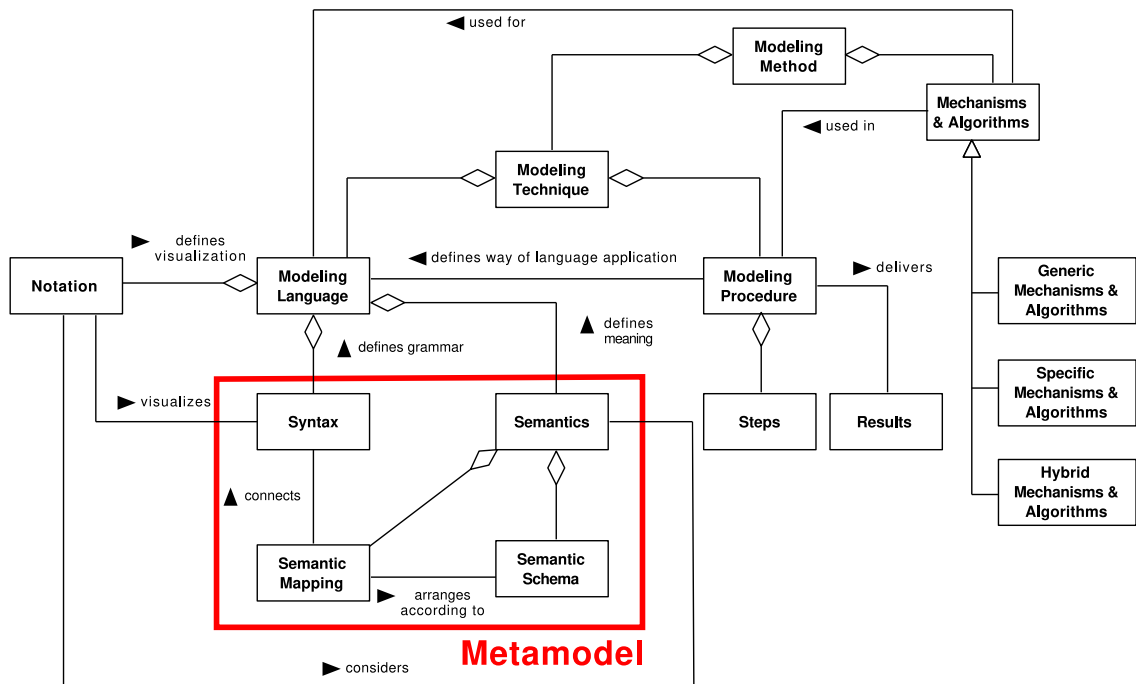
<sup>3</sup> W celu wyrażenia modelu zgodnego z danym metamodelem, konieczne jest posiadanie składni konkretnej języka modelowania (zob. rys. 2.2).



Rysunek 2.1. Warstwy modelowania wraz z przykładem ich zastosowania dla AOM.

Zarówno pojęcia pierwotne, jak i reguły składniowe z założenia abstrahują od sposobów rozwiązywania konkretnych problemów modelowania.

2. Część semantyczna mówi o znaczeniu poszczególnych pojęć pierwotnych, jak i reguł składniowych. Innymi słowy, odpowiada na pytania, czym są (co reprezentują) te pojęcia pierwotne. Każdy metamodel posiada zdefiniowaną semantykę i jest to semantyka pierwotna, tj. wynikająca z samej definicji tego metamodelu i nie należy jej mylić z semantyką nadawaną jej w późniejszym etapie przez użytkowników, gdyż ten element jest zwany *pragmatyką*, czyli opisem sposobu użycia.
3. Część pragmatyczna obejmuje opis sposobu wykorzystania metamodelu w celu wyrażenia pewnej semantyki modelu. To znaczy, wzorce oraz inne konstrukcje



Rysunek 2.2. Komponenty metod modelowania. Opracowano na podstawie [83]

wyrażone w konkretnym metamodelu danych mają charakter użytkowy, a co za tym idzie praktyczny. Za Thalheimem [143], w zakresie badań nad modelowaniem baz danych pragmatyka bywa w dużej mierze zaniedbywana, a zagadnienia związane ze składnią wychodzą na pierwszy plan.

Przykładem metamodelu danych może być model relacyjny<sup>4</sup> opracowany przez Codda [22], stanowiący podstawę relacyjnych baz danych. Metamodel ten operuje na takich kategoriach semantycznych<sup>5</sup> jak m.in.: *schemat bazy danych*, *schemat relacji*, *relacja*, *zależność funkcjonalna*, *złączenie* i pozwala opisywać rzeczywistość w taki sposób, aby móc ją wyrazić w formie bazy danych.

Na przestrzeni kilkudziesięciu ostatnich lat, w dobie rozwoju systemów informatycznych powstawało wiele licznych metamodeli danych. Różnią się one od siebie wieloma elementami, np. liczbą i rodzajem kategorii semantycznych. Z tego powodu wykorzystywane są w różnych celach. Część z nich operuje jedynie na strukturalnych aspektach danych, inne obejmują też aspekty behawioralne. Metamodeli takie jak Entity-Relationship Model (E-R) [19] ujmują dane jedynie na poziomie konceptualnym, tj. pozwalając wyrazić specyfikację danych skupiając się na jak najbardziej precyzyjnym odwzorowaniu konstrukcji myślowych autora modelu, ignorując tym

<sup>4</sup> Relational Model (RM)

<sup>5</sup> Kategoria semantyczna – pojęcie używane w ramach danego metamodelu lub innego systemu konceptualizacji rzeczywistości w celu odwoływania się do pewnych abstrakcyjnych bytów, którym w ramach tego opisu nadano pewne znaczenie.

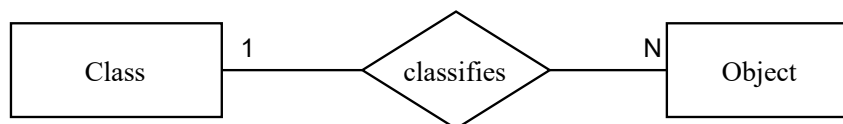
samym aspekty implementowalności. Inne, jak wcześniej wspomniany RM, w zamyśle autora był konceptualnym, formalnym metamodelem baz danych. Należy jednak wspomnieć, że stał się bezpośrednią podstawą większości istniejących dzisiaj systemów bazodanowych o charakterze strukturalnym, opartych o język Structured Query Language (SQL) [63].

## 2.1. Podejście Entity-Relationship

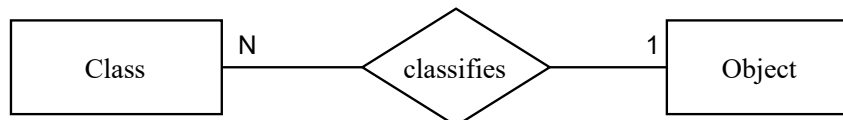
E-R [19] stanowi metamodel stosowany od lat 70-tych XX w. do modelowania danych, w szczególności baz danych. Mimo dojrzałości, metamodel ten jest wciąż w użyciu, jak również podlega ciągłemu rozwojowi [120]. W jego oryginalnej postaci, zostały zdefiniowane takie kategorie jak m.in.: *zbiór encji*, *zbiór związków*, *atrybut*, *encja*, *związek*. Jego reguły składniowe operują na łączeniu zbiorów encji przy pomocy zbiorów związków, jak również na wiązaniu zbiorów związków i encji z atrybutami. Modele wyrażane są najczęściej w postaci diagramów, które zostały zbiorczo opisane w ramach publikacji [135]. Do najpopularniejszych notacji należą:

- Chena (oryginalna) [19, 34],
- Teorey’a [140],
- Elmasri & Navathe [33],
- Barkera („kruczej stopki”) [6],
- Martina (Information Engineering) [42].

Każda z notacji implikuje zestaw symboli graficznych oznaczających kolejne kategorie semantyczne metamodelu. Ze względu na ograniczenia notacyjne, nie wszystkie z notacji pozwalają wyrazić każdy konstrukt gramatyczny metamodelu E-R. Dla przykładu, większość z wyżej wymienionych notacji wspiera jedynie związki binarne. Pomiedzy notacjami występują nie tylko różnice gramatyczne, ale również semantyczne. Istnieją co najmniej dwie semantyki odczytywania krotności oraz ograniczeń uczestnictwa elementów w związkach w ramach różnych notacji (rys. 2.3). *Look-here semantics* (rys. 2.3a) zakłada, że w celu uzyskania informacji o krotności (maksymalnej liczności) elementów uczestniczących w związku, należy patrzeć „*tutaj*”, tj. na liczbę umieszczoną na diagramie przy uczestniku związku. Alternatywną semantyką jest *Look-across semantics* (rys. 2.3b) – w tym przypadku należy patrzeć na liczbę znajdującą się po drugiej stronie związku. Zagadnienie semantyki krotności na w modelach związków-encji poruszono po raz pierwszy w publikacji [40], a następnie rozwinięto w [24, 142].



(a) Semantyka krotności *look-across*.



(b) Semantyka krotności *look-here*.

Rysunek 2.3. Zobrazowanie różnic pomiędzy *look-here semantics* oraz *look-across semantics*. W zależności od przyjętej semantyki, oba diagramy mogą reprezentować dwa zdania: „Jedna klasa klasyfikuje wiele obiektów. Jeden obiekt jest klasyfikowany przez jedną klasę.”

Opracowano na podstawie [13] oraz [135]

Wraz z rozwojem potrzeb oraz technik modelowania danych zaproponowano szereg rozszerzeń do oryginalnego E-R. Wynikiem modyfikacji stała się grupa rozszerzonych modeli związków-encji (EER) [5, 35, 140]. Model został wzbogacony o m.in.:

- mechanizm związków *generalizacja-specjalizacja*,
- związki agregujące (*part-of*),
- operatory AND i XOR pomiędzy związkami,
- złożone typy danych.

W publikacji [41] zaproponowano spójny, zunifikowany metamodel EER. Wyrażono ten metamodel w sposób konceptualny, przy użyciu jego samego oraz diagramu klas UML. Praca ta pozwoliła w pewien sposób uporządkować kategorie metamodelu oraz powiązania występujące pomiędzy nimi, pozwalając na tworzenie narzędzi transformacyjnych, walidacyjnych i innych opartych o ten metamodel.

## 2.2. Object-Role Modeling

Innym, koncepcyjnie odmiennym od E-R, metamodeliem służącym do wyrażania konceptualnych modeli danych jest ORM zaproponowany przez Halpina w latach 70-tych XX w. [57]. Kluczowa różnica polega na podejściu do konceptualizacji danych poprzez odejście od idei atrybutów. W ramach ORM świat postrzegany jest jako zbiór *obiektów*, które odgrywają *role*, tzn. biorą udział w związkach. Typy związków nazywane są w ORM *typami faktów*. Typy faktów mogą być unarne, binarne, oraz *n*-arne. W literaturze określa się następujące zalety podejścia bezatrybutowego do modelowania danych [59]:

- semantyczna stabilność – minimalizacja wpływu zmian spowodowanych koniecznością odnotowania informacji o atrybucie,
- naturalna werbalizacja – wszystkie fakty i reguły mogą być w łatwy sposób zwerbalizowane w postaci języka naturalnego, aby być możliwymi do zrozumienia przez eksperta dziedzinowego,
- populowalność<sup>6</sup> – przykładowe zbiory instancji typów faktów mogą być w dogodny sposób przedstawione tabelarycznie,
- unikanie wartości null – informacja o braku instancji pewnego faktu lub nieznaney informacji na jego temat powoduje zwyczajnie niezainstancjonowanie tego faktu.

Oczywistą wadą jest niemożliwość odnotowania semantyki modelu związanej z tym, czy dany fakt reprezentuje informację o atrybucie, tj. pewnej inherentnej własności innego bytu. Należy mieć świadomość, iż w pewnych zastosowaniach niemożliwość łatwego wyodrębnienia atrybutów z modelu może rodzić trudności, np. w fazie implementacji. Należy jednak dodać, iż istnieją procedury związane z prostym generowaniem struktur atrybutowych (np. diagramów E-R, diagramów klas UML) w sposób automatyczny, tworząc pewne perspektywy schematu ORM.

Oryginalne publikacje Halpina nie brały pod uwagę zagadnień metamodelowania. W późniejszych publikacjach przedstawiono jednak dwa metamodely opisujące w sposób uporządkowany kategorie semantyczne ORM oraz powiązania pomiędzy nimi. ORM został wyrażony przy pomocy swoich własnych kategorii. Później, dostosowano metamodel ORM do architektury Object Management Group (OMG) i przedstawiono opracowanie metamodelu wraz ze składnią abstrakcyjną w zgodności z Meta Object Facility (MOF) w publikacjach [117, 118].

Publikacja [61] specyfikuje obecnie używany dialekt metamodelu tj. ORM 2, skupiając się na przeglądzie jego kategorii semantycznych, jak również prezentuje notację graficzną. W artykułach [55, 56] Halpin opisuje modelowanie związków-encji z perspektywy ORM. Pokazano, w jaki sposób mapować wzorce projektowe często występujące w modelowaniu danych z notacji Barkera oraz Information Engineering do ORM. Zagadnienia związane z modelem polimorfizmu i dziedziczenia typów opisano w [65]. W artykule [62] poruszono problemy modelowania kolekcji, tj. zaprezentowano wzorce projektowe wykorzystujące ograniczenia krotności w zakresie modelowania struktur realizujących takie koncepcje jak *zbiór*, *wielozbiór* czy *ciąg*.

Metamodel ORM doczekał się również publikacji opisujących istotne zagadnienie, jakim jest translacja jego modeli do innych metamodeli. Publikacja [64] opisuje zasady transformacji do relacyjnego modelu danych. W artykule [10] omówiono metody projektowania schematów Extensible Markup Language (XML) przy uży-

<sup>6</sup> ang. *populatability*

ciu konceptualizacji w postaci modeli ORM, jak również optymalne transformacje ORM  $\mapsto$  Extensible Markup Language Schema (XMLS) [147].

### 2.3. Unified Modeling Language

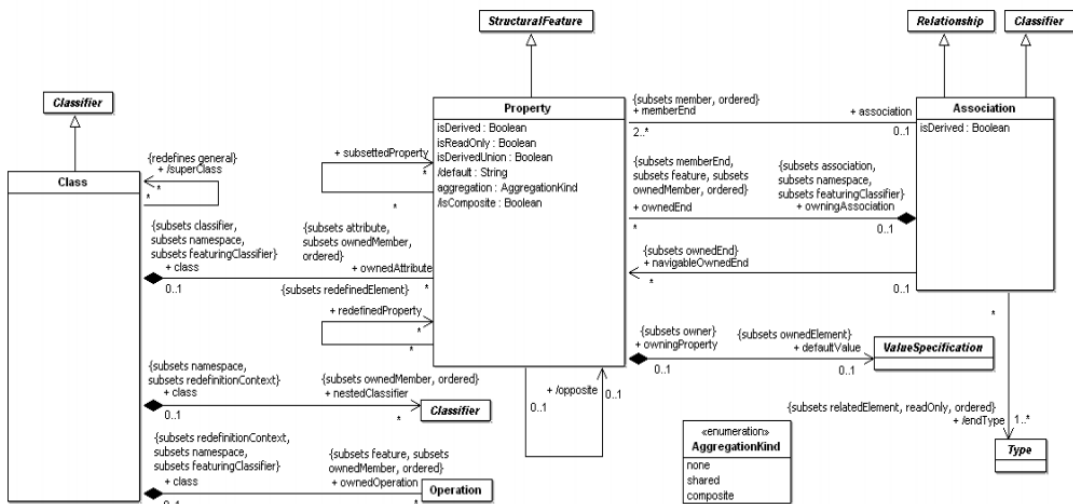
Jednym z najczęściej używanych standardów do modelowania jest UML [114] opracowany przez OMG. Standard określa składnię abstrakcyjną, składnię konkretną i semantykę wyrażoną poprzez szereg diagramów, które łącznie stanowią język używany m.in. w inżynierii oprogramowania do tworzenia modeli systemów (głównie systemów informatycznych).

Język ten opiera się na podejściu zorientowanym obiektowo do modelowania systemów informatycznych, zarówno w aspekcie modelowania strukturalnego (diagram klas, diagram komponentów, diagram obiektów, diagram rozlokowania, diagram struktury kompozytowej, diagram pakietów i diagram profili), jak i behawioralnym, tj. odnoszącym się do opisu zmian modelowanych elementów w czasie oraz interakcji pomiędzy nimi (np. diagram maszyny stanowej, diagram aktywności, diagram przypadków użycia, diagramy interakcji – diagram sekwencji, komunikacji, przeglądu interakcji oraz diagram czasowy).

Specyfika modelowania konceptualnego danych w UML opiera się na wykorzystaniu diagramu klas, którego wewnętrzną strukturę przedstawiono na rys. 2.4. Modelowane typy bytów przyjmują postać klas (*Class*), a ich właściwości mają charakter atrybutów tych klas, realizowanych przez kategorię *Property*. Język ten umożliwia precyzyjny opis związków pomiędzy bytami w oparciu o kategorie składniowe *Association* oraz *Property*. Związki w UML mogą być binarne lub  $n$ -arne, być cechowane przez dowolną krotność, fizyczną nawigowalność rozdzieloną od konceptualnej kierunkowości, mogą ograniczać czas życia elementów wiązanych przy pomocy kategorii *AggregationKind*. Język UML pozwala reprezentować struktury zarówno o charakterze grafowym, ale i hipergrafowym przy wykorzystaniu asocjacji  $n$ -arnych [46]. Należy jednak zauważyć, iż semantyka takich związków jest w UML trudna do uchwycenia i w swojej interpretacji odbiega od klasycznego ich rozumienia, np. w modelowaniu związków-encji, czy w reprezentacji wiedzy [20, 78] tzn. ma charakter raczej szeregu związków binarnych rozważanych łącznie. Świadczy o tym np. poniższy fragment specyfikacji [114]:

*W przypadku  $n$ -arnych asocjacji, kres dolny krotności końcówki zazwyczaj jest równy 0. Kres dolny krotności dla końcówki  $n$ -arnej asocjacji równy 1*





Rysunek 2.4. Metastruktura diagramu klas UML 2.5.1. Źródło: [114]

(lub więcej) implikuje, iż jedno (lub więcej) połączenie musi istnieć dla każdej możliwej kombinacji wartości dla pozostałych końcówek <sup>7</sup>.

UML wydaje się być atrakcyjny z punktu widzenia konceptualnego modelowania danych i przez wiele lat stanowił podstawę prac badawczych z tej dziedziny [97, 98, 102, 128], również o charakterze krytycznym. Zarzuty względem UML odnoszą się najczęściej do jego *niespójności*, *wieloznaczności*, *nieadekwatności* oraz *złożoności*. Krytyczny opis zagadnień modelowania rzeczywistych systemów z wykorzystaniem UML można odnaleźć m.in. w publikacjach [7, 47, 132].

Niejednoznaczność UML wynika przede wszystkim z nieformalnych opisów w języku naturalnym, które są przemieszane z ograniczeniami wyrażonymi w Object Constraint Language (OCL) oraz diagramami opisującymi semantykę pierwotną, wynikającą ze składni wyrażonej również w UML. Niejednokrotnie, użytkownik języka pozostawiony jest z możliwością własnej interpretacji, jego konstruktów, np. w przypadku interpretacji *AggregationKind* o wartości *shared*. Specyfikacja [114] definiuje ją jako:

*shared* – Określa, że *Property* ma semantykę współdzielonej agregacji. Precyzyjna semantyka współdzielonej agregacji jest odmienna w zależności od obszaru zastosowań i modelującego.<sup>8</sup>

<sup>7</sup> Tekst oryginalny: *For n-ary Associations, the lower multiplicity of an end is typically 0. A lower multiplicity for an end of an n-ary Association of 1 (or more) implies that one link (or more) must exist for every possible combination of values for the other ends.*

<sup>8</sup> Tekst oryginalny: *shared* – *Indicates that the Property has shared aggregation semantics. Precise semantics of shared aggregation varies by application area and modeler.*

Aby przezwyciężyć przynajmniej część z tych niedogodności, poczyniono starania mające na celu próbę pewnej formalizacji UML [37, 38, 115]. Formalizacja ta w szczególności odnosi się to precyzyjnej definicji semantyki w oparciu o matematyczny rygor definicyjny.

UML nie jest metamodeliem dedykowanym tylko do modelowania danych, lecz narzędziem zdecydowanie bardziej ogólnym. Mimo to, bywa używany do modelowania baz danych (np. relacyjnych [101], dokumentowych [146] czy grafowych [30]). Odbywa się to niejednokrotnie poprzez rozszerzenie go przy pomocy dedykowanych profili – lub bezpośrednio, interpretując w odpowiedni sposób diagramy klas [69, 108, 136]. Badania zaprezentowane w pracy [32] wykazały, że UML daje większy poziom zrozumienia projektowanych modeli danych, niż podejście E-R. Należy mieć jednak na uwadze, że eksperymenty przeprowadzane były w środowisku akademickim (grupy studentów), nie zaś komercyjnym. Z drugiej strony, w innym, empirycznym badaniu [31] przeanalizowano sposób używania konceptualnych metamodeli danych przez praktyków. Wykazano, że E-R jest używany ponad dwukrotnie częściej, niż UML, zarówno wśród osób o krótkim stażu, jak również ekspertów.

## 2.4. SBVR

SBVR jest standardem OMG stanowiącym narzędzie służące do półformalnego opisu złożonych bytów przy użyciu kontrolowanego języka naturalnego. Standard ten tworzony był od 2003 do 2008, kiedy to OMG zaproponowało jego wersję 1.0. Aktualną wersją SBVR, która została w niniejszej rozprawie ma numer 1.5 [113]. Aktywnie rozwijane elementy w zakresie wykorzystania SBVR dotyczą zagadnień mapowania pomiędzy różnymi komponentami architektury Model-Driven Architecture (MDA) opracowanymi przez OMG. Literatura przedmiotu opisuje rozwiązania w zakresie wykonanych metod oraz narzędzi dla następujących zagadnień:

- generowanie diagramów UML:
  - diagramów klas [110, 124],
  - diagramów aktywności [110, 124],
  - diagramów sekwencji [124],
  - diagramów klas adnotowanych wyrażeniami OCL [110],
- generowanie modeli bazodanowych [111] i adnotacja ich ograniczeniami o większej sile wyrazu [121] oraz
- wsparcie w generowaniu modeli procesów biznesowych w Business Process Model and Notation (BPMN) [134].

Inne prace z zakresu SBVR dotyczą akwizycji sformalizowanych reguł z języka naturalnego [106]. SBVR używany jest również w rozwijanych metodach automatycznego mapowania wymagań pomiędzy diagramami klas UML i językiem semi-naturalnym [133]. Aktywnie podejmowanym zagadnieniem jest również działanie odwrotne, tj. ekstrakcja i generowanie reguł SBVR z diagramów przypadków użycia [29].

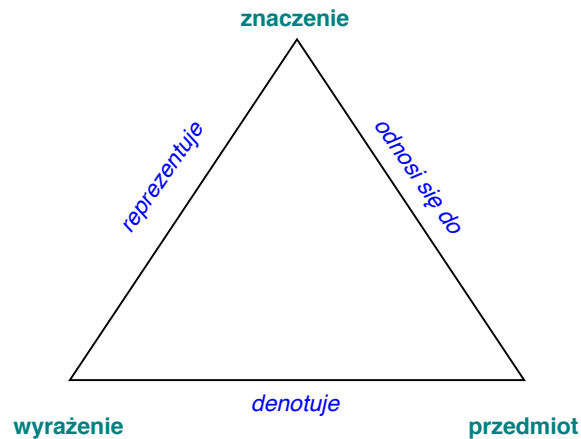
Podstawy związane z transformacją modeli biznesowych do modelu niezależnego od platformy (PIM – Platform Independent Model), a następnie do modelu specyficznego dla platformy (PSM – Platform Specific Model) zostały opisane przez Linehana w [99] podczas pracy nad samym standardem SBVR. Linehan opisał środowisko metodyki *Model-Driven Business Transformation* (MDBT) rozwijanej w IBM T.J. Watson Research Center, która ma na celu tworzenie wykonywalnych implementacji na bazie modeli biznesowych. Metodyka ta zakłada wykorzystywanie narzędzi wspierających język naturalny. Transformacja używa m.in. reguł wyrażonych w RuleML [12]. Artefakty UML, które są tworzone podczas transformacji to diagramy klas, diagramy stanów, diagramy przypadków użycia.

Inna metodyka transformacji wyrażeń SBVR do modeli UML została zaproponowana przez Raja [124]. Raj używa semantycznego słownika i reguł zapisanych w SBVR, aby automatyzować tworzenie modeli strukturalnych i behawioralnych: diagramów aktywności, sekwencji i klas. Tworzenie diagramów klas bazuje na wykorzystaniu metod transformacji wyspecyfikowanych w standardzie SBVR na potrzeby ekstrakcji nazw klas, instancji, atrybutów, związków generalizacja-specjalizacja. Metodyka ta wspiera również reguły mapowania do metod, klas, binarnych asocjacji, z uwzględnieniem ich krotności.

Istotnym wkładem naukowym w transformację wyrażeń SBVR do modeli UML i OCL jest narzędzie VeTIS [110]. Narzędzie pozwala na bezpośrednie mapowanie reguł strukturalnych SBVR na diagramy klas UML, natomiast reguł operacyjnych na operacje UML. Co więcej, reguły o charakterze modalnym (aletycznym i deontycznym) są przekształcane na postać wyrażeń OCL. Metodyka użycia VeTIS zaleca jednak niezależne definiowanie logiki procesów biznesowych w postaci np. diagramów aktywności UML lub diagramów BPMN, a wykorzystanie semantycznego słownika traktuje jedynie jako wsparcie.

Architektura SBVR opiera się o koncepcję trójkąta semiotycznego-semantycznego [112, 116, 145], który opisuje związki pomiędzy trzema warstwami: warstwą ekstensji (elementem składowym rzeczywistości), warstwą symbolu (elementem składowym języka) oraz warstwą znaczenia (elementem

składowym konceptualizacji). Zależności pomiędzy wspomnianymi elementami zobrazowano na rys. 2.5.



Rysunek 2.5. Trójkąt semiotyczno-semantyczny. Opracowano na podstawie [145] oraz [113]

**Wyrażenie** (ang. *expression*) oznacza symbole używane do komunikowania się (takie jak np. dźwięki, teksty, diagramy, gesty). Należy zauważyć, że jedno wyrażenie może odpowiadać wielu znaczeniom.

**Reprezentacja** (ang. *representation*) oznacza połączenie pomiędzy wyrażeniem a znaczeniem. Każda reprezentacja przypisuje dokładnie jedno znaczenie do jednego wyrażenia.

**Znaczenie** (ang. *meaning*) określa, co oznacza dane wyrażenie. Określa, jak myślimy o rzeczach.

**Ekstensja** (ang. *extension*) oznacza rzeczy, do których jest przypisane znaczenie. Mogą być one czymkolwiek – nawet wyrażeniami, reprezentacjami i znaczeniami, które znajdują się w rozważanej rzeczywistości.

W tabeli 2.1 przedstawiono sposób, w jaki poszczególne rzeczy przyporządkowane są do wyżej wymienionych kategorii w SBVR. Posłużono się w tym celu przykładem wycinka rzeczywistości w domenie modelowania, obrazując jak poszczególne kategorie (ekstensja, znaczenie, reprezentacja, wyrażenie) są używane do odwzorowania odpowiednio bytów, pojęć, przyporządkowań znaczenia do terminów oraz symboli. Każdy wiersz stanowi osobny przykład. Pierwszy wiersz opisuje pojęcie ogólne, drugi pojęcie jednostkowe, natomiast trzeci pojęcie czasownikowe.

SBVR pozwala definiować słowniki terminologiczne oraz reguły biznesowe o charakterze półformalnym, przy użyciu języka naturalnego. Wyrażenia w języku naturalnym mogą zostać zdefiniowane nieformalnie, półformalnie lub całkowicie formalnie. Formalizację wyrażeń uzyskuje się poprzez użycie czcionki wskazującej na:

Tablica 2.1. Przyporządkowanie rzeczy do poszczególnych kategorii w SBVR. Przykłady w dziedzinie modelowania.

Ekstensja	Znaczenie	Reprezentacja	Wyrażenie
istniejące modele, faktyczne artefakty wykonane przez inżynierów oprogramowania	pojęcie model – sposób konceptualizacji modeli, nasze o nich wyobrażenia	przypisanie do pojęcia model terminu „model”	ciąg znaków „model”
		zdefiniowanie pojęcia ‘model’ jako ‘uproszczony obraz wycinka rzeczywistości’	ciąg znaków ‘uproszczony obraz wycinka rzeczywistości’
właściwy metamodel asocjacyjny, konstrukt opracowany i opisany w [88]	pojęcie jednostkowe ‘AOM’ – sposób konceptualizacji tego metamodelu, nasze o nim wyobrażenia	„AOM” jako przypisanie terminu do pojęcia jednostkowego ‘AOM’	ciąg znaków ”AOM”
posiadanie – dla dowolnego metamodelu w zakresie każdej z jego kategorii	to, co rozumie się poprzez posiadanie kategorii przez metamodel	sformułowanie pojęcia czasownikowego “ <u>metamodel</u> posiada <u>kategorię</u> ” będące szablonem z symbolami zastępczymi ‘ <u>metamodel</u> ’ oraz ‘ <u>kategoria</u> ’	Napis “ <u>metamodel</u> posiada <u>kategorie</u> ”

- Odwołanie się do pojęcia zdefiniowanego w ramach słownika, np. [metamodel](#). Takie odwołanie oznacza, że dany termin należy rozumieć dokładnie w taki sposób, jak przedstawiono w jego definicji we wpisie do słownika. Definicje mogą być wyrażone w języku naturalnym, w SBVR, albo częściowo w SBVR.
- Odwołanie się do słowa kluczowego o semantyce zdefiniowanej w ramach standardu SBVR, np. [each](#), któremu przypisano semantykę [universal quantification](#).

Pełną specyfikację adnotacji języka naturalnego wzorcami wyrażen oraz czcionkami określonymi w SBVR zawierają aneksy A i B standardu [113]. Specyfikację użytej konwencji w niniejszej pracy dostosowanej do języka polskiego przedstawiono w załączniku A.

## 2.5. Metamodel asocjacyjny

Metamodelem danych, który stanowi główną oś niniejszej dysertacji jest AOM. Jest to rozwiązanie, które powstało stosunkowo niedawno<sup>9</sup>. AOM został opracowywany początkowo jako platforma do modelowania baz danych – Association-Oriented Database Metamodel (AODB), w celu efektywnego modelowania złożonych syste-

<sup>9</sup> Pierwsza publikacja na jego temat – [94] została wydana w roku 2014.

mów wymagających precyzyjnego opisu semantyki, takich jak systemy reprezentacji wiedzy [88]. Wraz z rozwojem koncepcji, AODB został uogólniony do AOM i stanowi metamodel danych, w ramach którego możliwe jest modelowanie zarówno aspektów bazodanowych systemów informatycznych, jak również aspektów konceptualnych. Umożliwia on modelowanie na poziomie konceptualnym, który jednocześnie jest poziomem fizycznym. Oznacza to, iż przechodząc poprzez poszczególne poziomy konceptualizacji nie ma potrzeby wykonywania jakichkolwiek transformacji między metamodelami danych. Transformacje takie często mają miejsce w klasycznym procesie tworzenia modeli relacyjnych baz danych opartym o E-R [39], który zakłada trzy poziomy modelowania:

- konceptualny (model wyrażony często w metamodelu E-R),
- logiczny (model wyrażony jako RM),
- fizyczny (model wyrażony w konkretnym systemie zarządzania bazami danych)

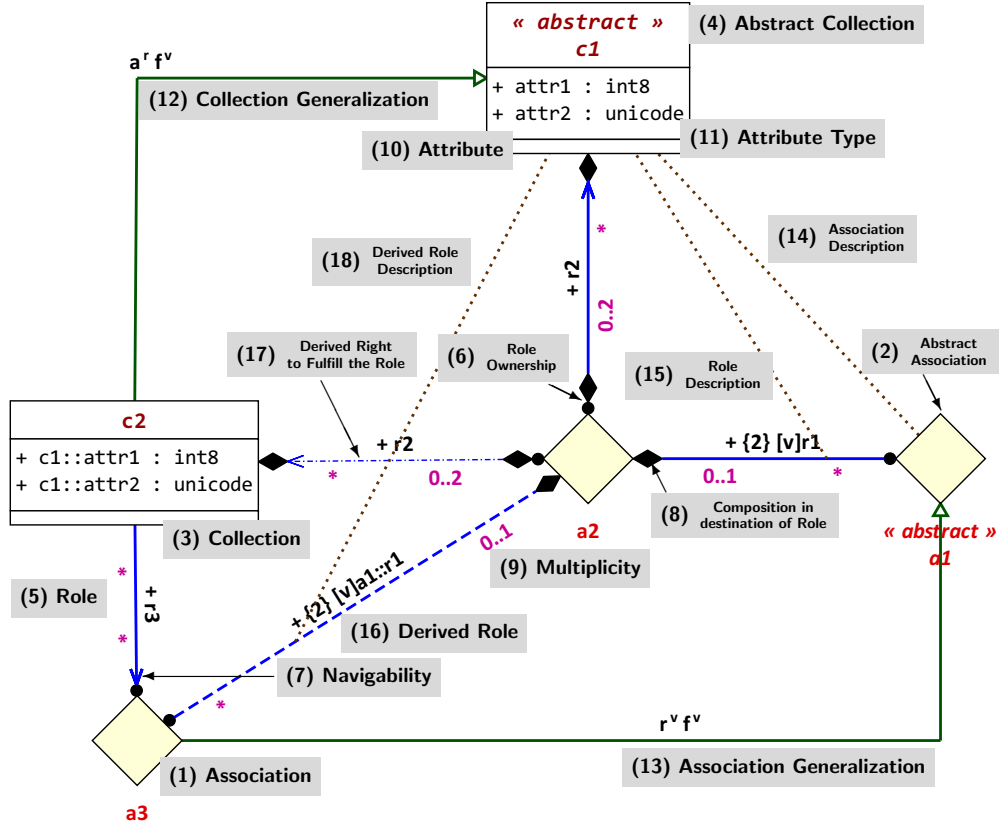
lub podejściu MDA do wytwarzania oprogramowania, gdzie następuje transformacja pomiędzy modelami PIM oraz PSM. AOM pozwala definiować modele danych o wysokim poziomie złożoności, m.in. poprzez następujące mechanizmy [91]:

- natywne wsparcie dla związków  $n$ -arnych,
- możliwość ograniczenia krotności zarówno po stronie związku, jak również po stronie elementu wiązanego,
- możliwość wzajemnego ograniczania czasu życia wiązanych elementów,
- rozdział dziedziczenia danych i powiązań,
- definiowanie związków, generalizacja-specjalizacja opartych o wirtualność, dziedziczalność i selektywny polimorfizm.

Część z tych mechanizmów jest unikalna dla AOM. Modele asocjacyjne opisywane są poprzez następujące pojęcia (rys. 2.6):

- intensjonalne:
  - *database* (*Db*) – baza danych,
  - *association* (*Assoc*) – asocjacja,
  - *collection* (*Coll*) – kolekcja,
  - *role* (*Role*) – rola,
  - *attribute* (*Attr*) – atrybut;
- ekstensjonalne:
  - *association object* (*AssocObj*) – obiekt asocjacji,
  - *object* (*Obj*) – obiekt,
  - *role object* (*RoleObj*) – obiekt roli.





Rysunek 2.7. Przykładowy diagram intensjonalny AML. Źródło: [88]

$$\begin{aligned}
 c_1^\emptyset \dots \diamond a_1^\emptyset[*] &\xrightarrow{+r_1^v(\diamond a_2) \dots c_1^\emptyset} [0..1] \diamond a_2; \\
 \diamond a_2[0..2] &\xrightarrow{+r_2(\square c_1^\emptyset)} [*] \square c_1^\emptyset; \\
 \diamond a_3[*] &\xrightarrow{+r_3(\square c_2)} [*] \square c_2; \\
 \diamond a_3 &\xrightarrow{f^v} \diamond a_1^\emptyset; \\
 \square c_2 &\xrightarrow{a^r, f^v} \square c_1^\emptyset; \\
 c_1^\emptyset &\langle +attr_1 : int8, +attr_2 : unicode \rangle
 \end{aligned} \tag{2.1}$$

Artykuł [89] opisuje rozwiązania AOM dotyczące modelowania związków  $n$ -arnych, ze szczególnym uwzględnieniem rozwiązania problemu niejednoznaczności. W wielu językach modelowania utrata semantyki związków  $n$ -arnych wynika ze sposobu ich realizacji jako zbioru związków binarnych. Mechanizm ten nazywany jest *reifikacją* lub „*obiektywizacją*” i jest zagadnieniem badawczym występującym zarówno w aspekcie modelowania informacji [27], jak i w reprezentacji wiedzy [50, 127, 131]. Podejście reifikacyjne stoi w opozycji do stosowanych w modelowaniu asocjacyjnym rozwiązań, zakładających tworzenie modeli, w których związki traktowane są jako samodzielne kategorie (ang. *first-class citizens*) [9, 109].

AOM wprowadza złożony mechanizm definiowania związków kategoryzacyjnych



(typu *generalizacja-specjalizacja*), opisany w [87], który dotyczy zarówno danych jak i związków. Dziedziczenie *kolekcji* (*Coll*) realizujących dane i *asocjacji* (*Assoc*) realizujących związki jest całkowicie rozdzielone i niezależne. W artykule opisano konsekwencje wynikające z definiowania *kolekcji* oraz *asocjacji* w wielopoziomowych strukturach hierarchicznych, uwzględniając:

- selektywność w zakresie dziedziczenia oraz polimorfizmu realizowanego przez metamodel jako *prawo do pełnienia ról*),
- tryby dziedziczenia: naturalny, wirtualny i realny,
- ograniczania w zakresie dziedziczalności<sup>10</sup>,
- porównanie algorytmów realizujących dziedziczenie z mechanizmem w programowaniu obiektowym na przykładzie języka C++.

Poniżej przedstawiono wybrane, najważniejsze publikacji opisujące pragmatykę AOM.

- W publikacji [92] przedstawiono moduł związków SKB. Wykorzystano AOM jako platformę do konstrukcji hipergrafowych modeli danych. Uzyskany moduł bazy wiedzy pozwala na precyzyjne określanie powiązań pomiędzy pojęciami w zakresie ich arności, hierarchiczności ról w nich występujących, ograniczenia relacyjne oraz uwzględnianie potencjalnie dozwolonych i zabronionych powiązań. W pracy przedstawiono budowę modułu, wraz z opisem intensjonalnych i ekstensjonalnych aspektów warstwy informacyjnej AOM.
- W publikacji [90] przedstawiono struktury danych i algorytmy dedykowane reprezentacji i przetwarzaniu cyklicznych przedziałów wartości. Przedstawione rozwiązania stanowią struktury reprezentacji przestrzenno-czasowej wiedzy rozmytej. Do realizacji tych struktur wykorzystano AOM, z tego powodu publikacja ta stanowi istotne studium przypadku tego metamodelu.
- W publikacji [150] wykorzystano AOM do reprezentacji metamodelu Unified Process Metamodel (UPM) służącego do modelowania procesów ciągłych. W ramach rozprawy, opisano składnię abstrakcyjną, konkretną oraz semantykę metamodelu, jak również implementację w AOM wzbogaconą o wyrażenia formalne.

---

<sup>10</sup> Zdaniem autora niniejszej rozprawy, pojęcie *dziedziczalności* jest jednym z postulatów, który sprawia, iż modelowanie asocjacyjne jest oddzielnym paradygmatem modelowania od modelowania obiektowego. Konsekwencją wprowadzenia takiego mechanizmu jest rezygnacja ze spełniania tzw. „zasady podstawienia Liskov” [100] (ang. *Liskov substitution principle*). Ceną rezygnacji z podstawialności jest utrudnienie realizacji wyrażen języka zapytań Association-Oriented Modeling Language (AQL) oraz innych konstruktów behawioralnych operujących na modelach asocjacyjnych opartych o generalizację-specjalizację. W zamian, uzyskujemy silną własność pozwalającą realizować bardzo specyficzny rodzaj polimorfizmu opartego o wyjątki, tj. zachowując własność typu, lecz nie zachowując pełnej struktury typu nadrzędnego.

## 2.6. Transformacje modeli danych

Poprzez termin *translacja*, zarówno potocznie, jak i technicznie rozumie się: przekład, tłumaczenie z jednego języka na inny. Termin ten posiada również znaczenie w geometrii Euklidesowej i określa transformację, która powoduje przesunięcie każdego punktu obiektu o pewien wektor, powodując jego *przełożenie* w przestrzeni. W niniejszej pracy pojęcie translacji rozumiane jest jako przekształcenie modelu w model, przy czym model źródłowy i docelowy wyrażane są w ramach różnych metamodeli, a ponadto operacja ta powinna być dokonana przy uwzględnieniu postulatów minimalizacji zmiany semantyki. W pewnym sensie analogicznie do geometrii, w modelowaniu danych można wyróżnić termin ogólniejszy – *transformacja* – który rozumiany jest jako przekształcenie do innej postaci. Termin *translacja* podkreśla, iż działania transformujące modele są zbieżne z działaniami tłumacza lub translatora podczas przetwarzania języków naturalnych i formalnych.

Problematyka transformacji modeli danych jest istotna w inżynierii oprogramowania, zarówno z teoretycznego, jak również z praktycznego punktu widzenia. W praktyce inżynierskiej, szczególnie istotne są zagadnienia związane z metamodelami stanowiącymi podstawę systemów opartych o bazy danych. Znamienita większość strukturalnych baz danych opiera się na RM [22], tzn. RM pełni rolę najniższego poziomu składowania danych. Natomiast modelowanie odbywa się często w modelach konceptualnych np. w modelu obiektowym, E-R. Implementacja może polegać na zastosowaniu mapowania obiektowego-relacyjnego (ang. *object-relational mapping*) [71], które jest klasycznym przykładem automatyzacji procesu translacji modeli pomiędzy dwoma metamodelami. Podejście polegające na bezpośrednim modelowaniu warstwy składu danych w modelu relacyjnym obarczone jest wieloma problemami. Jednym z istotnych problemów tego typu jest tzw. *niezgodności impedancji danych* [70] polegająca na mieszaniu dwóch paradygmatów w ramach jednego kodu źródłowego. Bardzo ważnym zagadnieniem np. w kontekście mapowania obiektowo-relacyjnego jest utrata semantyki [80], która wynika z posiadania przez RM mniejszej pojemności semantycznej.

Zagadnienie transformacji modeli znajduje szereg zastosowań oraz jest aktywnie rozwijane w ramach inżynierii sterowanej modelami<sup>11</sup>[126]. Transformacje modeli mogą odbywać się zarówno w obrębie tego samego metamodelu, jak również przechodzić pomiędzy różnymi metamodelami [96].

W pracy [26] sklasyfikowano podejścia do transformacji modeli. W pracy scharakteryzowano i opisano znane wówczas metody transformacji Model-to-Code (M2C) oraz Model-to-Model (M2M). Transformacje M2C obejmują przekształcenia, w ra-

---

<sup>11</sup> Model-Driven Engineering (MDE)

mach których wejściowy model przekształcany jest do postaci ciągu znaków (tekstu). Transformacje M2M uwzględniają przekształcenia, gdzie zarówno na wejściu, jak i na wyjściu przekształcenia znajduje się model. W artykule przedstawiono złożoną taksonomię transformacji uwzględniając właściwości, takie jak m.in.: cechy reguł transformacyjnych, cechy związków między źródłem, a elementem, również w aspekcie śledzenia zmian, cechy harmonogramowania i zorganizowania reguł, kierunkowość transformacji.

Same podejścia do tworzenia narzędzi służących do transformacji M2M zostały sklasyfikowane na następujące grupy [26]:

- Podejście polegające na bezpośredniej manipulacji (ang. *direct-manipulation approach*). Rozwiązania tego typu polegają głównie na opracowaniu narzędzi do samodzielnej implementacji reguł transformacyjnych przy użyciu języków programowania ogólnego przeznaczenia, takich jak np. C++ lub Java.
- Podejście relacyjne (ang. *relational approach*). Kategoria takich rozwiązań łączy podejścia deklaratywne oparte o matematyczne pojęcie relacji. W tym ujęciu, źródłowe i docelowe elementy w procesie transformacji opisane są przy pomocy relacji oraz dodatkowych ograniczeń [1].
- Podejście bazujące na transformacji grafów (ang. *graph-transformation-based approach*). W rozwiązaniach opartych o teorię grafów transformację mają postać typowanych, etykietowanych, atrybutowanych grafów reprezentujących np. modele UML'opodobne.
- Podejście strukturalne (ang. *structural approach*), które zakłada dwie fazy: pierwsza mająca na celu utworzenie hierarchicznej struktury modelu docelowego i druga, której celem jest ustalenie atrybutów oraz wzajemnych powiązań docelowych. O przebiegu algorytmów decyduje narzędzie, natomiast użytkownik dostarcza jedynie reguły transformacji.
- Podejście hybrydowe (ang. *hybrid approach*) łączy różne techniki przedstawione powyżej.

W literaturze przedmiotu dotyczącej transformacji modeli danych rozważa się transformacje pomiędzy różnymi modelami danych, również takich, które zostały zdefiniowane w oderwaniu od MDA. W szczególności, ze względu na wiele zastosowań praktycznych i łatwą możliwość wdrożenia rezultatów, często opisywane są translacje z i do modelu relacyjnego, obiektowego, związków-encji, oraz XML [43]. Atzeni i in. w pracy [3] rozważają zagadnienie translacji schematów modeli danych w ujęciu niezależnym od modelu z wykorzystaniem tzw. operatora ModelGen. Zastosowane podejście bazuje na użyciu języka Datalog [139] do wyrażania reguł

translacyjnych, a poszczególne komponenty translacyjne posiadają odzwierciedlenie w postaci relacyjnej. Interesującą pozycją literaturową w podobnym nurcie jest [119], w ramach której opisano metodę automatycznej translacji modeli danych. Zaproponowana translacja opiera się o zdefiniowany na potrzeby translacji wspólny metamodel, który pozwala klasyfikować kategorie modelowania w różnych modelach<sup>12</sup> Sposób reprezentacji modeli, metamodeli i schematów oparty jest o XML.

## 2.7. Obecne nurty rozwoju modelowania konceptualnego

Mnogość dostępnych metamodeli danych generuje bardzo istotne pytanie: *Jaki metamodel zastosować dla konkretnego zagadnienia?* Trudno wskazać jedną odpowiedź. W niniejszym rozdziale przedstawiono szereg prac dostarczających informacji umożliwiających porównanie metamodeli w zakresie kompromisów pomiędzy bogactwem semantycznym, a prostotą modelowania. Porównania takie wydają się być jednak subiektywne, w szczególności, gdy autorem dyskusji jest autor metamodelu starający się podnieść zalety swojego narzędzia<sup>13</sup>. Wydaje się, że należy się zgodzić ze stwierdzeniem Halpina [57], który przypisuje zmaganiom pomiędzy różnymi technikami modelowania konceptualnego rangę „*okazjonalnych, religijnych wojen*”<sup>14</sup>.

Interesujące są również prace badawcze prowadzone celem tworzenia metamodeli unifikujących istniejące podejścia. Do istotnych i ciekawych osiągnięć w tym zakresie należy metamodel unifikujący modele danych wyrażone w ORM, EER oraz UML [85]. Poza zaproponowanym rozwiązaniem, niezwykle wartościowym wkładem naukowym [85] jest jego *Appendix A* zawierający tabelę zestawiającą kategorie semantyczne poszczególnych metamodeli wraz z przyrównaniem do ich odpowiedników w innych metamodelach. Tabela ta pozwala w prosty sposób porównać złożoność metamodeli w aspekcie realizacji poszczególnych technik opisu złożoności świata.

Obecnie rozwijane zagadnienia w głównym nurcie przede wszystkim skupiają się na [138]:

- synergii uzyskanej poprzez połączenie inżynierii ontologii z modelowaniem konceptualnym,
- modelowania konceptualnego na potrzeby *Big Data*,
- zagadnień związanych z inżynierią oprogramowania (w tym rozwoju oprogramowania sterowanego modelami, ang. *Model-Driven Engineering*,

<sup>12</sup> W niniejszej pracy w sposób odmienny określono terminologię, metamodel [119] nazywany jest tutaj *metametamodel*, model danych *metamodelem*, natomiast, schemat – modelem.

<sup>13</sup> zob. np. [58] autorstwa T. Halpina, autora ORM

<sup>14</sup> org. „*By and large, the history of information systems modeling has been characterized by a plethora of techniques and notations, with occasional religious wars between proponents of different approaches.*”

- bogactwem semantycznym w modelach konceptualnych,
- modelowaniem procesów biznesowych.

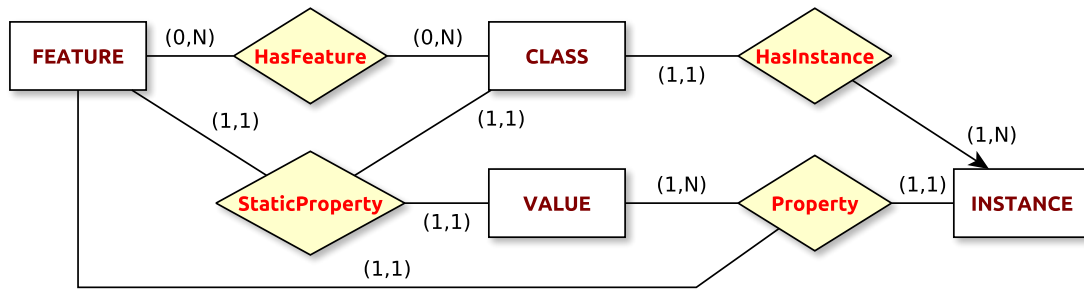
## 2.8. Podsumowanie

W celu wykazania istotności przedsięwzięcia naukowego związanego z rozwojem metod modelowania asocjacyjnego, przeprowadzono analizę własności modelowania danych we wspomnianych metamodelach: UML, ORM, EER, AOM. Przeprowadzono również analizę w zakresie możliwości zamodelowania prostego modelu reprezentacji wiedzy tak, aby uwzględnić w sposób bezpośredni jak najwięcej wymagań. Przyjęto poniższy zestaw wymagań:

1. Klasa jest związkiem łączącym zbiór instancji ze zbiorem cech.
2. Dla jednej klasy można utworzyć wiele instancji, przy czym jedna instancja może być tylko jednej klasy.
3. Klasa posiada wiele unikalnych cech, cecha może opisywać wiele klas.
4. Właściwość jest połączeniem instancji, dla której określona jest wartość cechy, przy czym właściwość musi posiadać przynajmniej jedną wartość, która może występować w dowolnej liczbie właściwości, musi być określona dla dokładnie jednej instancji i cechy.
5. Właściwość statyczna jest specyficznym rodzajem właściwości, która nie odnosi się instancji, natomiast określa klasę (dokładnie jedną).
6. Czas życia cechy determinuje czas życia właściwości określonych dla tej cechy oraz analogicznie dla instancji i właściwości, klasy i instancji, klasy i właściwości statycznej.
7. Wartość nie musi posiadać informacji o właściwościach, w których występuje.

Implementację modelu w EER przedstawiono na rys. 2.8. Pojemność semantyczna tego metamodelu nie pozwoliła jednakże w pełni, zrealizować wszystkich wymagań.

1. **Brak mechanizmu dziedziczenia związków.** Wymaganie (5) zakłada, że właściwość statyczna jest specyficznym rodzajem właściwości i doprecyzowuje jej semantykę. Wymaganie (4) określiło właściwość jako kategorię łączącą. EER (ani E-R) nie posiada możliwości tworzenia związków kategoryzacji związków, dlatego spełnienie wymagania (5) było niemożliwe.
2. **Brak mechanizmu łączenia związków ze sobą.** Wymaganie (5) zakłada wiązanie w ramach właściwości statycznej klasy. W wymaganiu (1) określono, iż klasa jest



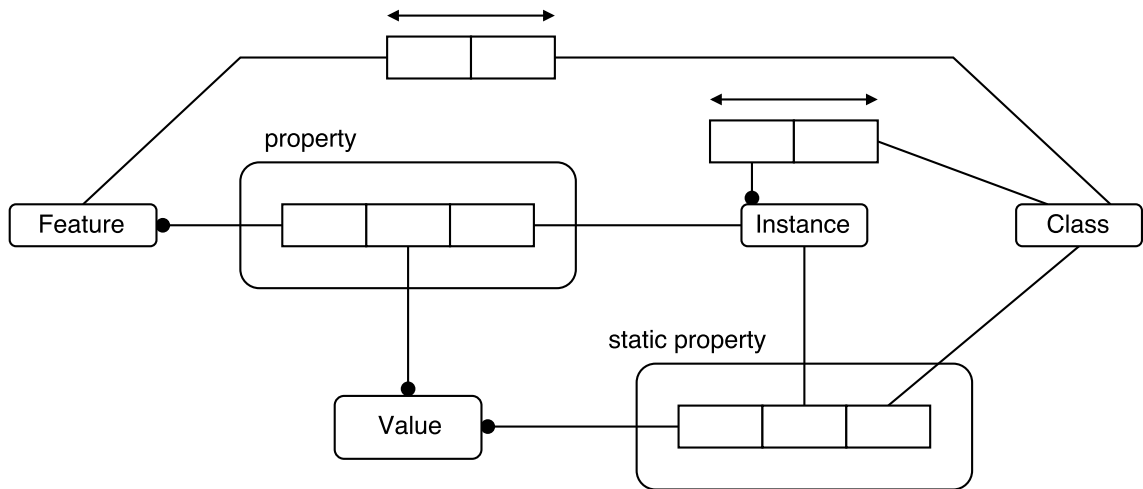
Rysunek 2.8. Struktura modelu realizująca wymagania 1-7 wyrażona w metamodelu EER.

elementem łączącym. Ze względu na brak możliwości łączenia związków pomiędzy sobą, spełnienie wymagania (5) nie było możliwe.

3. **Brak możliwości ograniczania krotności po stronie związku.** W wymaganiu (2) określono, iż instancja może być tylko jednej klasy. Wymaganie (1) określa, że klasa jest elementem łączącym. Zatem, wymaganie (2) nie może być w pełni zrealizowane, gdyż wymagałoby określenia, w ilu dokładnie związkach może wystąpić element.
4. **Brak możliwości wiązania czasu życia związku i encji.** W wymaganiu (6) określono, że czas życia instancji nie może być dłuży niż czas życia klasy. Oznacza to konieczność wiązania czasem życia elementu wiążącego (klasa jest elementem wiążącym – (1)) oraz elementu wiążanego – instancji. EER nie zapewnia takiego mechanizmu. Wymaganie (6) nie może być zatem w pełni zrealizowane.
5. **Brak możliwości wiązania czasem życia w związkach  $n$ -arnych** Mechanizm *existence dependency* w modelowaniu związków-encji jest ograniczony do związków binarnych. W wymaganiu (6) określono konieczność powiązania cechy, instancji, klasy i właściwości statycznej zależnością czasu życia. Właściwość statyczna jest związkiem ternarnym, ponieważ wiąże cechę, klasę i instancję. Nie jest zatem możliwe w pełni zrealizowanie wymagania (6).

Implementacja modelu przy użyciu ORM przedstawiono na rys. 2.9. Pojemność semantyczna tego metamodelu również nie pozwoliła w pełni zrealizować wszystkich wymagań.

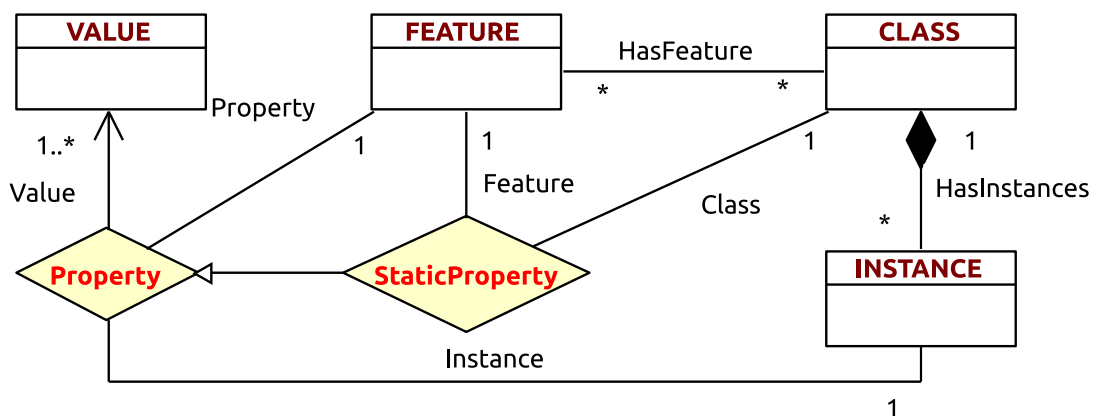
1. **Brak możliwości łączenia związków.** Związki ORM realizowane są jako typy faktów. Typy faktów stanowią realizacje  $n$ -arnych predykatów, stanowiących złączenia  $n$  ról. Nie ma możliwości wiązania ich związkami. Nie jest zatem możliwa pełna realizacja wymagania (5), które dotyczy uczestnictwa klasy, będącej związkiem (1) w związku właściwości statycznej.



Rysunek 2.9. Struktura modelu realizująca wymagania 1-7 wyrażona w metamodelu ORM.

2. **Brak możliwości dziedziczenia związków.** Jak stwierdzono powyżej, w ORM nie jest możliwe wzajemne wiązanie typów faktów, w tym związkiem generalizacja-specjalizacja. Oznacza to, iż nie jest możliwe w pełni zrealizowanie wymagania (5) w zakresie określenia związku właściwości statycznej jako specyficznego rodzaju związku właściwości.
3. **Brak zależności czasu życia** ORM nie posiada mechanizmu wiązania czasu życia obiektów. Nie jest zatem możliwe zamodelowanie wymagania (6), gdzie określono konieczność zrealizowania takiego ograniczenia w modelu danych.

Implementacja modelu w UML przedstawiono na rys. 2.10. Pojemność semantyczna tego metamodelu nie pozwoliła również w pełni, zrealizować wszystkich wymagań.



Rysunek 2.10. Struktura modelu realizująca wymagania 1-7 wyrażona w metamodelu UML.

1. **Brak mechanizmu ograniczania dziedziczalności składowych.** UML, a co za tym idzie, modelowanie obiektowe nie posiada mechanizmu ograniczania dziedziczenia

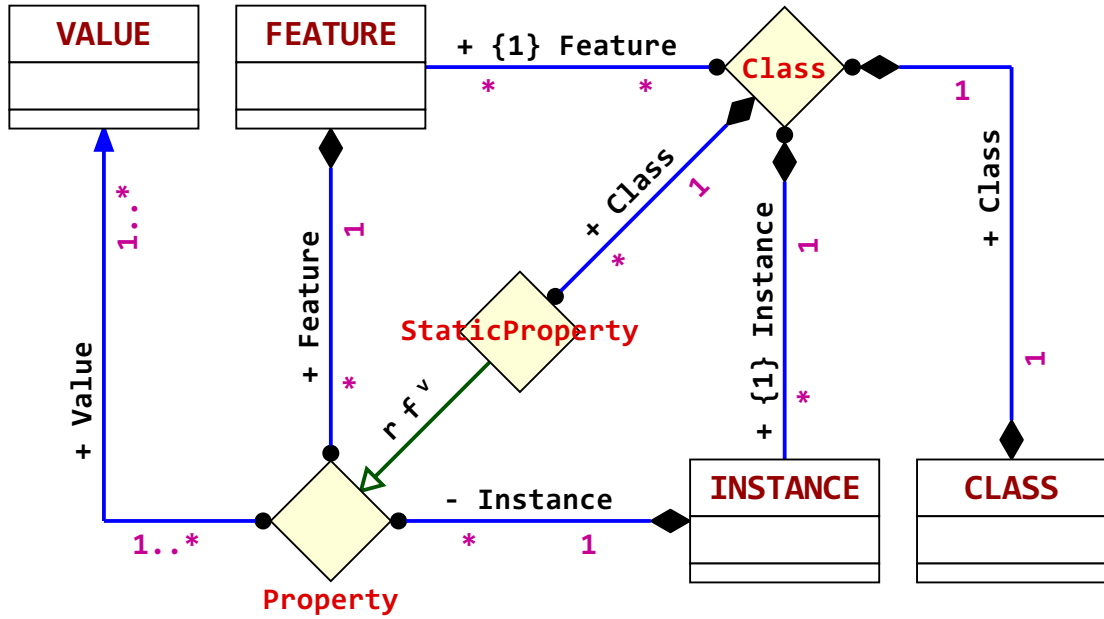
czalności. W wymaganiu (4) określono, że właściwość stanowi element łączący klasę, cechę i wartość. Następnie, w wymaganiu (5) określono, iż właściwość statyczna jest specyficznym przypadkiem właściwości, ale takim, który łączy klasę, a nie łączy instancji. Mimo istnienia mechanizmu wiązania asocjacji związkami generalizacja-specjalizacja, nie jest możliwe zaimplementowanie wymagania (5) w pełni, bez jego modyfikacji.

2. **Brak mechanizmu łączenia związków ze sobą.** Wymaganie (5) zakłada wiązanie w ramach właściwości statycznej klasy. W wymaganiu (1) określono, iż klasa jest elementem łączącym. Ze względu na brak możliwości łączenia związków pomiędzy sobą w UML, spełnienie wymagania (5) nie było możliwe.
3. **Brak możliwości wiązania czasu życia związku i encji.** W wymaganiu (6) określono, że czas życia instancji nie może być dłuży niż czas życia klasy. Oznacza to konieczność wiązania czasem życia elementu wiążącego (klasa jest elementem wiążącym – (1)) oraz elementu wiążanego – instancji. UML nie zapewnia takiego mechanizmu. Wymaganie (6) nie może być zatem w pełni zrealizowane.
4. **Brak możliwości wiązania czasem życia w związkach  $n$ -arnych.** Mechanizm określania `AggregationKind` jako `composite` w ramach `Property` w UML modelowaniu obiektowym jest ograniczony do związków binarnych. W wymaganiu (6) określono konieczność powiązania cechy, instancji, klasy i właściwości statycznej zależnością czasu życia. Właściwość statyczna jest związkiem ternarnym, ponieważ wiąże cechę, klasę i instancję. Nie jest zatem możliwe w pełni zrealizowanie wymagania (6).

Implementacja modelu w AOM przedstawiono na rys. 2.11. Każde z wymagań 1-7 mogło być spełnione bez uproszczeń, zmiany semantyki i komplikowania modelu:

1. W wymaganiu 1 klasa została zrealizowana jako klasa i kolekcja o tożsamyich nazwach połączone bikompozycyjną rolą o krotnościach *jeden-do-jeden* (wzorcem `Bicompositive Association-Collection Tandem (BACT)`, zob. 4.2.1). Część asocjacyjna klasy została połączona z kolekcją reprezentującą instancję oraz kolekcją reprezentującą cechę.
2. Wymaganie 2 określa ograniczenie związane z faktem, iż instancja może być tylko jednej klasy. Zrealizowano je ograniczając krotność wystąpień obiektów instancji w asocjacji `Class` po stronie związku.
3. Wymaganie 3 definiuje krotność roli `Feature` jako wiele. Dodatkowo, określono unikalność roli `Feature` jako `{1}`.
4. Wymaganie 4 definiuje postać właściwości. Właściwość została zrealizowana jako asocjacja ternarna pomiędzy cechą, instancją i wartością. Przyjęte w wymaganiu





Rysunek 2.11. Struktura modelu realizująca wymagania 1-7 wyrażona w metamodelu AOM.

- krotności były możliwe do zrealizowania w AOM, w tym określenie obligatoryjności dla występowania w powiązaniu co najmniej jednej wartości.
5. Podczas realizacji wymagania 5 utworzono specjalizację asocjacji *Property*: *StaticProperty*. Wykorzystując własność dziedziczalności, ograniczono dziedziczalność roli *Instance* w asocjacji *Property*. Utworzono również dodatkową rolę *Class*, którą pełni asocjacja *Class*. Okazało się to możliwe ze względu na możliwość pełnienia ról w związkach przez inne związki.
  6. Wymaganie 6 dotyczące uzależniania czasu życia rozważanych elementów modelu od siebie mogło być zrealizowane w AOM przy użyciu ról kompozycyjnych (*Feature*, *Instance*, *Class*).
  7. Wymaganie 7 zrealizowano poprzez określenie nawigowalności jako jednokierunkowej w roli *Value*.

Tablica 2.2. Podsumowanie istotnych własności wybranych metamodeli danych

		<b>EER</b>	<b>ORM</b>	<b>UML</b>	<b>AOM</b>
modelowanie bytów		✓	✓	✓	✓
	związek typ bytu – byt	✓	*	✓	✓
	związek typ bytu – atrybut	✓		✓	✓
modelowanie powiązań		✓	✓	✓	✓
	związek generalizacja-specjalizacja	✓	✓	✓	✓
	selektywność dziedziczenia atrybutów i związków				✓
	dziedziczalność składowych				✓
	wirtualność dziedziczenia				✓
	związki nazwane	✓	✓	✓	✓
	ograniczenia kategorii pełniącej rolę	entity set	entity type, value type, objectificated fact type	class	collection, association
	<i>n</i> -arność związków	✓	✓	*	✓
	dowolność w ograniczeniu krotności ról	✓	✓	✓	✓
	ograniczenia uczestnictwa w związkach	✓	✓	✓	✓
	ograniczenia liczebności powtórzeń elementów w związku		*	*	✓
	nawigowalność związków		↔	× lub ← lub ↔	← lub ↔
	ograniczanie czasu życia	*		*	✓

Z tabeli 2.2 oraz przeanalizowanego przykładu można wysnuć wniosek, iż metamodel asocjacyjny posiada pewne unikalne własności w zakresie modelowania danych w systemach informatycznych.

Symbol ✓ oznacza, że dana właściwość jest w pełni spełniona w ramach metamodelu. Symbol \* oznacza, że ta właściwość jest spełniona częściowo. Pusta komórka określa brak możliwości wykorzystania danej koncepcji modelowania w danym metamodelu. Własności te wiążą się z możliwością precyzyjnej realizacji wymagań odnoszących się do semantyki modelu. Duża pojemność semantyczna AOM pozwala bez utraty pierwotnego znaczenia oddać te wymagania w postaci modelu konceptualnego. Modelowanie asocjacyjne jako świeży kierunek badawczy zostanie w niniejszej pracy uzupełniony o propozycję metodyki modelowania asocjacyjnego poprzez wykorzystanie wzorców. Co więcej, zostanie dokonana analiza równoważności konceptualnej struktur możliwych do wyrażania w innych, wspomnianych w ramach tego rozdziału, konceptualnych metamodelach danych, która skutkować będzie możliwością rozumowania na temat implementacji modeli asocjacyjnych przy użyciu innych metamodeli. Podjęte przedsięwzięcie badawcze pozwoli lepiej umiej-

scowić AOM w dziedzinie modelowania konceptualnego. Zdaniem autora, kierunek ten doskonale wpisuje się w określone w [138] przyjęte kierunki rozwoju tej dziedziny wiedzy.

### 3. System pojęć w dziedzinie modelowania danych

Niniejszy rozdział stanowić będzie opis przyjętej konceptualizacji pojęć w dziedzinie modelowania danych, która została zdefiniowana jako słownik w języku SBVR. Opracowany system Conceptual Layer of Metamodels (CLOM) pozwala na wyrażanie konstrukcji myślowych wykorzystujących pojęcia odwzorowujące koncepcje myślowe [76]. System pojęć obejmuje nadzbiór powiązanych między sobą pojęć występujących w domenie modelowania. Każdy z metamodeli posiada swoisty zbiór pojęć, na których operuje, a przyjęty system zapewnia możliwość abstrakcji od konkretnego metamodelu. CLOM posiada własność uogólnienia niektórych pojęć zapewniając właściwy poziom konceptualizacji. Dla przykładu, w systemie przewidziano pojęcie *asocjacji*, które może przyjmować w różnych metamodelach różne postacie, np:

- asocjacja (AOM, UML),
- związek (E-R),
- typ faktu (ORM).

W literaturze znane są pewne podejścia do specyfikacji konceptualizacji, które mogłyby zostać wykorzystane do domeny modelowania. Przykładem jest metamodel Bunge-Wand-Weber (BWW) [149] oparty o ontologię Bungego [16, 17]. Mimo zaproponowanego metamodelu o nią opartego [86], ontologia ta jednakże nie spełnia potrzeb konceptualizacji systemu pojęciowego w ujęciu potrzeb niniejszej pracy, ze względu na to, iż oryginalna ontologia Bungego stanowiąca podstawy prac Wanda i Webera jest dostosowana do świata materialnego<sup>1</sup> – Bunge wyróżnia dwa typy obiektów: obiekty konkretne oraz konstrukty (do których odnosi się jako *substantial individuals*<sup>2</sup> lub *formalne, idealne* obiekty). Za Allenem [2] należy zacytować samego Bungego [16]:

*Pozostawiamy naukom formalnym, tj. logice, matematyce i semantyce zadanie badania (oraz tworzenia) formalnych lub idealnych obiektów, które można*

---

<sup>1</sup> a nie konceptualnego, jakim jest świat modelowania i metamodelowania systemów informacyjnych

<sup>2</sup> tzn. byty posiadające własności

*opisać pewnymi prawami, takich jak zbiory i kategorie. Przyjmujemy rzeczywiste (przyrodnicze lub społeczne) nauki oraz rzeczywistą ontologię za jedyne dyscypliny dotyczące konkretnych obiektów. Przypisujemy również ontologii zadanie konstruowania najbardziej ogólnych teorii dotyczących takich i tylko takich obiektów.*<sup>3</sup>

Wartą wspomnienia wydaje się być również praca [51], w której autor ustala relacje pomiędzy językiem modelowania, a ontologią referencyjną i proponuje podstawową ontologię modelowania konceptualnego. Praca zapoczątkowała bardzo ważny nurt modelowania konceptualnego: Unified Foundational Ontology (UFO). UFO to ontologia wykorzystywana do integrowania referencyjnych modeli konceptualnych w ramach wielu dziedzin, takich jak m.in.: inżynieria oprogramowania[52], usług, wiedzy prawniczej [49], modelowania agentowego [53]. UFO stanowi, w pewnym sensie, rozszerzenie podejścia Wanda i Webera i obejmuje takie zagadnienia modelowania jak typy obiektów, relacje taksonomiczne, asocjacje i zależności między asocjacjami, rolami, właściwościami i typami danych.

W pracy przyjęto, iż wykorzystanie SBVR do utworzenia konceptualizacji dziedziny modelowania danych będzie atrakcyjne z następujących powodów:

- SBVR ma siłę wyrazu zbliżoną do Web Ontology Language (OWL) [104], istnieją prace twierdzące, iż jest możliwe wzajemne odwzorowywanie obu podejść [125], a nawet twierdzi się, że „*SBVR może stanowić przyjazny interfejs dla OWL*” [84].
- SBVR posiada silne konotacje z dziedziną modelowania, gdyż został utworzony z myślą o byciu elementem MDA. Oparty jest o MOF i może być transformowalny do postaci modeli, np. UML.
- SBVR definiuje werbalizacje w kontrolowanym języku naturalnym, a co za tym idzie, jest relatywnie łatwy do interpretacji.

W literaturze nie odnaleziono żadnego podejścia opartego o SBVR<sup>4</sup>, które pozwala na możliwie swobodne podejście do konceptualizacji bez nacisku na całkowitą formalizację. Dodatkowo, przyjęte podejście opiera się o precyzyjną definicję grup ograniczeń dla specyfikowanych pojęć. Zdefiniowano również podstawowe pojęcia w domenie modelowania, które stanowią luźne nawiązanie do podejścia przedsta-

---

<sup>3</sup> Tekst oryginalny: *We leave formal science, i.e., logic, mathematics, and semantics, the task of studying (and creating) formal or ideal objects of the law abiding kind, such as sets and categories. We take factual (natural or social) science and ontology to be the only disciplines concerned with concrete objects. And we assign ontology the task of constructing the most general theories concerning such and only such objects.*

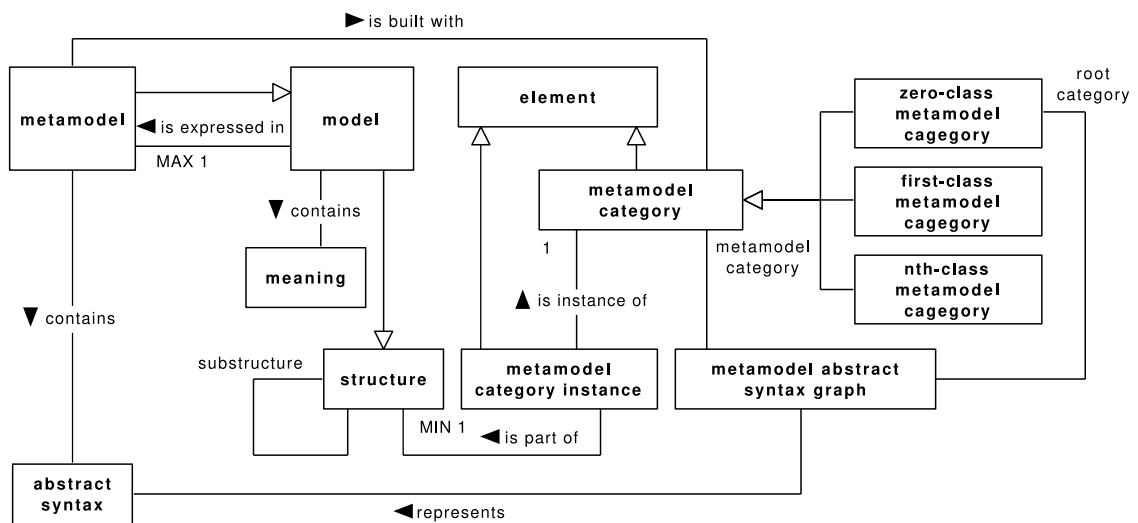
<sup>4</sup> poza samym SBVR wyrażonym przy pomocy samego siebie, lecz zdaniem autora, zdefiniowane terminy pojęciowe oraz system pojęć jest zbyt ogólny, by rozważać domenę modelowania danych

wionego w [83]. Należy również dodać, iż zaproponowane w CLoM podejście do konceptualizacji związków skupia się wyłącznie na związkach strukturalnych. Dlatego, nie uwzględniono części rodzajów związków w modelowaniu informacji (takich jak *materializacja*, *punkt widzenia*, *generacja*, por. [28]).

CLoM obejmuje zarówno pojęcia należące do warstwy intensjonalnej modelowania (typ bytu, asocjacja, cecha), jak również pojęcia należące do warstwy ekstensjonalnej (pojęcie, związek, wartość). Pojęcia zostały pogrupowane ze względu na temat, którego dotyczą, tj. pojęcia rdzenne, pojęcia związane z modelowaniem klasyfikatorów, cech, bytów, kategoryzacją pojęć, oraz powiązań. Niezależnie, wyróżniono pomocnicze grupy pojęć, tj. pojęcia podstawowe oraz związane z modelowaniem wzorców.

### 3.1. Podstawowe elementy systemu pojęciowego

Na rysunku 3.1 przedstawiono diagram pojęć SBVR zawierający elementy CLoM obrazujące podstawowe pojęcia z dziedziny modelowania. W dalszej części rozdziału umieszczono definicje tych pojęć.



Rysunek 3.1. Diagram pojęć obrazujący podstawowe pojęcia modelowania

#### element

Definicja: część składowa pewnej całości

#### model

Definicja: uproszczony obraz wycinka rzeczywistości

Pojęcie ogólne: [struktura](#)

Uwaga: Z punktu widzenia tworzenia modeli, stanowi on pewną specyficzną strukturę, której nadano określoną semantykę.

## model contains meaning

Forma synonimiczna: model zawiera semantykę

## metamodel

Definicja: model, który zawiera składnię abstrakcyjną oraz stanowi ramy opisu modeli

Przykład: AOM

Przykład: EER

Przykład: ORM

Przykład: UML

Przykład: język naturalny

Pojęcie ogólne: model

## model is expressed in metamodel

Forma synonimiczna: model jest wyrażony w ramach metamodelu

Konieczność: każdy model jest wyrażony w dokładnie jednym metamodelu

## abstract syntax

Definicja: struktura opisująca metamodel abstrahująca od reprezentacji

Synonim: składnia abstrakcyjna

## metamodel contains abstract syntax

Typ pojęcia: pojęcie czasownikowe: część-całość

Forma synonimiczna: metamodel zawiera składnię abstrakcyjną

## metamodel contains meaning

Typ pojęcia: pojęcie czasownikowe: część-całość

Forma synonimiczna: metamodel zawiera semantykę

## structure

Definicja: niepusty zbiór instancji kategorii metamodelu

Synonim: struktura

## substructure

Definicja: struktura, która jest częścią pewnej struktury

Typ pojęcia: rola

Synonim: podstruktura

Uwaga: Podstruktury pozwalają rozumować o strukturach złożonych z innych struktur. Relację pomiędzy strukturą, a podstrukturą można przyrównać do relacji bycia podzbiorem.

## metamodel category instance is part of structure

Typ pojęcia: pojęcie czasownikowe: część-całość

Forma synonimiczna: struktura posiada instancję elementu metamodelu

Forma synonimiczna: instancja elementu metamodelu jest częścią struktury

## metamodel abstract syntax graph

Definicja: ukorzeniony graf składający się z węzłów reprezentujących kategorie metamodelu oraz krawędzi odpowiadających związkom pomiędzy kategoriami metamodelu

Synonim: [graf składni abstrakcyjnej metamodelu](#)

Konieczność: **każda** [metamodel abstract syntax graph](#) **posiada** **dokładnie** **jeden** [zero-class metamodel category](#)

## **metamodel abstract syntax graph represents abstract syntax**

Forma synonimiczna: [graf składni abstrakcyjnej metamodelu](#) **reprezentuje** [składnię abstrakcyjną](#)

## **metamodel category**

Definicja: [element pewnego grafu składni abstrakcyjnej metamodelu](#), **który** jest jego węzłem

Synonim: [kategoria metamodelu](#)

Synonim: [zero-order structure](#)

Pojęcie ogólne: [element](#)

## **metamodel is built with metamodel category**

Forma synonimiczna: [metamodel](#) **jest zbudowany z** [kategorii metamodelu](#)

## **zero-class metamodel category**

Definicja: arbitralnie wskazana [kategoria metamodelu](#), **lub** pseudokategoria reprezentująca [model](#), **która** stanowi korzeń [grafu składni abstrakcyjnej metamodelu](#)

Pojęcie ogólne: [metamodel category](#)

Synonim: [kategoria metamodelu klasy zerowej](#)

Synonim: [root category](#)

Przykład: *Db* w AOM

Uwaga: Wyodrębnienie pseudokategorii reprezentującej model wynika z faktu, iż nie każdy metamodel w ramach swojej definicji zawiera kategorię klasy zerowej, lecz z praktycznego punktu widzenia zawsze można rozważać pewnego rodzaju *punkt wejścia* do samego modelu.

## **first-class metamodel category**

Definicja: [kategoria metamodelu](#), **której** odległość od [zero-class metamodel category](#) w [metamodel abstract syntax graph](#) wynosi 1

Pojęcie ogólne: [metamodel category](#)

Przykład: *Assoc* w AOM, *Class* w UML

Synonim: [kategoria metamodelu klasy pierwszej](#)

Synonim: [kategoria pierwotna metamodelu](#)

Uwaga: Kategorie pierwotne metamodelu reprezentują elementy, które są możliwe do wyrażenia bezpośrednio w samym modelu, bez konieczności udziału innych pojęć (poza [kategorią metamodelu klasy zerowej](#)).

## **second-class metamodel category**

Definicja: [kategoria metamodelu](#), **której** odległość od [zero-class metamodel category](#) w [metamodel abstract syntax graph](#) wynosi 2

Pojęcie ogólne: [metamodel category](#)

Przykład: *Role* w AOM, *Association* w UML

## **n<sup>th</sup>-class metamodel category**

Definicja: [kategoria metamodelu](#), **której** odległość od [zero-class metamodel category](#) w [metamodel abstract syntax graph](#) wynosi *n*



Pojęcie ogólne: [metamodel category](#)

Uwaga: Pojęcie to jest pojęciem generycznym, co oznacza że generuje osobne pojęcie  $\forall n \in \mathbb{N}$ .

### metamodel category instance

Definicja: [element będący częścią struktury](#) i [będący instancją kategorii metamodelu](#)

Synonim: [instancja kategorii metamodelu](#)

Pojęcie ogólne: [element](#)

### metamodel category instance is instance of metamodel category

Forma synonimiczna: [instancja kategorii metamodelu](#) *jest instancją* [kategorii metamodelu](#)

## 3.2. Abstrakcyjność pojęciowa i konkretność pojęć modelowania

Aby przy pomocy terminów wyrażać pojęcia związane z dziedziną modelowania w sposób konkretny, należy zadbać o to, by zdefiniowany system pojęciowy operował na wszystkich jego warstwach (rys. 2.1), np.:

- (M3) – aby wyrażać pojęcia w zakresie metamodelu (np. `Class` w MOF),
- (M2) – aby wyrażać pojęcia w zakresie metamodelu (np. `Class` w UML),
- (M1) – aby wyrażać pojęcia w zakresie modelu (konkretna klasa, np. `Person`),
- (M0) – by móc odnieść się do konkretnych danych opisanych przez model (np. obiekt klasy `Person` reprezentujący konkretną osobę).

Na potrzeby CLoM, z dbałości o zachowanie jednoznaczności przyjęto podejście generatywne do tworzenia nowych pojęć, realizujące powstawanie zależności między pojęciami definiowanymi w modelowaniu jako związek *instanceOf*. Pojęcie stanowiące *instancję* w rozumieniu związku *instanceOf* pewnej kategorii modelowania będzie nazywane *konkretyzacją* tej kategorii. Analogicznie, związek przeciwny został tutaj nazwany *abstrakcją pojęciową*. Nomenklaturę taką zastosowano, aby uniknąć zbieżności terminologicznej związanej z przyjętym systemem konceptualizacji (SBVR), który instancję definiuje jako: [thing that is in an extension of a concept](#). Oznacza to, że ekstensją ([extension](#)) abstrakcyjnych pojęć posiadających semantykę [pojęć modelowania](#) jest ogół [kategorii metamodeli](#) charakteryzowanych poprzez spełnienie funkcjonalności i ograniczeń zdefiniowanych przez dane [pojęcie modelowania](#). Przykładowo, do ekstensji pojęcia [asocjacji](#) należy kategoria modelowania *Association* w języku UML, *typ faktu* w ORM oraz *Assoc* w AOM. Niezależnie, każda z tych kategorii stanowi instancję pojęcia jednostkowego, odpowiednio [association \(UML\)](#), [fact type \(ORM\)](#) [Assoc \(AOM\)](#).

*Konkretyzacje* tych pojęć odnoszą się do wystąpień (instancji – w sensie modelowania) [kategorii modelowania](#), które wchodzą w skład [struktur](#). Naturalnie, relacja konkretyzacji występuje na każdym styku poszczególnych warstw w dziedzinie

modelowania, tzn. M0-M1, M1-M2, M2-M3. W zakresie pojęć na styku warstw M1-M2, dla pojęcia asocjacji należy rozważyć pojęcie konkretnej asocjacji definiowanej jako: pojęcie modelowania, które jest konkretyzacją asocjacji. Do ekstensji konkretnej asocjacji należą wszelkie asocjacje, które w dowolnym świecie możliwym zostałyby zamodelowane, dla przykładu asocjacja a2 umieszczona w modelu wyrażonym na diagramie przedstawionym na okładce publikacji [88]. Przykłady stosowania *abstrakcji pojęciowej* i *konkretyzacji* na styku warstw M2-M1 w modelowaniu asocjacyjnym zobrazowano w tabeli 3.1 podsumowującej powyższe rozważania.

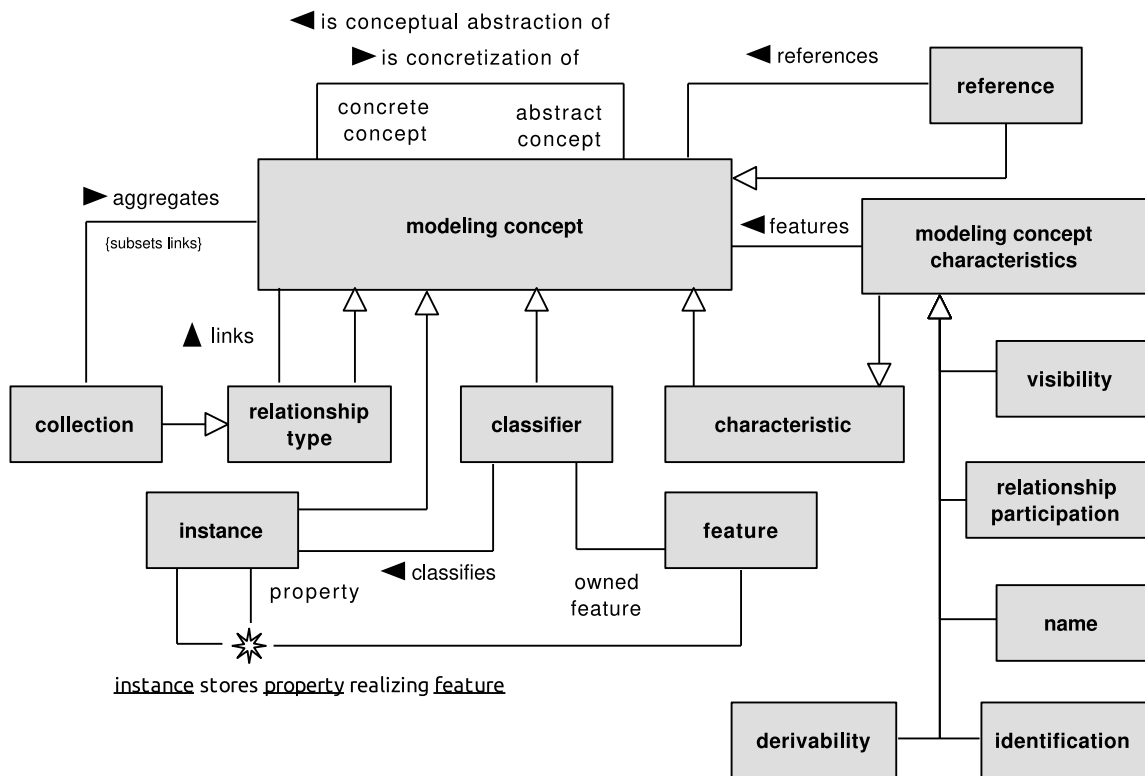
Tablica 3.1. Tabela obrazująca zależności między pojęciami abstrakcyjnymi, konkretnymi i ich ekstensjami

	pojęcie ogólne	pojęcie jednostkowe	ekstensja pojęcia ogólnego	ekstensja pojęcia jednostkowego
abstrakcyjne	<u>asocjacja</u>	<u>Assoc (AOM)</u>	ogół wszystkich kategorii modelowania we wszystkich metamodelach danych realizujących semantykę nazwanego związku pomiędzy innymi kategoriami	kategoria asocjacji zdefiniowana w ramach AOM
konkretne	<u>konkretna asocjacja</u> zdefiniowana jako <u>pojęcie modelowania</u> , które jest konkretyzacją <u>asocjacji</u>	konkretna asocjacja w modelu asocjacyjnym zdefiniowana jako <u>pojęcie modelowania</u> odnoszące się do asocjacji a2 z okładki [88], które jest konkretyzacją <u>Assoc (AOM)</u>	ogół wszystkich instancji kategorii modelowania utworzonych przez podmioty modelujące systemy informatyczne realizujących semantykę połączenia pomiędzy konkretnymi kategoriami	asocjacja a2 z okładki [88]

### 3.3. Definicja uogólnionego zbioru mechanizmów i właściwości modelowania

#### 3.3.1. Pojęcia rdzenne

Na rysunku 3.2 przedstawiono diagram pojęć rdzennych CLoM. Diagram obrazuje podstawowe pojęcie modeling concept będące pojęciem ogólnym dla pojęć reprezentujących główne aspekty modelowania danych, takie jak klasyfikatory, typy związków, instancje, cechy, charakterystyki i referencje. Dodatkowo, na diagramie przedstawiono podstawowe charakterystyki pojęć realizujące ich ograniczenia i właściwości.



Rysunek 3.2. Diagram pojęć przedstawiający powiązania między rdzennymi pojęciami modelowania

## modeling concept

Definicja: [pojęcie](#) wykorzystywane w modelowaniu

Uwaga: Pojęcie modelowania stanowi podstawowy budulec metamodelu występujący wraz z powiązaniem między innymi pojęciami modelowania. Pojęcie modelowania może obejmować zarówno mechanizmy definiowania struktur, aspekty behawioralne, jak również ograniczenia zdefiniowane w ramach metamodeli.

Pojęcie ogólne: [pojęcie](#)

Synonim: [pojęcie modelowania](#)

## abstract concept is conceptual abstraction of concrete concept

Forma synonimiczna: [concrete concept](#) is concretization of [abstract concept](#)

Forma synonimiczna: [pojęcie abstrakcyjne](#) jest abstrakcją [pojęciową](#) [pojęcia](#) konkretnego

Forma synonimiczna: [pojęcie konkretne](#) jest konkretyzacją [pojęcia](#) abstrakcyjnego

## abstract concept

Definicja: [pojęcie modelowania](#) będące na meta-poziomie względem wszystkich [pojęć modelowania](#), które są jego [konkretyzacją](#)

Synonim: [pojęcie abstrakcyjne](#)

Typ pojęcia: [rola](#)

## concrete concept

Definicja: [pojęcie modelowania](#), dla którego istnieje [pojęcie modelowania](#) na meta-poziomie będące jego [abstrakcją](#)

Synonim: [pojęcie konkretne](#)

Typ pojęcia: [rola](#)

### **classifier**

Definicja: [pojęcie modelowania](#) posiadające [cechy](#) umożliwiając klasyfikowanie [instancji](#)

Synonim: [klasyfikator](#)

Pojęcie ogólne: [pojęcie modelowania](#)

### **feature**

Definicja: [pojęcie modelowania](#) służące do opisywania [klasyfikatorów](#) i realizowane poprzez [właściwości](#) w kontekście [instancji](#)

Synonim: [cecha](#)

Pojęcie ogólne: [pojęcie modelowania](#)

### **classifier has owned feature**

Forma synonimiczna: [klasyfikator](#) *posiada* [inherentną cechę](#)

### **owned feature**

Typ pojęcia: [rola](#)

Definicja: [cecha posiadana przez pewien](#) [klasyfikator](#)

Synonim: [inherentna cecha](#)

### **instance**

Definicja: [pojęcie modelowania](#) *należące do* warstwy ekstensjonalnej modelowania [klasyfikowane](#) *przez* [klasyfikator](#)

Synonim: [instancja](#)

Pojęcie ogólne: [pojęcie modelowania](#)

### **classifier classifies instance**

Definicja: [instancja](#) *posiada* [właściwości, które realizują wszystkie](#) [cechy posiadane przez](#) [klasyfikator](#)

Forma synonimiczna: [klasyfikator](#) *klasyfikuje* [instancję](#)

Forma synonimiczna: [instancja](#) *jest klasyfikowana przez* [klasyfikator](#)

### **property**

Definicja: [instancja, która jest realizacją](#) [cechy przechowującej](#) ją [instancji](#)

Typ pojęcia: [role](#)

Synonim: [właściwość](#)

### **instance stores property realizing feature**

Definicja: [właściwość](#) wchodzi w skład opisu stanu [instancji](#) poprzez zapewnienie realizacji [cechy](#)

Forma synonimiczna: [instancja](#) *przechowuje* [właściwość realizującą](#) [cechę](#)

Forma synonimiczna: [cecha](#) *jest realizowana przez* [właściwość przechowywaną przez](#) [instancję](#)

Konieczność: [cecha posiadana przez](#) [klasyfikator](#) *klasyfikujący daną* [instancję](#) *musi być realizowana* *przez* [właściwość przechowywaną przez](#) [daną](#) [instancję](#)

### **characteristic**

Definicja: [pojęcie modelowania](#), *które* specyfikuje ograniczenia i mechanizmy zdefiniowane w [metamodelach](#)

Definicja: [charakterystyka pojęcia modelowania](#) lub [charakterystyka roli](#) lub [charakterystyka klasyfikatora](#) lub [charakterystyka atrybutu](#) lub [charakterystyka cechy](#) lub [charakterystyka kategoryzacji](#)

Synonim: [charakterystyka](#)

Pojęcie ogólne: [pojęcie modelowania](#)

### **modeling concept characteristic**

Definicja: [widoczność](#) lub [uczestnictwo w związku](#) lub [nazwa](#) lub [identyfikacja](#) lub [pochodność](#)

Synonim: [charakterystyka pojęcia modelowania](#)

Pojęcie ogólne: [charakterystyka](#)

### **modeling concept characteristic features modeling concept**

Forma synonimiczna: [charakterystyka pojęcia modelowania](#) *cehuje* [pojęcie modelowania](#)

### **visibility**

Definicja: [charakterystyka pojęcia modelowania](#), która ogranicza możliwość odwoływania się do [danego pojęcia modelowania](#) ze względu na kontekst odwołania

Synonim: [widoczność](#)

Pojęcie ogólne: [charakterystyka pojęcia modelowania](#)

### **relationship participation**

Definicja: [charakterystyka pojęcia modelowania](#), która ogranicza możliwość uczestnictwa w określonych rolach instancji [relationship type](#)

Synonim: [uczestnictwo w związku](#)

Pojęcie ogólne: [charakterystyka pojęcia modelowania](#)

### **name**

Definicja: [charakterystyka pojęcia modelowania](#), która zadaje możliwość odwoływania się do danego [pojęcia modelowania](#)

Synonim: [nazwa](#)

Pojęcie ogólne: [charakterystyka pojęcia modelowania](#)

### **identification**

Definicja: [charakterystyka pojęcia modelowania](#), która zapewnia możliwość użycia [go](#) do zidentyfikowania innego [pojęcia modelowania](#)

Synonim: [identyfikacja](#)

Pojęcie ogólne: [charakterystyka pojęcia modelowania](#)

### **derivability**

Definicja: [charakterystyka pojęcia modelowania](#), która określa, iż [pojęcie modelowania](#) ma charakter wyliczeniowy, pozyskany poprzez analizę innych [pojęć modelowania](#)

Synonim: [pochodność](#)

Pojęcie ogólne: [charakterystyka pojęcia modelowania](#)

### **reference**

Definicja: [pojęcie modelowania](#) stanowiące wskazanie pewnego [pojęcia modelowania](#)

Uwaga: Referencja umożliwia wielokrotne odwoływanie się do tego samego pojęcia bez konieczności multiplikacji instancji pojęcia.

Synonim: [referencja](#)

Pojęcie ogólne: [pojęcie modelowania](#)

### **[reference](#) [references](#) [modeling](#) [concept](#)**

Forma synonimiczna: [referencja](#) *odnosi się do* [pojęcia modelowania](#)

Forma synonimiczna: [referencja](#) *do* [pojęcia modelowania](#)

### **[relationship](#) [type](#)**

Definicja: [pojęcie modelowania](#) określające specyfikę powiązania pomiędzy [pojęciami modelowania](#)

Definicja: [kategoryzacja](#) *lub* [asocjacja](#) *lub* [kolekcja](#)

Synonim: [typ związku](#)

Pojęcie ogólne: [pojęcie modelowania](#)

### **[relationship](#) [type](#) [links](#) [modeling](#) [concept](#)**

Forma synonimiczna: [typ związku](#) *łączy* [pojęcie modelowania](#)

### **[collection](#)**

Definicja: [typ związku](#) umożliwiający zbiorczą [referencję](#) do innych [pojęć modelowania](#)

Synonim: [zbiorowość](#)

Synonim: [kolekcja](#)

Pojęcie ogólne: [typ związku](#)

### **[collection](#) [aggregates](#) [modeling](#) [concept](#)**

Definicja: [kolekcja](#) *łączy* [pojęcie modelowania](#)

Forma synonimiczna: [zbiorowość](#) *agreguje* [pojęcie modelowania](#)

Uwaga: Łączenie pojęć modelowania przez zbiorowości ma charakter umożliwiający dalsze rozumowanie na ich temat w taki sposób, iż konkretne zestawy pojęć reprezentują indywidualnie wyróżnione z uniwersum. W przypadku stwierdzenia faktu, odwołanie do zbiorowości zakłada odniesienie się do zbiorowości jako całości, nie zaś do jej poszczególnych elementów.

## **3.3.2. Pojęcia związane z modelowaniem klasyfikatorów**

Na rysunku 3.3 przedstawiono diagram pojęć obrazujący pojęcia modelowania związane z modelowaniem klasyfikatorów. W szczególności, wprowadzono dedykowane charakterystyki klasyfikatora, takie jak nieinstancjonalność oraz identyfikator.

### **[classifier](#) [characteristic](#)**

Definicja: [nieinstancjonalność](#) *lub* [identyfikator](#)

Synonim: [charakterystyka klasyfikatora](#)

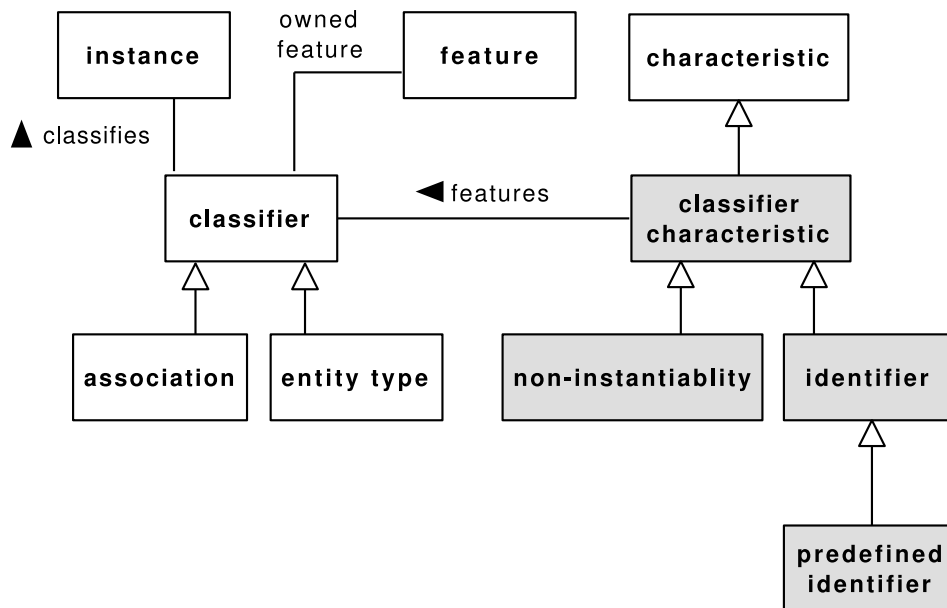
Pojęcie ogólne: [charakterystyka](#)

### **[classifier](#) [characteristic](#) [features](#) [classifier](#)**

Forma synonimiczna: [charakterystyka klasyfikatora](#) *cehuje* [klasyfikator](#)

### **[non-instantiability](#)**

Definicja: [charakterystyka klasyfikatora](#), *która* pozwala na definiowanie [konkretnego klasyfikatora](#) nieklasyfikującego [instancji](#) bezpośrednio



Rysunek 3.3. Diagram pojęć przedstawiający powiązania między pojęciami modelowania związanych z modelowaniem klasyfikatorów

Uwaga: *Bezpośredniość* w definicji dotyczy konieczności istnienia innego klasyfikatora  $k_2$ , który będzie powiązany z danym klasyfikatorem  $k_1$  (np. poprzez związek kategoryzacji, stanowiąc jego podtyp) oraz założenia, że  $k_2$  jest instancjonowalny. Wtedy należy założyć, że  $k_1$  *pośrednio* klasyfikuje instancje  $k_2$ .

Synonim: [nieinstancjonowalność](#)

Pojęcie ogólne: [charakterystyka klasyfikatora](#)

### identifier

Definicja: [charakterystyka klasyfikatora](#), która określa, czy [konkretny klasyfikator](#) *posiada* unikalny [element](#) lub mechanizm, który jednoznacznie definiuje [ten konkretny klasyfikator](#)

Synonim: [identyfikator](#)

Pojęcie ogólne: [charakterystyka klasyfikatora](#)

### predefined identifier

Definicja: [identyfikator](#), który określa, czy [każdy konkretny klasyfikator](#) *posiada* unikalny [element](#) lub mechanizm, który jednoznacznie definiuje [ten konkretny klasyfikator](#)

Synonim: [predefiniowany identyfikator](#)

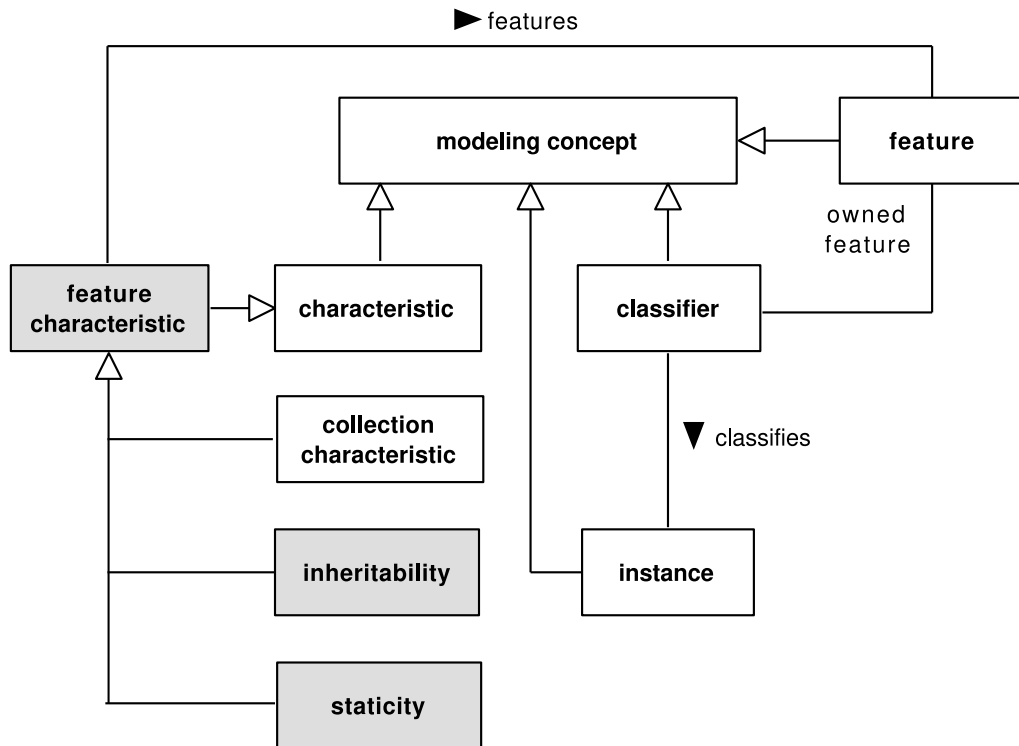
Pojęcie ogólne: [identyfikator](#)

### 3.3.3. Pojęcia związane z modelowaniem cech

Na rysunku 3.4 przedstawiono fragment CLoM odnoszący się do modelowania cech. W szczególności, przedstawiono charakterystyki cech, takie jak dziedziczalność oraz statyczność.

#### feature characteristic

Definicja: [dziedziczalność](#) lub [statyczność](#) lub [charakterystyka zbiorowości](#)



Rysunek 3.4. Diagram pojęć przedstawiający powiązania między pojęciami modelowania związanymi z modelowaniem cech

Synonim: [charakterystyka cechy](#)

Pojęcie ogólne: [charakterystyka](#)

### [feature characteristics](#) [features](#) [feature](#)

Forma synonimiczna: [charakterystyka cechy](#) [cechuje](#) [cechę](#)

### [inheritability](#)

Definicja: [charakterystyka cechy](#), która określa czy [dana cecha](#) może podlegać [dziedziczeniu](#)

Synonim: [dziedziczalność](#)

Pojęcie ogólne: [charakterystyka cechy](#)

### [staticity](#)

Definicja: [charakterystyka cechy](#), która określa czy [właściwość realizująca daną cecha](#) jest wspólna dla [każdej instancji](#), która tę [cechę przechowuje](#)

Synonim: [statyczność](#)

Pojęcie ogólne: [charakterystyka cechy](#)

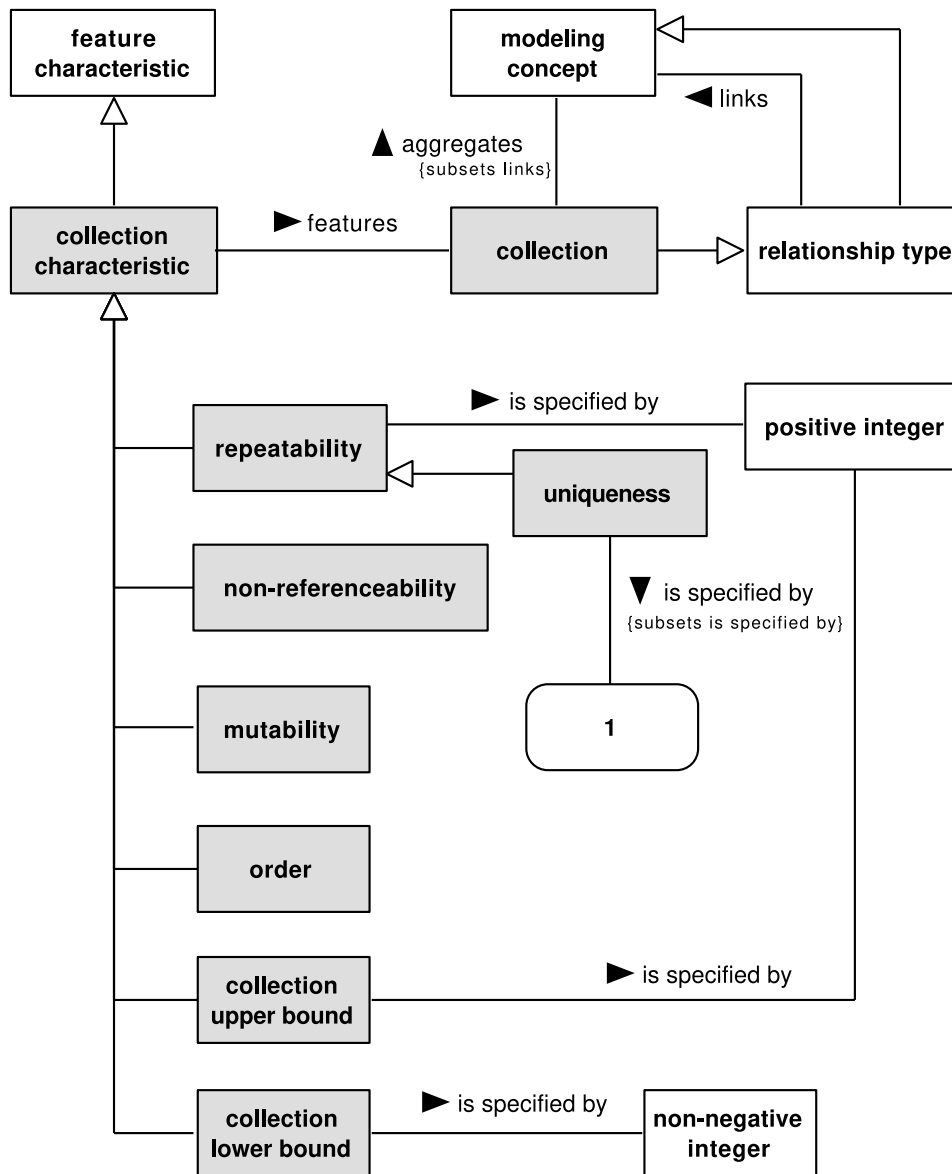
Uwaga: W przypadku, gdy cecha jest statyczna, zakłada się, iż posiada jedną właściwość, która jest charakterystyczna dla klasyfikatora, a jego instancje posiadają referencję do tej właściwości.

#### 3.3.4. Pojęcia związane z modelowaniem zbiorowości

Na rysunku 3.5 przedstawiono fragment CLoM odnoszący się do modelowania zbiorowości. W szczególności, przedstawiono takie charakterystyki zbiorowości, jak



powtarzalność, niereferencyjność, zmienność, porządek oraz ograniczenia w postaci kresów liczności zbiorowości.



Rysunek 3.5. Diagram pojęć przedstawiający powiązania między pojęciami modelowania związanymi z modelowaniem zbiorowości

### collection characteristic

Definicja: [charakterystyka cechy](#) określająca ograniczenie związane z modelowaniem [zbiorowości](#)

Synonim: [charakterystyka zbiorowości](#)

Pojęcie ogólne: [charakterystyka cechy](#)

### order

Definicja: [charakterystyka kolekcji](#), która zakłada istnienie funkcji określającej porządek [elementów danej zbiorowości](#)

Synonim: [kolejność](#)

Synonim: [porządek](#)

Pojęcie ogólne: [charakterystyka kolekcji](#)

### **mutability**

Definicja: [charakterystyka zbiorowości](#), która zakłada możliwość dodawania, usuwania lub modyfikacji elementów [zbiorowości](#)

Synonim: [zmiennosc](#)

Synonim: [mutowalnosc](#)

Pojęcie ogólne: [charakterystyka kolekcji](#)

### **repeatability**

Definicja: [charakterystyka zbiorowości](#), która ogranicza możliwość wielokrotnego występowania tej samej [własności](#) w [danej cesze](#)

Uwaga: Szczególnym przypadkiem powtarzalności jest unikalność, która zakłada, że w danej zbiorowości własność może wystąpić maksymalnie jednokrotnie. Jednakże, w ogólnym przypadku, własność ta jest odzwierciedlana przez pewną liczbę naturalną determinującą maksymalną ilość powtórzeń.

Synonim: [powtarzalnosc](#)

Pojęcie ogólne: [charakterystyka kolekcji](#)

### **repeatability is specified by positive integer**

Forma synonimiczna: [powtarzalnosc jest określona przez dodatnią liczbę całkowitą](#)

### **uniqueness**

Definicja: [powtarzalnosc](#), która ogranicza liczbę wstąpień tej samej [własności](#) w [danej cesze](#) do 1

Definicja: [powtarzalnosc](#), która jest określona przez 1

Synonim: [unikalnosc](#)

Pojęcie ogólne: [powtarzalnosc](#)

### **non-referenceability**

Definicja: [charakterystyka zbiorowości](#), która zabrania [przechowywania](#) przez [daną instancję wartości](#), które są [referencjami](#)

Synonim: [niereferencyjnosc](#)

Pojęcie ogólne: [charakterystyka zbiorowości](#)

### **collection lower bound**

Definicja: [charakterystyka zbiorowości](#), która ogranicza minimalną liczbę wystąpień [własności](#) w [danej cesze](#)

Synonim: [kres dolny zbiorowości](#)

### **collection lower bound is specified by non-negative integer**

Forma synonimiczna: [kres dolny zbiorowości jest określony przez nieujemną liczbę całkowitą](#)

### **collection upper bound**

Definicja: [charakterystyka zbiorowości](#), która ogranicza maksymalną liczbę [własności](#) w [danej cesze](#)

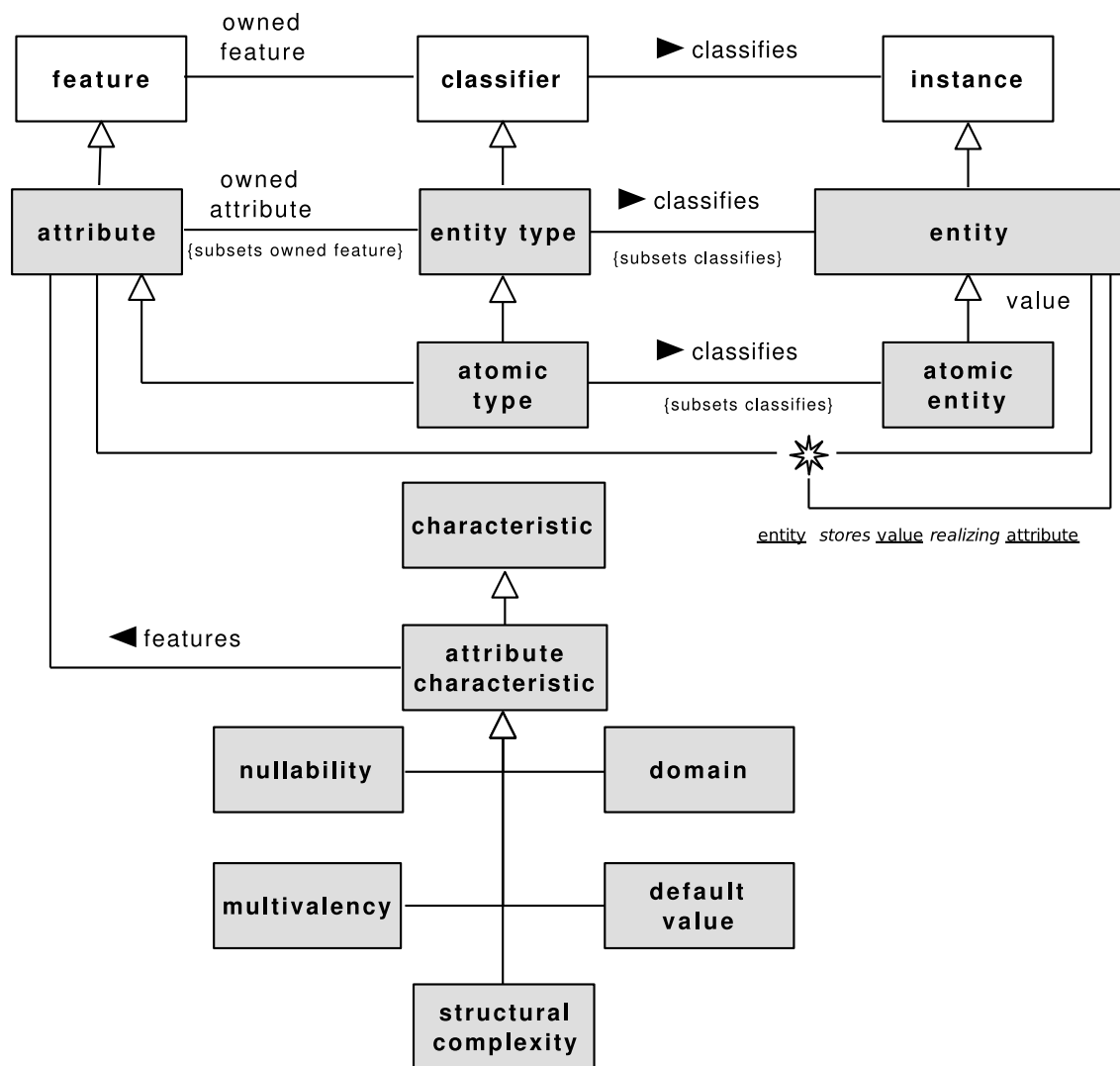
Synonim: [kres górny zbiorowości](#)

## collection upper bound is specified by positive integer

Forma synonimiczna: kres górny zbiorowości jest określony przez dodatnią liczbę całkowitą

### 3.3.5. Pojęcia związane z modelowaniem bytów

Na rysunku 3.6 przedstawiono fragment CLoM odnoszący się do modelowania bytów. W szczególności, przedstawiono pojęcia takie jak byt, typ bytu, atrybut, wartość oraz szereg charakterystyk dotyczących atrybutów.



Rysunek 3.6. Diagram pojęć przedstawiający powiązania między pojęciami modelowania związanymi z modelowaniem bytów

### entity

Definicja: instancja, która może przechowywać wartości oraz może być klasyfikowana przez typ bytu

Synonim: byt

Pojęcie ogólne: instancja

## entity type

Definicja: klasyfikator, który może *posiadać* atrybuty oraz *klasyfikuje* byt

Synonim: typ bytu

Pojęcie ogólne: klasyfikator

## entity type classifies entity

Forma synonimiczna: typ bytu *klasyfikuje* byt

Forma synonimiczna: byt *jest klasyfikowany przez* typ bytu

## attribute

Definicja: cecha, która może *być posiadana przez* typ bytu oraz może *być realizowana przez* wartość

Synonim: atrybut

Pojęcie ogólne: cecha

## entity type has owned attribute

Forma synonimiczna: typ bytu *posiada* inherentny atrybut

Forma synonimiczna: inherentny atrybut *jest posiadany przez* typ bytu

## owned attribute

Typ pojęcia: role

Definicja: atrybut, który *jest częścią* typu bytu

Synonim: inherentny atrybut

## value

Typ pojęcia: role

Definicja: właściwość, która może *być przechowywana przez* byt oraz może *realizować* atrybut

Synonim: wartość

Pojęcie ogólne: właściwość

## entity stores value realizing attribute

Forma synonimiczna: byt *przechowuje* wartość realizującą atrybut

Forma synonimiczna: wartość *realizuje atrybut i jest przechowywana przez* byt

## atomic type

Definicja: typ bytu, który *nie posiada* inherentnych atrybutów

Synonim: typ atomowy

## atomic entity

Definicja: byt, który *jest klasyfikowany przez* atomowy typ

Uwaga: W zależności od przyjętej konceptualizacji w ramach metamodelu, instancjami tego pojęcia mogą być wartości typów prostych, takie jak: 7, 3.14, M, *prawda*, etc. albo bardziej złożone byty, takie jak ciągi znaków, jeśli przyjęto, że są traktowane jako niepodzielne.

Synonim: byt atomowy

## attribute characteristic

Definicja: zerowalność lub wielowartościowość lub typ lub wartość domyślna lub strukturalna złożoność

Synonim: [charakterystyka atrybutu](#)

Pojęcie ogólne: [charakterystyka cechy](#)

### **attribute characteristic features attribute**

Forma synonimiczna: [charakterystyka atrybutu](#) [cechuje](#) [atrybut](#)

### **domain**

Definicja: [charakterystyka atrybutu](#), która określa [zbiór wartości](#), które mogą [realizować dany atrybut](#) oraz mogą być [przechowywane przez byt](#)

Synonim: [domena](#)

Pojęcie ogólne: [charakterystyka atrybutu](#)

### **nullability**

Definicja: [charakterystyka atrybutu](#), która określa, czy [wartości realizujące dany atrybut](#) mogą przyjmować wartość null

Uwaga: null reprezentuje brak wartości lub innymi słowy wartość o nieokreślonej semantyce

Synonim: [zerowalność](#)

Pojęcie ogólne: [charakterystyka atrybutu](#)

### **multivalency**

Definicja: [charakterystyka atrybutu](#), która określa, czy [byty klasyfikowane przez typ bytu posiadający dany atrybut](#), mogą [przechowywać więcej niż jedną wartość realizującą dany atrybut](#)

Synonim: [wielowartościowość](#)

Pojęcie ogólne: [charakterystyka atrybutu](#)

### **structural complexity**

Definicja: [charakterystyka atrybutu](#), która określa, czy [domeną cechującą dany atrybut](#) może być [typ bytu](#), który nie jest typem atomowym

Synonim: [strukturalna złożoność](#)

Pojęcie ogólne: [charakterystyka atrybutu](#)

### **default value**

Pojęcie ogólne: [charakterystyka atrybutu](#)

Definicja: [charakterystyka atrybutu](#), która umożliwia określenie predefiniowanej [wartości realizującej dany atrybut](#) dla [wszystkich bytów](#), dla których nie określono danej [wartości](#)

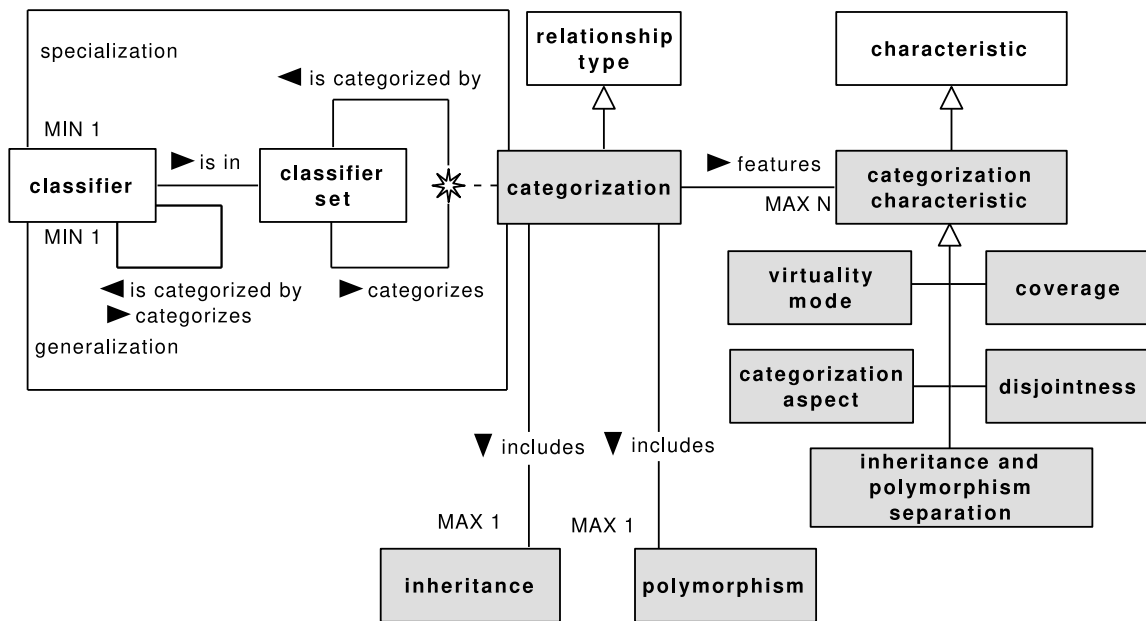
Synonim: [wartość domyślna](#)

## **3.3.6. Pojęcia związane z kategoryzacją pojęć**

Na rysunku 3.7 przedstawiono fragment CLoM odnoszący się do modelowania związków kategoryzujących. W szczególności, przedstawiono takie aspekty związków tego typu jak dziedziczenie, polimorfizm oraz szereg charakterystyk opisujących ograniczenia związane z modelowaniem związków o charakterze generalizacji-specjalizacji.

### **classifier<sub>1</sub> categorizes classifier<sub>2</sub>**

Definicja: [klasyfikator<sub>1</sub>](#) jest [generalizacją w ramach danej kategoryzacji](#) oraz [klasyfikator<sub>2</sub>](#) jest [specjalizacją w ramach danej kategoryzacji](#)



Rysunek 3.7. Diagram pojęć przedstawiający powiązania między pojęciami modelowania związanych z modelowaniem kategoryzacji

Forma synonimiczna: [klasyfikator<sub>1</sub>](#) *kategoryzuje* [klasyfikator<sub>2</sub>](#)

Forma synonimiczna: [klasyfikator<sub>2</sub>](#) *jest kategoryzowany przez* [klasyfikator<sub>1</sub>](#)

### classifier set

Definicja: [zbiór](#), którego [elementami są](#) [klasyfikatory](#)

Pojęcie ogólne: [zbiór](#)

Synonim: [zbiór klasyfikatorów](#)

### classifier set<sub>1</sub> *categorizes* classifier set<sub>2</sub>

Definicja: [sytuacja](#), taka że [każdy](#) [klasyfikator](#) [będący w](#) [zbiornie](#) [klasyfikatorów<sub>1</sub>](#) *jest kategoryzowany przez* [każdy](#) [klasyfikator](#) [będący w](#) [zbiornie](#) [klasyfikatorów<sub>2</sub>](#)

Konieczność: [classifier set<sub>1</sub>](#) i [classifier set<sub>2</sub>](#) są rozłączne

Forma synonimiczna: [zbiór klasyfikatorów<sub>1</sub>](#) *kategoryzuje* [zbiór klasyfikatorów<sub>2</sub>](#)

Forma synonimiczna: [zbiór klasyfikatorów<sub>2</sub>](#) *jest kategoryzowany przez* [zbiór klasyfikatorów<sub>1</sub>](#)

Uwaga: Przyjęto, iż każdy klasyfikator jest równoważny semantycznie z jednoelementowym zbiorem zawierającym ten klasyfikator. A zatem wyrażenie [klasyfikator<sub>k</sub>](#) *kategoryzuje* [zbiór klasyfikatorów<sub>1</sub>](#) jest równoważne wyrażeniu [zbiór klasyfikatorów](#) *zawierający* [klasyfikator<sub>k</sub>](#) *kategoryzuje* [zbiór klasyfikatorów<sub>1</sub>](#).

### categorization

Definicja: [faktyczna sytuacja](#), taka że [zbiór klasyfikatorów<sub>1</sub>](#) *kategoryzuje* [zbiór klasyfikatorów<sub>2</sub>](#)

Definicja: [typ związku](#), który *zawiera* [dziedziczenie](#) *lub* [polimorfizm](#)

Synonim: [kategoryzacja](#)

Pojęcie ogólne: [typ związku](#)

### taxonomy

Definicja: skierowany graf acykliczny, którego węzłami są instancje [konkretnych klasyfikatorów](#), a krawędziami instancje [konkretnych kategoryzacji](#)

Synonim: [taksonomia](#)

### **categorization path**

Definicja: ścieżka w [taksonomii](#)

Uwaga: Termin ścieżka odnosi się do pojęcia ścieżki w teorii grafów.

Synonim: [ścieżka kategoryzacji](#)

### **generalization**

Definicja: [klasyfikator](#), który w danej [konkretnej kategoryzacji](#) posiada mniejszą odległość od ujścia [taksonomii](#) niż [specjalizacja posiadana przez daną konkretną kategoryzację](#)

Typ pojęcia: [role](#)

Synonim: [generalizacja](#)

Synonim: [kategoria bazowa](#)

### **specialization**

Definicja: [klasyfikator](#), który w danej [konkretnej kategoryzacji](#) posiada większą odległość od ujścia [taksonomii](#) niż [specjalizacja posiadana przez daną konkretną kategoryzację](#)

Typ pojęcia: [role](#)

Synonim: [specjalizacja](#)

### **categorization *has* generalization**

Forma synonimiczna: [kategoryzacja posiada generalizację](#)

Forma synonimiczna: [generalizacja jest generalizacją w ramach kategoryzacji](#)

### **categorization *has* specialization**

Forma synonimiczna: [kategoryzacja posiada specjalizację](#)

Forma synonimiczna: [specjalizacja jest specjalizacją w ramach kategoryzacji](#)

### **inheritance**

Definicja: mechanizm pozyskiwania [cech](#) w [kategoryzacji](#) przez [specjalizację](#) [posiadaną przez daną kategoryzację](#) z [generalizacji](#) [posiadanej przez daną kategoryzację](#)

Synonim: [dziedziczenie](#)

### **acquired feature**

Pojęcie ogólne: [cecha](#)

Forma synonimiczna: [pozyskana cecha](#)

### **acquired feature *originates in* feature**

Forma synonimiczna: [pozyskana cecha ma początek w cesze](#)

Uwaga: Pojęcie pozwala wyrazić sytuację, w której poprzez mechanizm dziedziczenia w ramach pewnej taksonomii cechy są przekazywane na kolejnych poziomach hierarchii. Cecha biorąca udział w tym wyrażeniu, jest cechą, która stanowi źródło pozyskiwania cechy w ramach całej taksonomii.

## original feature

Synonim: cecha źródłowa

Definicja: cecha, w której *ma początek* pewna cecha w ramach taksonomii

## categorization includes inheritance

Forma synonimiczna: kategoryzacja zawiera dziedziczenie

Konieczność: *każda* kategoryzacja zawiera co najwyżej jedno dziedziczenie

## polymorphism

Definicja: mechanizm zastępowalności generalizacji posiadanej przez daną kategoryzację przez specjalizację posiadaną przez daną kategoryzację

Uwaga: Zastępowalność rozumie się jako możliwość podstawienia elementu zastępującego, w miejsce elementu zastępowanego w danym kontekście. Podstawienie może spowodować zmiany na poziomie gramatycznym lub semantycznym.

Synonim: polimorfizm

## specialization is substituted for generalization

Forma synonimiczna: specjalizacja jest podstawiona za generalizację

## categorization includes polymorphism

Forma synonimiczna: kategoryzacja zawiera polimorfizm

Konieczność: *każda* kategoryzacja zawiera co najwyżej jeden polimorfizm

## categorization characteristic

Definicja: tryb wirtualności *lub* aspekt kategoryzacji *lub* pokrycie *lub* rozłączność *lub* selektywne dziedziczenie własności

Pojęcie ogólne: charakterystyka

Synonim: charakterystyka kategoryzacji

## categorization characteristic features categorization

Forma synonimiczna: kategoryzacja jest cechowana przez categorization characteristic

## coverage

Definicja: charakterystyka kategoryzacji, która określa, czy byty klasyfikowane przez generalizację posiadaną przez daną kategoryzację *muszą być klasyfikowane przez przynajmniej jedną specjalizację posiadaną przez daną kategoryzację*

Pojęcie ogólne: charakterystyka kategoryzacji

Synonim: pokrycie

## disjointness

Definicja: charakterystyka kategoryzacji, która określa, czy byty klasyfikowane przez generalizację posiadaną przez daną kategoryzację *mogą być klasyfikowane przez więcej niż jedną specjalizację posiadaną przez daną kategoryzację*

Pojęcie ogólne: charakterystyka kategoryzacji

Synonim: rozłączność



## separation of inheritance and polymorphism

Definicja: charakterystyka kategoryzacji, która określa, czy dana kategoryzacja nie musi zawierać dziedziczenia oraz polimorfizmu

Uwaga: Rozdział dziedziczenia i polimorfizmu w ramach związku kategoryzacji pozwala tworzyć związki kategoryzacji, w ramach których istnieje możliwość selektywnego włączenia/wyłączenia dziedziczenia lub polimorfizmu.

Synonim: rozdział dziedziczenia i polimorfizmu

Pojęcie ogólne: charakterystyka kategoryzacji

## categorization aspect

Definicja: charakterystyka kategoryzacji, która daje możliwość grupowania kategoryzacji w celu nadania tej grupie dodatkowej semantyki

Synonim: aspekt kategoryzacji

## virtuality mode

Definicja: charakterystyka kategoryzacji, która określa sposób dziedziczenia cech, które są dziedziczone więcej niż jedną ścieżką kategoryzacji

Uwaga: Istnienie trybu wirtualności daje możliwość odwoływania się do wielokrotnie dziedziczonych cech z tego samego źródła, ale różnymi ścieżkami. Sposób ten może pozwalać na zwielokrotnienie tej cechy, albo jej ujednoznacznienie – wtedy rozważa się tzw. cechy wirtualne.

Pojęcie ogólne: charakterystyka kategoryzacji

Synonim: tryb wirtualności

### **3.3.7. Pojęcia związane z modelowaniem powiązań**

Na rysunku 3.8 przedstawiono fragment CLoM odnoszący się do modelowania nazwanych powiązań. W rozdziale przedstawiono definicje pojęć odnoszących się do powiązań, w szczególności dotyczących asocjacji, ich ról oraz ograniczeń związanych z rolami.

#### relationship

Definicja: instancja, która może przechowywać instancje roli oraz może być klasyfikowana przez asocjację

Pojęcie ogólne: instancja

Synonim: związek

#### association

Definicja: klasyfikator, który może posiadać rolę oraz klasyfikuje związek

Synonim: asocjacja

Pojęcie ogólne: klasyfikator

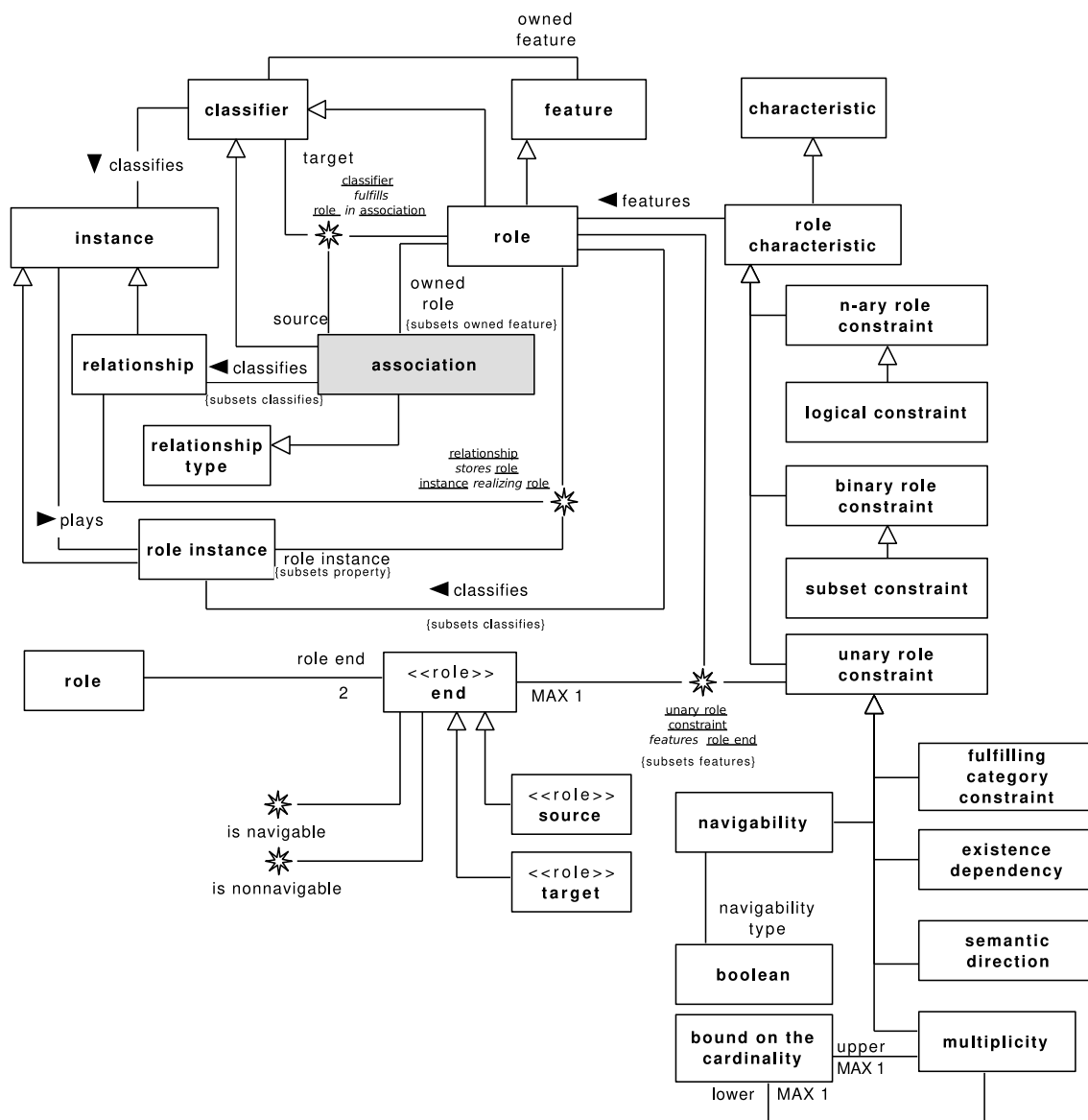
#### association arity

Definicja: liczba ról posiadanych przez daną asocjację

Synonim: arność asocjacji

#### association has association arity

Forma synonimiczna: asocjacja posiada arność asocjacji



Rysunek 3.8. Diagram pojęć przedstawiający powiązania między pojęciami modelowania związanych z modelowaniem asocjacji

### association is binary

Definicja: asocjacja posiada arność asocjacji równą 2

Forma synonimiczna: asocjacja jest binarna

### role instance

Definicja: właściwość, która może być przechowywana przez związek oraz realizuje rolę

Typ pojęcia: role

Pojęcie ogólne: właściwość

Synonim: instancja roli

### instance plays role instance

Forma synonimiczna: instancja bierze udział w instancji roli

Konieczność: *jest konieczne, aby instancja biorąca udział w danej instancji roli była klasyfikowana przez klasyfikator pełniący rolę, która jest realizowana przez daną instancję roli przechowywaną przez związek klasyfikowany przez asocjację posiadającą daną rolę*

### **participation of instance in role instance**

Definicja: *faktyczna sytuacja taka, że instancja bierze udział w instancji roli*

Synonim: uczestnictwo instancji w instancji roli

### **role**

Definicja: *cecha, która może być posiadana przez asocjację oraz może być realizowana przez instancję roli*

Pojęcie ogólne: cecha

Synonim: rola

### **end**

Definicja: *pojęcie, które pełnią pojęcia modelowania powiązane z rolą.*

Synonim: końcówka

Typ pojęcia: sbvr:role

### **role has end**

Synonim: końcówka

Konieczność: *każda rola posiada dokładnie 2 końcówki*

### **role end**

Typ pojęcia: końcówka roli

Definicja: *końcówka posiadana przez rolę.*

Synonim: końcówka roli

Typ pojęcia: sbvr:role

### **target**

Definicja: *końcówka roli, do której przypisano klasyfikator pełniący rolę w asocjacji*

Pojęcie ogólne: końcówka roli

Synonim: cel

### **source**

Definicja: *końcówka roli, do której przypisano asocjację posiadającą rolę*

Pojęcie ogólne: końcówka roli

Synonim: źródło

### **classifier fulfills role in association**

Forma synonimiczna: klasyfikator pełni rolę roli w asocjacji

### **role fulfillment**

Definicja: *faktyczna sytuacja taka, że klasyfikator pełni rolę w asocjacji*

Synonim: pełnienie roli

### **owned role**

Definicja: *rola, która jest częścią asocjacji*

Synonim: inherentna rola

Typ pojęcia: role

### role characteristic

Definicja: n-argumentowe ograniczenie roli lub dwuargumentowe ograniczenie roli lub jednoargumentowe ograniczenie roli

Synonim: charakterystyka roli

Pojęcie ogólne: charakterystyka

### unary role constraint

Definicja: charakterystyka roli, która *cechuje* dokładnie jedną rolę

Synonim: jednoargumentowe ograniczenie roli

Pojęcie ogólne: charakterystyka roli

### unary role constraint features role end for role

Forma synonimiczna: jednoargumentowe ograniczenie roli *cechuje* końcówkę roli dla roli

Forma synonimiczna: końcówka roli jest cechowana przez jednoargumentowe ograniczenie roli

Konieczność: *każde* jednoargumentowe ograniczenie roli *cechuje* *co najwyżej jedną* końcówkę roli dla dokładnie jednej roli

### binary role constraint

Definicja: charakterystyka roli, która *cechuje* dokładnie 2 role

Synonim: dwuargumentowe ograniczenie roli

Pojęcie ogólne: charakterystyka roli

### n-ary role constraint

Definicja: charakterystyka roli, która *cechuje* *przynajmniej* 2 role

Synonim: n-argumentowe ograniczenie roli

Pojęcie ogólne: charakterystyka roli

### logical role constraint

Definicja: n-argumentowe ograniczenie roli, które wprowadza reguły logiczne, w ramach których operandami są fakty determinujące uczestnictwo instancji w instancji roli

Synonim: ograniczenie logiczne ról

Uwaga: Poprzez termin *reguły logiczne* rozumiane są wyrażenia składające się z  $n$ -arnych operatorów logicznych, takich jak np. koniunkcja, alternatywa, alternatywa wykluczająca.

Pojęcie ogólne: n-argumentowe ograniczenie roli

### subset constraint

Definicja: dwuargumentowe ograniczenie roli, które wymaga aby *każda* instancja biorąca udział w instancji roli realizującej daną rolę  $r_1$  *musiała brać udział w* instancji roli realizującej rolę  $r_2$

Synonim: ograniczenie podzbioru

### fulfilling category constraint

Definicja: jednoargumentowe ograniczenie roli, które określa zbiór klasyfikatorów posiadających możliwość pełnienia roli

Synonim: ograniczanie pełniącej kategorii

Pojęcie ogólne: jednoargumentowe ograniczenie roli

## existence dependency

Definicja: jednoargumentowe ograniczenie roli, które wymusza zakończenie czasu życia instancji zależnej kiedy nastąpi zakończenie czasu życia instancji niezależnej

Synonim: zależność istnienia

Synonim: zależność czasu życia

Uwaga: Zależność instancji oznacza skierowanie ograniczenia w ramach roli, którą ogranicza. Elementem zależnym może być zarówno instancja odgrywająca instancję roli, jak również związek, który tę instancję przechowuje. Analogicznie w przypadku elementu niezależnego.

Pojęcie ogólne: jednoargumentowe ograniczenie roli

## navigability

Definicja: jednoargumentowe ograniczenie roli, które zakłada posiadanie referencji przez końcówkę roli

Synonim: nawigowalność

Uwaga: Nawigowalność oznacza określenie możliwości kierunku przejścia pomiędzy końcówkami roli. Możliwe jest definiowanie ról jednokierunkowych o nawigowalności określonej jako źródło → cel, cel → źródło oraz dwukierunkowych.

## navigability has navigability type

Forma synonimiczna: nawigowalność posiada typ nawigowalności

## navigability type

Definicja: wartość logiczna, która określa, czy instancje klasyfikatora pełniącego rolę końcówki roli cechowanej przez nawigowalność posiadają referencje do instancji klasyfikowanych przez klasyfikator pełniący rolę drugiej końcówki roli

Synonim: typ nawigowalności

## role end is navigable

Definicja: końcówka roli jest cechowana przez nawigowalność, która posiada typ nawigowalności, który jest równy prawda

Forma synonimiczna: końcówka roli jest nawigowalna

Konieczność: jeżeli końcówka roli jest nienawigowalna, to ta końcówka roli nie jest nawigowalna

## role end is nonnavigable

Definicja: końcówka roli jest cechowana przez nawigowalność, która posiada typ nawigowalności, który jest równy fałsz

Forma synonimiczna: końcówka roli jest nienawigowalna

Konieczność: jeżeli końcówka roli jest nawigowalna, to ta końcówka roli nie jest nienawigowalna

## semantic direction

Definicja: jednoargumentowe ograniczenie roli, które określa kierunek czytania roli

Synonim: semantyczna kierunkowość

Uwaga: Semantyczna kierunkowość dotyczy mechanizmów modelowania określających odwzorowanie treści łączonych elementów w ramach modelu w konstrukcje lingwistyczne typu SPO (*Subject–Predicate–Object*). Możliwość określenia takiego kierunku ułatwia osobom czytającym model zrozumienie charakteru powiązania.

## multiplicity

Definicja: jednoargumentowe ograniczenie roli, które określa kres górny lub kres dolny uczestnictwa instancji w instancji roli

Synonim: krotność

## upper bound

Pojęcie ogólne: collection upper bound

Definicja: kres liczebności, który określa maksymalną liczbę uczestnictwa instancji w instancji roli

Synonim: kres górny

## lower bound

Pojęcie ogólne: collection lower bound

Definicja: kres liczebności, który określa minimalną liczbę uczestnictwa instancji w instancji roli

Synonim: kres dolny

## multiplicity can have upper bound

Forma synonimiczna: krotność może posiadać kres górny

Konieczność: krotność może posiadać co najwyżej jeden kres górny

## multiplicity can have lower bound

Forma synonimiczna: krotność może posiadać kres dolny

Konieczność: krotność może posiadać co najwyżej jeden kres dolny

## bound on the cardinality

Definicja: liczba naturalna, która jest wartością graniczną krotności

Synonim: kres liczebności

Uwaga: Przedział wyznaczony przez wartości graniczne obejmuje je.

Typ pojęcia: role

## **3.4. Pojęcia związane z modelowaniem wzorców**

Niniejszy podrozdział zawiera definicje pojęć dedykowane opisowi wzorców modelowania rozważanych w ramach niniejszej pracy. Diagram na rys. 3.9 obrazuje najistotniejsze pojęcia w ramach tej gałęzi systemu pojęć.

### pattern concept

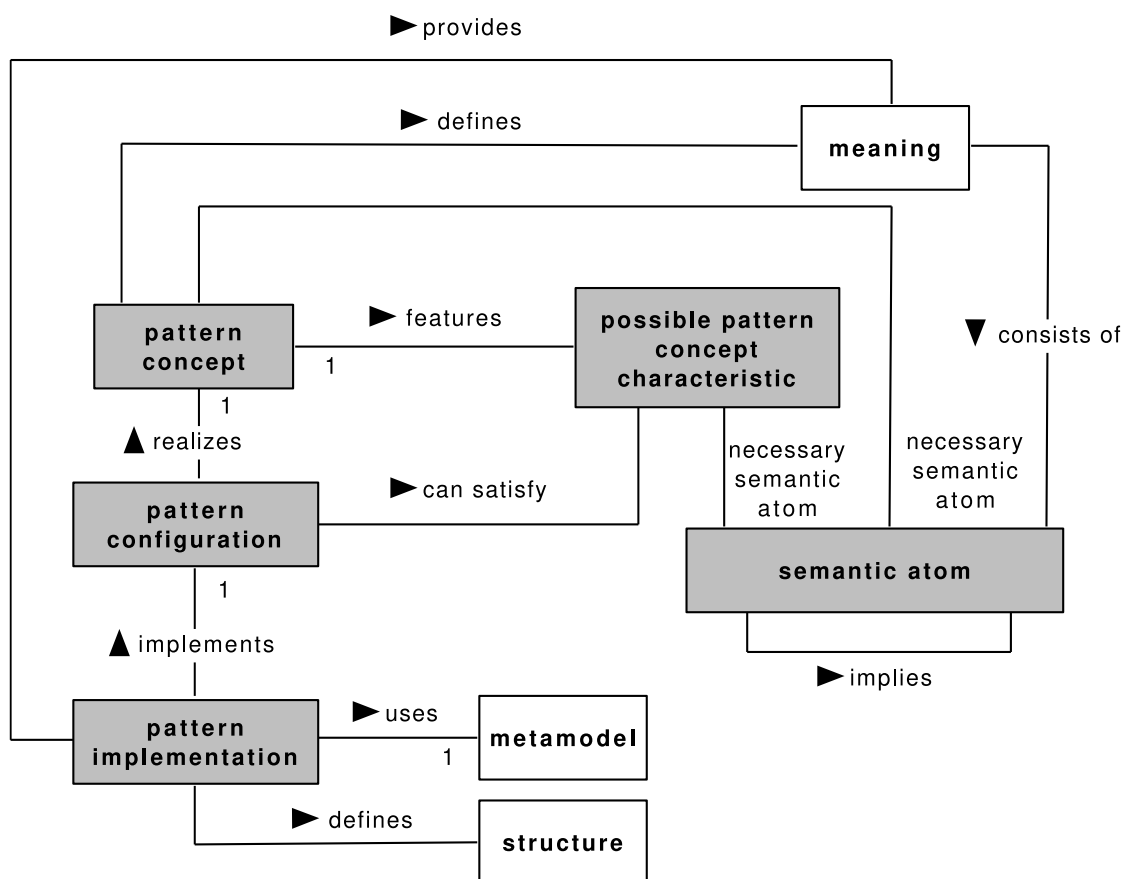
Definicja: zbiór pojęć definiujący znaczenie wzorca w zakresie koniecznych atomów semantycznych oraz własności możliwych

Synonim: koncepcja wzorca

Uwaga: Koncepcja wzorca stanowi jego ramy definicyjne, tj. określa pełną przestrzeń semantyczną służącą do możliwości określania, jakie struktury mogą spełniać wymagania określone w ramach danego wzorca. Jest to najwyższy poziom abstrakcji danego pojęcia w zakresie rozumowania na temat wzorców.

### pattern concept defines meaning

Forma synonimiczna: koncepcja wzorca definiuje znaczenie



Rysunek 3.9. Pojęcia związane z modelowaniem wzorców

### pattern configuration

Definicja: konfiguracja koniecznych atomów semantycznych i podzbiór zbioru własności możliwych charakteryzująca konkretne zestawienie właściwości semantycznych badanych w zakresie klasyfikacji struktury jako instancji danego wzorca

Synonim: konfiguracja wzorca

### pattern configuration realizes pattern concept

Definicja: konfiguracja wzorca określa podzbiór własności możliwych cechujących koncepcję wzorca

Forma synonimiczna: konfiguracja wzorca realizuje koncepcję wzorca

Konieczność: każda konfiguracja wzorca realizuje dokładnie jedną koncepcję wzorca

Uwaga: Koncepcja wzorca może przyjmować różne postaci, dzięki wzbogacaniu jej przy pomocy własności możliwych. Dzięki konfiguracji wzorca, istnieje możliwość realizacji różnych wersji/konfiguracji ogólnej koncepcji wzorca.

### pattern implementation

Definicja: odwzorowanie konfiguracji wzorca na postać struktury zgodnej z gramatyką i semantyką danego metamodelu

Synonim: implementacja wzorca

### pattern implementation implements pattern configuration

Forma synonimiczna: implementacja wzorca implementuje konfigurację wzorca

Konieczność: **każda** implementacja wzorca **implementuje** **dokładnie jedną** konfigurację wzorca

### pattern implementation uses metamodel

Forma synonimiczna: implementacja wzorca **używa** metamodel

Konieczność: **każda** implementacja wzorca **używa** **dokładnie jeden** metamodel

### pattern implementation defines structure

Forma synonimiczna: implementacja wzorca **definiuje** strukturę

### possible pattern concept characteristic

Definicja: niezależna, opcjonalna cecha koncepcji wzorca **realizowana przez** konfigurację wzorca

Synonim: własność koncepcji wzorca

Synonim: własność możliwa

### pattern concept features possible pattern concept characteristic

Forma synonimiczna: koncepcja wzorca **jest cechowana przez** własność możliwą

Konieczność: **Każda** własność możliwa **cechuje** **dokładnie jedną** koncepcję wzorca

### pattern configuration can satisfy possible pattern concept characteristic

Forma synonimiczna: konfiguracja wzorca **może spełniać** własność możliwą

Forma synonimiczna: własność możliwa **cechuje** konfigurację wzorca

Konieczność: własność możliwa **nie może cechować** konfiguracji wzorca, **która nie realizuje** koncepcji wzorca **cechowanej przez tę** własność możliwą

### semantic atom

Definicja: niepodzielny fakt **lub** reguła opisująca semantykę

Synonim: atom semantyczny

### semantic atom<sub>1</sub> implies semantic atom<sub>2</sub>

Forma synonimiczna: atom semantyczny<sub>1</sub> **implikuje** atom semantyczny<sub>2</sub>

### semantic atom is part of meaning

Forma synonimiczna: meaning **consists of** semantic atom

Forma synonimiczna: atom semantyczny **jest częścią znaczenia**

Forma synonimiczna: znaczenie **posiada** atom semantyczny

### necessary semantic atom

Definicja: atom semantyczny, **który jest** niezbędny do określenia istoty znaczenia koncepcji wzorca **lub** własności możliwej

Typ pojęcia: role

Synonim: konieczny atom semantyczny

### pattern concept has necessary semantic atom

Forma synonimiczna: koncepcja wzorca **posiada** konieczny atom semantyczny

### possible pattern concept characteristic has necessary semantic atom

Forma synonimiczna: własność możliwa **posiada** konieczny atom semantyczny



### 3.5. Podsumowanie

W niniejszym rozdziale przedstawiono system pojęciowy CLoM, dostarczający terminów służących do specyfikacji semantyki dotyczącej dziedziny modelowania. System ten stanowi punkt widzenia na świat modelowania i metamodelowania, starając się zarazem uporządkować najistotniejsze pojęcia stanowiące budulec opisu modeli obejmując nadzbiór tych pojęć oraz technik występujących w różnych metamodelach danych. CLoM cechowany jest założeniem świata otwartego<sup>5</sup>, tzn. ograniczenia i potencjalne związki pojęciowe, które nie są jawnie określone w systemie, należy rozumieć jako nieznanne. Nie należy zatem pojmować go jako *metamodel* sam w sobie. Co więcej, system ma charakter rozszerzalny i pozwala na definiowanie nowych terminów definiujących charakterystyki oraz ograniczenia pojęć modelowania.

---

<sup>5</sup> ang. *open-world assumption* (OWA)

## 4. Wzorce w modelowaniu asocjacyjnym

Wzorce projektowe stanowią ogólne i odtwarzalne rozwiązania często powtarzających się problemów [44, 152]. Ich wykorzystanie pozwala sprowadzić pracę związaną z projektowaniem do wyboru odpowiedniego, abstrakcyjnego wzorca i umiejętnego dostosowania go do projektowanego elementu systemu. Podejście bazujące na wzorcach znajduje swoje zastosowanie zarówno w wielu dziedzinach inżynierii związanej z programowaniem oraz projektowaniem, m.in. projektowaniu oprogramowania [21, 48], w projektowaniu ontologii [45, 68], w modelowaniu danych [11]. Wzorce są również ugruntowane w zagadnieniach modelowania konceptualnego [4].

Z pojęciem wzorca wiąże się również pojęcie *antywzorca*, to znaczy zidentyfikowanych, najczęściej popełnianych błędów. Celem antywzorców jest dostarczenie narzędzi do identyfikacji takich struktur, a następnie zaleceń w zakresie działań naprawczych mających na celu zniwelowanie skutków ich zastosowania. Dodatkowo, w modelowaniu opartym o wzorce rozważa się również pojęcie *archetypu*. Archetypem nazywa się *głęboko abstrakcyjną strukturę wykraczającą poza jednostkowe zastosowanie* [11]. Archetypy pozwalają wydobyć pewne podstruktury współdzielone pomiędzy wieloma dziedzinami, których rozważenie może wspomóc podejście do efektywnego modelowania. Przykładem archetypów może być określenie szablonu struktury przechowującej adresy, płatności, informacje o transakcjach.

W niniejszym opracowaniu potraktowano wzorce jako abstrakcyjny względem dziedziny, konceptualny budulec modeli rzeczywistości o charakterze powtarzalnym. Co więcej, uniezależniono istnienie wzorców od konkretnych metamodeli, w ramach których zostaną one zaimplementowane, a ich definicja przebiegać będzie poprzez trzy płaszczyzny:

1. poziom koncepcyjny, nieformalny, określający koncepcję wzorca
2. poziom specyfikacyjny, semi-formalny, określający koncepcję wzorca przy użyciu CLoM,
3. poziom implementacyjny, formalny, pokazujący w jaki sposób dana koncepcja zostaje zrealizowana w ramach konkretnego metamodelu.

W ramach poziomu koncepcyjnego przedstawione zostały zamysły koncepcji wzorców w języku naturalnym oraz opisy problemów, które te wzorce rozwiązują.

Dodatkowo, wyszczególniono uczestników wzorców wraz z opisem powiązań pomiędzy uczestnikami.

Poziom specyfikacyjny opiera się na systemie pojęciowym zdefiniowanym w rozdziale 3. Bazując na przyjętej konceptualizacji świata modelowania, zdefiniowane zostały wyrażenia SBVR opisujące semantykę koncepcji wzorców. W celu uproszczenia zapisu, przyjęto niejawne pojęcia jednostkowe oznaczające koncepcje konkretnych wzorców, np. lista, słownik, oraz ich własności, np. unikalność listy. Wyrażenia mają postaci faktów i reguł, które specyfikują kolejne wymagania, które muszą zostać zrealizowane, aby stwierdzić realizację koncepcji wzorca. Wyrażenia mają charakter atomowy, tj. wyliczane są w formie niepodzielnej. Wyrażenia zostały podzielone na dwa typy:

1. **konieczne** atomy semantyczne – wyrażenia, które stanowią konieczności bezwzględnie wymagane do stwierdzenia realizacji koncepcji wzorca,
2. własności **możliwe** – składające się z atomów semantycznych – wyrażenia, których koniunkcja jest wymagana do tego, aby stwierdzić, że wzorzec jest charakteryzowany przez daną własność.

Poziom implementacyjny opisuje modele wzorów poprzez przedstawienie konceptualnych modeli asocjacyjnych, które je realizują wraz z opisem implementacji poszczególnych atomów semantycznych, zarówno koniecznych jak i tych opisujących własności możliwe. W ramach opisów użyto terminu implementacja w jego szerokim znaczeniu, w odniesieniu do modeli konceptualnych w celu położenia nacisku na fakt, iż stanowią one realne i wymierne artefakty powstające w ramach przyjętego podejścia do modelowania danych. W rozdziale przedstawiono implementacje każdego z wzorców w postaci diagramu AML prezentującego część intensjonalną modelu oraz zbioru równoważnych wyrażen AFN.

W ramach rozdziału wyróżniono dwa typy wzorców: wzorce abstrakcyjne oraz dedykowane AOM. Wzorce abstrakcyjne odwołują się do typowych problemów modelowania danych. Wzorce dedykowane AOM mają sens istnienia jedynie w ramach metamodelu asocjacyjnego i powinny być przez jego pryzmat rozważane. Niezależnie od rodzaju, specyfikacja wzorca została określona w sposób pojęciowy, abstrahujący od samej implementacji.

#### 4.1. Wzorce abstrakcyjne

W niniejszym podrozdziale opisane zostały abstrakcyjne wzorce modelowania.

#### 4.1.1. Lista

##### Opis problemu

Rdzeniem problemu rozwiązywanego przez koncepcję wzorca listy jest zagadnienie *grupowania elementów*. Koncepcja wzorca ma na celu uchwycenie elementu grupującego jako osobnego, niezależnego pojęcia, dla którego samo grupowanie jest przyczyną wyodrębnienia.

##### Elementy składowe

Pojęcia stanowiące uczestników wzorca:

1. GRUPA – pojęcie pełniące rolę abstrakcji elementu grupującego.
2. ELEMENT – pojęcie pełniące rolę abstrakcji elementu grupowanego.

Pojęcie stanowiące powiązanie pomiędzy uczestnikami:

1. GRUPOWANIE – pojęcie pełniące rolę związku zachodzącego pomiędzy GRUPĄ a ELEMENTEM.

##### Własności

Pojęcia stanowiące własności wzorca *listy*:

1. UNIKALNOŚĆ – własność wzorca występująca, gdy w ramach grupowania element grupowany występuje nie więcej, niż jednokrotnie.
2. PRZYPIŚANIE  $n$ -OGRANICZONE – własność wzorca występująca, gdy każdy element łączony występuje w związku grupowania co najmniej  $n$  razy.
3. ZAPEŁNIENIE – własność wzorca występująca, gdy element grupujący musi grupować co najmniej jeden element grupowany.
4. PRZYNALEŻNOŚĆ  $n$ -OGRANICZONA – własność wzorca występująca, gdy element grupowany może występować co najwyżej w  $n$  grupowaniach.
5. BINAWIGOWALNOŚĆ – własność wzorca określająca, czy element grupowany posiada informację o istniejącym połączeniu pomiędzy nim, a listą. Innymi słowy, własność ta definiuje listę jako nawigowalną, gdy element grupowany posiada bezpośrednią referencję<sup>1</sup> do listy, która go grupuje.
6. ZALEŻNOŚĆ ISTNIENIA ELEMENTU OD GRUPY – własność wzorca określająca zależność istnienia elementu grupowanego od elementu grupującego, w ujęciu behawioralnym: w przypadku usuwania lub przerwania istnienia elementu grupującego, następuje zakończenie czasu życia elementu grupowanego.

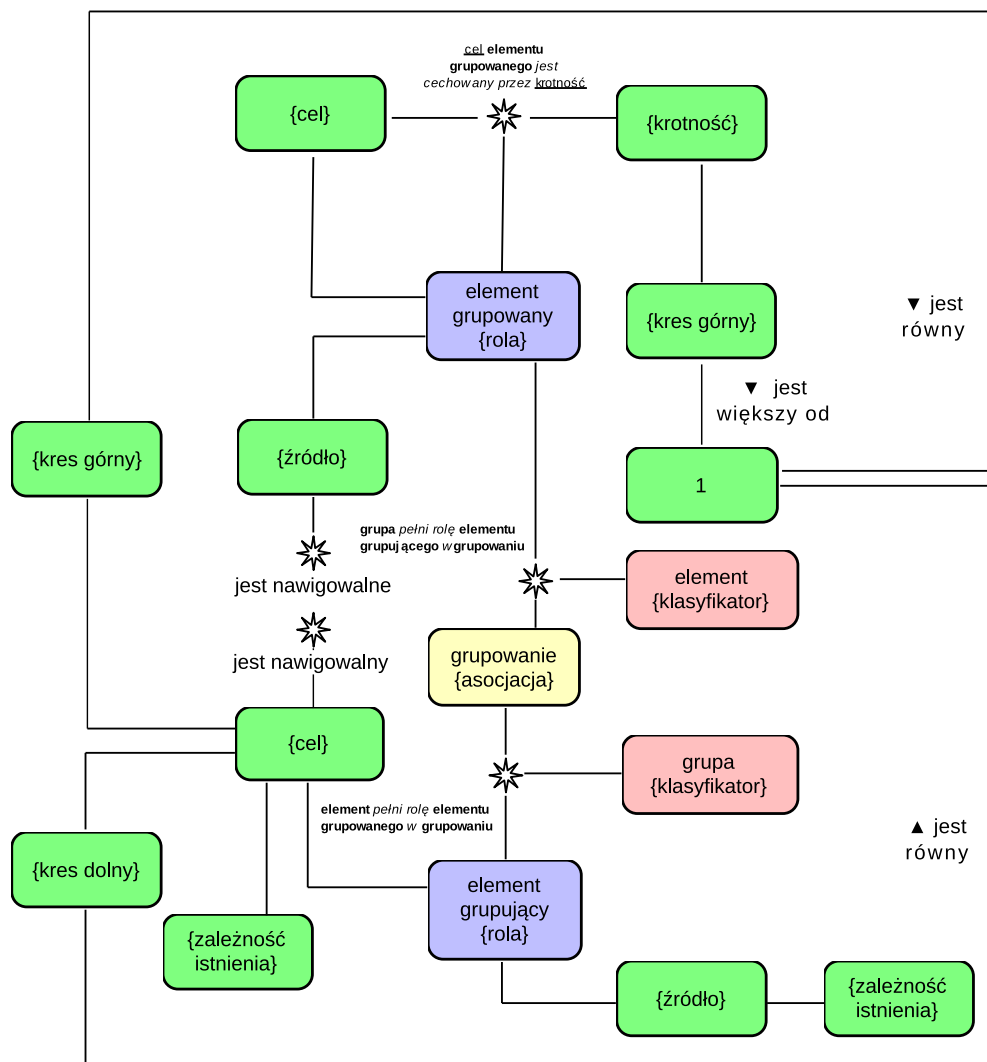
---

<sup>1</sup> *Bezpośrednia referencja* rozumiana jest jako możliwość odwołania się do danego elementu bez konieczności wykonywania jakichkolwiek operacji przeszukiwania.

7. ZALEŻNOŚĆ ISTNIENIA GRUPY OD ELEMENTU – własność wzorca określająca zależność istnienia elementu grupującego od elementu grupowanego, w ujęciu behawioralnym: w przypadku usuwania lub przerywania istnienia elementu grupowanego, następuje zakończenie czasu życia elementu grupującego.
8. UPORZĄDKOWANIE – własność wzorca określająca, czy kolejność elementów jest istotna.

### Specyfikacja wzorca

Diagram pojęć na rys. 4.1 przedstawia konieczne atomy semantyczne wzorca lista.



Rysunek 4.1. Diagram pojęć prezentujący konieczne atomy semantyczne koncepcji wzorca lista

*Konieczne atomy semantyczne*

#### K.1 lista jest konkretyzacją koncepcji wzorca

- K.2 lista zawsze posiada grupę, która jest konkretyzacją klasyfikatora
- K.3 lista zawsze posiada element, który jest konkretyzacją klasyfikatora
- K.4 grupowanie jest konkretyzacją asocjacji
- K.4.1 grupa zawsze pełni rolę elementu grupującego w grupowaniu
- K.4.1.1 cel elementu grupującego jest nawigowalny
- K.4.1.2 źródło elementu grupowanego jest nawigowalne
- K.4.1.3 cel elementu grupującego zawsze jest cechowany przez krotność, która posiada kres górny oraz kres dolny, który jest równy 1
- K.4.1.4 źródło elementu grupującego zawsze jest cechowane przez krotność, która posiada kres górny oraz kres dolny, który jest równy 1
- K.4.1.5 cel elementu grupującego zawsze jest cechowany przez zależność istnienia
- K.4.1.6 źródło elementu grupującego zawsze jest cechowane przez zależność istnienia
- K.4.2 element może pełnić rolę elementu grupowanego w grupowaniu
- K.4.2.1 cel elementu grupowanego zawsze jest cechowany przez krotność, która posiada kres górny, który jest większy od 1

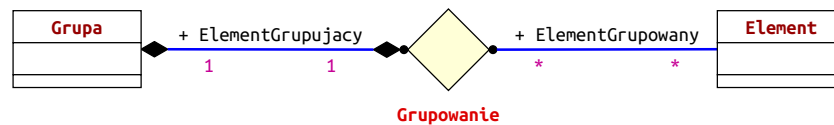
*Atomy semantyczne własności możliwych*

- M.1 unikalność listy jest konkretyzacją własności wzorca
- M.1.1 cel elementu grupowanego jest cechowany przez unikalność
- M.2 n- ograniczone przypisanie do listy jest konkretyzacją własności wzorca
- M.2.1 źródło elementu grupowanego jest cechowane przez krotność, która posiada kres dolny, który jest większy lub równy n
- M.3 zapełnienie listy jest konkretyzacją własności wzorca
- M.3.1 cel elementu grupowanego jest cechowany przez krotność, która posiada kres dolny, który jest większy lub równy 1
- M.4 binawigowalność listy jest konkretyzacją własności wzorca
- M.4.1 cel elementu grupowanego jest nawigowalny

- M.4.2 źródło elementu grupującego jest nawigowalne
- M.5 zależność istnienia elementu od grupy jest konkretyzacją własności wzorca
- M.5.1 cel elementu grupowanego jest cechowany przez zależność istnienia
- M.6 zależność istnienia grupy od elementu jest konkretyzacją własności wzorca
- M.6.1 źródło elementu grupowanego jest cechowane przez zależność istnienia
- M.7 uporządkowanie listy jest konkretyzacją własności wzorca
- M.7.1 element grupowany jest cechowany przez kolejność
- M.8 n-ograniczona przynależność do listy jest konkretyzacją własności wzorca
- M.8.1 źródło elementu grupowanego jest cechowane przez krotkość, która posiada kres dolny równy n

### Implementacja rozwiązania

Implementacja koniecznych atomów semantycznych wzorca lista



Rysunek 4.2. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne

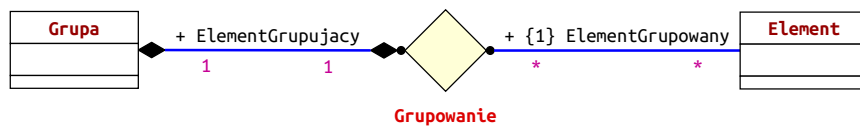
$$\diamond \text{Grupowanie} \left\{ \begin{array}{l} [*] \xrightarrow{+ElementGrupowany(\square Element)} [*] \square Element \\ [1] \xrightarrow{+ElementGrupujacy(\square Grupa)} [1] \square Grupa \end{array} \right\}; \quad (4.1)$$

Rysunek 4.2 przedstawia implementację koniecznych atomów semantycznych wzorca lista. Poniżej przedstawiono odniesienie kolejnych elementów modelu do realizacji koniecznych atomów semantycznych wzorca:

1. Utworzenie modelu asocjacyjnego realizuje konieczny atom semantyczny K.1.
2. Implementacja K.2 została zrealizowana poprzez utworzenie kolekcji  $\square Grupa$ .
3. Implementacja K.3 została zrealizowana poprzez utworzenie kolekcji  $\square Element$ .
4. Implementacja K.4 została zrealizowana poprzez utworzenie asocjacji  $\diamond Grupowanie$ .
5. Implementacja K.4.1 została zrealizowana poprzez utworzenie roli o nazwie *ElementGrupujacy* w asocjacji  $\diamond Grupowanie$ .

6. Implementacja K.4.1.1 - K.4.1.6 została zrealizowana poprzez implementację asocjacyjnego wzorca BACT opisanego w rozdziale 4.2.1.
7. Implementacja K.4.2 została zrealizowana poprzez utworzenie roli o nazwie *ElementGrupowany* w asocjacji  $\diamond$ *Grupowanie*.
8. Implementacja K.4.2.1 została zrealizowana poprzez nadanie tej roli krotności \* po stronie elementu wiążanego.

Implementacja koniecznych atomów semantycznych wzorca lista z własnością możliwą: unikalność listy

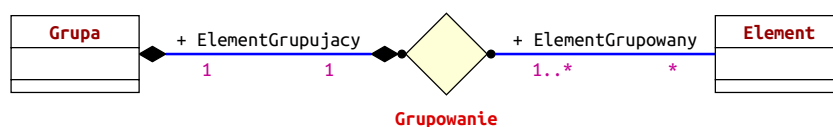


Rysunek 4.3. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne oraz własność możliwą: unikalność listy

$$\diamond\text{Grupowanie} \left( \begin{array}{c} [*] \xrightarrow{\{1\} + \text{ElementGrupowany}(\square \text{Element})} [*] \square \text{Element} \\ [1] \xrightarrow{+ \text{ElementGrupujacy}(\square \text{Grupa})} [1] \square \text{Grupa} \end{array} \right); \quad (4.2)$$

Na rysunku 4.3 przedstawiono realizację wzorca listy wraz z własnością możliwą unikalność listy. Implementacja zakłada realizację wszystkich koniecznych atomów semantycznych wzorca, wraz z atomami M.1 oraz M.1.1 określonymi w ramach atomów semantycznych własności możliwych. Implementacja M.1 polega na utworzeniu nowego modelu listy, natomiast implementacja M.1.1 została zrealizowana poprzez zapewnienie unikalności w roli *ElementGrupowany* asocjacji  $\diamond$ *Grupowanie*.

Implementacja koniecznych atomów semantycznych wzorca lista z własnością możliwą: n-ograniczone przypisanie do listy



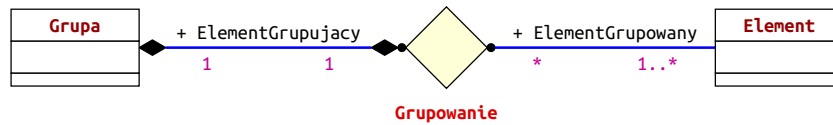
Rysunek 4.4. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne oraz własność możliwą: przypisanie do listy



$$\diamond Grupowanie \left( \begin{array}{l} [1..*] \xrightarrow{+ElementGrupowany(\square Element)} [*] \square Element \\ [1] \xrightarrow{+ElementGrupujacy(\square Grupa)} [1] \square Grupa \end{array} \right); \quad (4.3)$$

Rysunek 4.4 przedstawia realizację wzorca listy wraz z własnością możliwą [n-ograniczone przypisanie do listy](#) dla  $n = 1$ . Implementacja zakłada realizację wszystkich koniecznych atomów semantycznych wzorca, wraz z atomami semantycznymi M.2 oraz M.2.1. Implementacja M.2 polega na utworzeniu nowego modelu listy, natomiast implementacja M.2.1 została zrealizowana poprzez określenie krotności roli *ElementGrupowany* w asocjacji  $\diamond Grupowanie$  po stronie związku jako  $1..*$ .

Implementacja koniecznych atomów semantycznych wzorca lista z własnością możliwą: [zapełnienie listy](#)

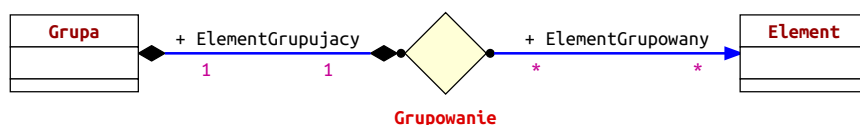


Rysunek 4.5. Diagram modelu asocjacyjnego wzorca listy implementującego konieczne atomy semantyczne oraz własność możliwą: zapełnienie listy

$$\diamond Grupowanie \left( \begin{array}{l} [*] \xrightarrow{+ElementGrupowany(\square Element)} [1..*] \square Element \\ [1] \xrightarrow{+ElementGrupujacy(\square Grupa)} [1] \square Grupa \end{array} \right); \quad (4.4)$$

Na rysunku 4.5 przedstawiono realizację wzorca listy wraz z własnością możliwą [zapełnienie listy](#). Implementacja zakłada realizację wszystkich koniecznych atomów semantycznych wzorca, wraz z atomami semantycznymi M.3 oraz M.3.1. Implementacja M.3 polega na utworzeniu nowego modelu listy, natomiast implementacja M.3.1 została zrealizowana poprzez określenie krotności roli *ElementGrupowany* w asocjacji  $\diamond Grupowanie$  po stronie elementu związanego jako  $1..*$ .

Implementacja koniecznych atomów semantycznych wzorca lista z własnością możliwą: [binawigowalność listy](#)

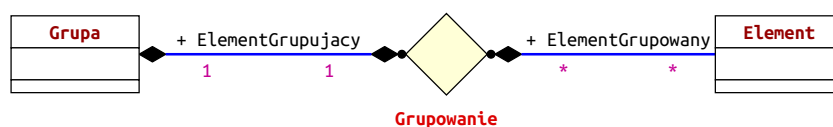


Rysunek 4.6. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne oraz negację własności możliwej: binawigowalność

$$\diamond Grupowanie \left\langle \begin{array}{l} [1] \xrightarrow{+ElementGrupujacy(\square Grupa)} [1] \square Grupa \\ [*] \xrightarrow{+ElementGrupowany(\square Element)} [*] \square Element \end{array} \right\rangle; \quad (4.5)$$

Implementacja wzorca lista przedstawiona na wyr. 4.1 spełnia semantykę własności możliwej binawigowalności listy. W przypadku implementacji w metamodelu asocjacyjnym, binawigowalność jest własnością domyślną, co powoduje, iż własność możliwa binawigowalność listy jest domyślnie spełniona przez modele asocjacyjne realizujące wzorzec listy. Model asocjacyjny dopuszcza jednak implementacje, w których występuje uninawigowalność. Przekładając to na przyjęty system pojęciowy, realizacja uninawigowalności stanowiłaby zanegowanie tej własności (rys. 4.6).

Implementacja koniecznych atomów semantycznych wzorca lista z własnością możliwą: zależność istnienia elementu od grupy



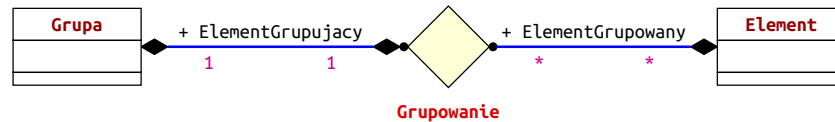
Rysunek 4.7. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne oraz własność możliwą: zależność istnienia elementu od grupy

$$\diamond Grupowanie \left\langle \begin{array}{l} [*] \xrightarrow{+ElementGrupowany(\square Element)} [*] \square Element \\ [1] \xrightarrow{+ElementGrupujacy(\square Grupa)} [1] \square Grupa \end{array} \right\rangle; \quad (4.6)$$

Na rysunku 4.7 przedstawiono realizację wzorca listy wraz z własnością możliwą zależność istnienia elementu od grupy. Implementacja zakłada realizację wszystkich koniecznych atomów semantycznych wzorca, wraz z atomami semantycznymi M.5 oraz M.5.1. Implementacja atomu semantycznego M.5 polega na utworzeniu nowego modelu listy, natomiast implementacja M.5.1 została zrealizowana poprzez

zapewnienie kompozycji po stronie związku w roli *ElementGrupowany* asocjacji  $\diamond$ *Grupowanie*.

*Implementacja koniecznych atomów semantycznych wzorca lista z własnością możliwą: zależność istnienia grupy od elementu*



Rysunek 4.8. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne oraz własność możliwą: zależność istnienia grupy od elementu

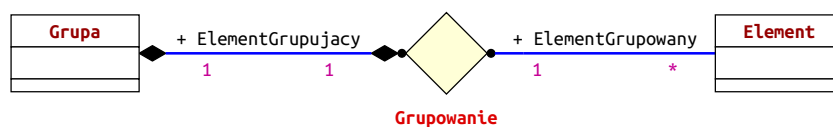
$$\diamond \text{Grupowanie} \left\{ \begin{array}{l} [*] \xrightarrow{+ElementGrupowany(\square Element)} [*] \square Element \\ [1] \xrightarrow{+ElementGrupujacy(\square Grupa)} [1] \square Grupa \end{array} \right\}; \quad (4.7)$$

Na rysunku 4.8 przedstawiono realizację wzorca listy wraz z własnością możliwą zależność istnienia grupy od elementu. Implementacja zakłada realizację wszystkich koniecznych atomów semantycznych wzorca, wraz z atomami semantycznymi M.6 oraz M.6.1. Implementacja atomu semantycznego M.6 polega na utworzeniu nowego modelu listy, natomiast implementacja atomu semantycznego M.6.1 została zrealizowana poprzez zapewnienie kompozycji po stronie elementu wiążanego w roli *ElementGrupowany* asocjacji  $\diamond$ *Grupowanie*.

*Implementacja koniecznych atomów semantycznych wzorca lista z własnością możliwą: uporządkowanie listy*

Implementacja wzorca lista przedstawiona na wyr. 4.1 spełnia semantykę własności możliwej uporządkowanie listy, ponieważ realizuje atomy semantyczne M.7 i M.7.1. W metamodelu asocjacyjnym wszystkie zbiorowości, takie jak kolekcje oraz role są uporządkowane. W efekcie, jedynie co jest możliwe to zignorowanie istniejącego porządku. Musi to odbywać się, jednakże, w innej warstwie – na poziomie semantyki modelu.

*Implementacja koniecznych atomów semantycznych wzorca lista z własnością możliwą: n-ograniczona przynależność*



Rysunek 4.9. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne oraz własność możliwą:  $n$ -ograniczona przynależność do listy dla  $n = 1$

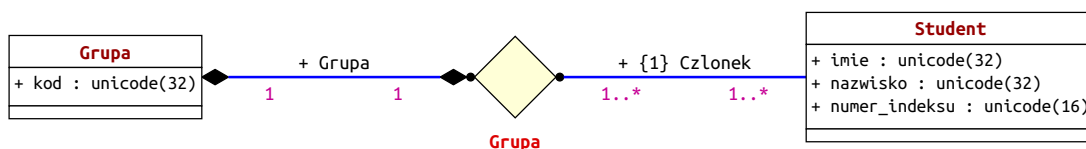
$$\diamond Grupowanie \left( \begin{array}{c} [1] \xrightarrow{+ElementGrupowany(\square Element)} [*] \square Element \\ [1] \xrightarrow{+ElementGrupujacy(\square Grupa)} [1] \square Grupa \end{array} \right); \quad (4.8)$$

Na rysunku 4.9 przedstawiono realizację wzorca listy wraz z własnością możliwą  $n$ -ograniczona przynależność do listy dla  $n = 1$ . Implementacja zakłada realizację wszystkich koniecznych atomów semantycznych wzorca, wraz z atomami semantycznymi M.8 oraz M.8.1. Implementacja atomu semantycznego M.8 polega na utworzeniu nowego modelu listy, natomiast implementacja atomu semantycznego M.8.1 została zrealizowana poprzez określenie krotności roli *ElementGrupowany* w asocjacji  $\diamond Grupowanie$  po stronie związku jako 1.

### Przykład zastosowania

Przykładem zastosowania listy może być uproszczony model grup zajęciowych studentów. Rolę uczestnika wzorca o nazwie GRUPA pełni tutaj reprezentacja grupy w postaci asocjacji  $\diamond Grupa$ , do której są przypisani studenci, natomiast rolę grupowanego elementu – reprezentacja studenta (kolekcja  $\square Student$ ). Na diagramie AML obrazującym przykład zastosowania wzorca (rys. 4.10) pokazano model, który posiada następujące własności możliwe:

- unikalność listy – pozwala reprezentować fakt związany z tym, iż student w określonej grupie może znajdować się tylko raz,
- binawigowalność listy – reprezentuje możliwość wzajemnej nawigacji zarówno od grupy do studenta, jak również w odwrotnym kierunku,
- zapełnienie listy – od grupy wymaga się posiadania przynajmniej jednego studenta,
- 1-ograniczone przypisanie do listy – każdy student musi przynależeć co najmniej do jednej grupy.



Rysunek 4.10. Przykład wykorzystania wzorca modelowania listy

#### 4.1.2. Słownik

##### Opis problemu

Problem rozwiązywany przez wzorzec słownika polega na możliwości wyboru dokładnie jednego bytu z pewnego ściśle określonego zbioru wyliczeniowego dopuszczalnych bytów. Ma to na celu opisanie pewnej cechy elementu opisywanego. Słownik ma w swoim założeniu posiadanie charakteru ponownego użycia, to znaczy z założenia powinien umożliwiać opisanie pewnej cechy należącej do różnych klasyfikatorów. Wzorzec powiązany jest z techniką *pivotingu* [137] w projektowaniu relacyjnych baz danych.

##### Elementy składowe

Pojęcia stanowiące uczestników wzorca:

1. WYLICZENIE BYTÓW – pojęcie pełniące rolę abstrakcji elementu opisującego inne elementy.
2. ELEMENT OPISYWANY – pojęcie pełniące rolę abstrakcji elementu opisywanego.

Pojęcie stanowiące powiązanie pomiędzy uczestnikami:

1. OPISYWANIE – pojęcie pełniące rolę związku zachodzącego pomiędzy WYLICZENIEM BYTÓW a ELEMENTEM OPISYWANYM

##### Własności

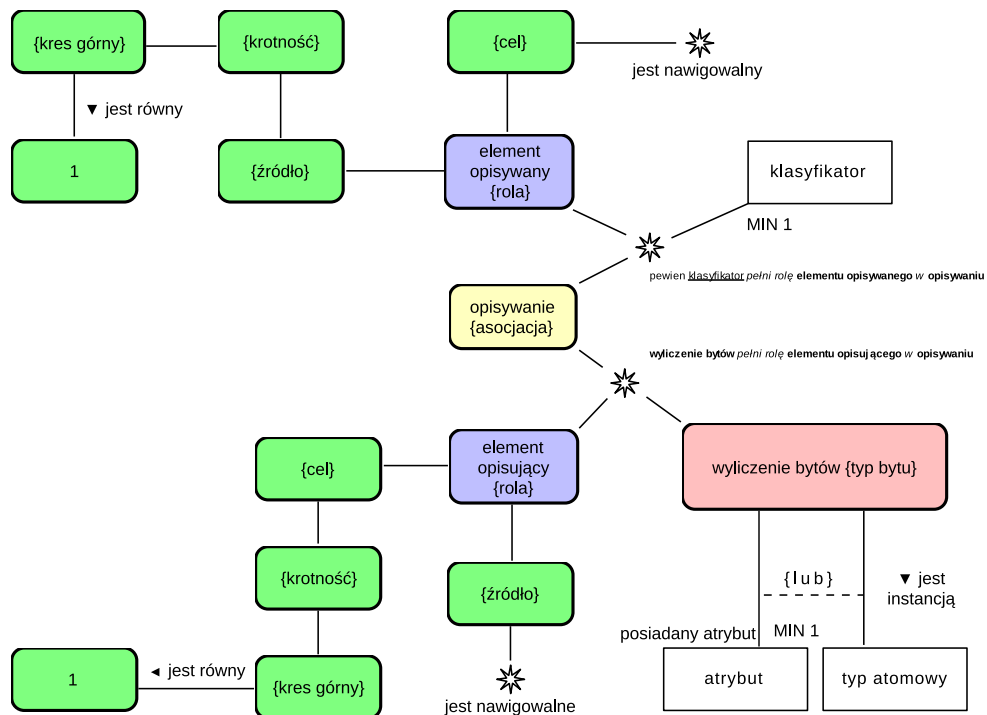
Pojęcia stanowiące własności wzorca *słownika*:

1. BINAWIGOWALNOŚĆ – własność wzorca określająca, czy WYLICZENIE BYTÓW posiada informacje o istniejącym połączeniu pomiędzy tym wyliczeniem, a elementem opisywanym.
2. *n*-OGRANICZENIE UŻYCIA – własność wzorca określająca, co najwyżej ile elementów WYLICZENIE BYTÓW może opisywać ELEMENT OPISYWANY.
3. *n*-MINIMALNA OPISYWALNOŚĆ – własność wzorca określająca, co najmniej ile elementów WYLICZENIE BYTÓW musi opisywać ELEMENT OPISYWANY.

4. OBLIGATORYJNOŚĆ OPISYWANIA – własność wzorca określająca, czy ELEMENT OPISYWANY musi być opisywany przez WYLICZENIE BYTÓW.
5. ZALEŻNOŚĆ ISTNIENIA ELEMENTU OPISYWANEGO OD WYLICZENIA BYTÓW – własność wzorca określająca, czy w ujęciu behawioralnym podczas przerwania istnienia WYLICZENIA BYTÓW, następuje zakończenie czasu życia ELEMENTU OPISYWANEGO.
6. UPORZĄDKOWANIE – własność wzorca określająca, czy kolejność WYLICZENIA BYTÓW jest istotna.

### Specyfikacja wzorca

Diagram pojęć na rys. 4.11 przedstawia konieczne atomy semantyczne wzorca słownik.



Rysunek 4.11. Diagram pojęć prezentujący konieczne atomy semantyczne koncepcji wzorca słownik

#### Konieczne atomy semantyczne

K.1 słownik jest konkretyzacją koncepcji wzorca

K.2 słownik zawsze posiada wyliczenie bytów, które jest konkretyzacją typu bytu

K.2.1 wyliczenie bytów musi posiadać co najmniej 1 atrybut lub być instancją typu atomowego

- K.3 słownik *zawsze posiada* opisywanie, *które jest konkretyzacją* asocjacji
- K.3.1 wyliczenie bytów *zawsze pełni rolę* elementu opisującego w opisywaniu
- K.3.1.1 cel elementu opisującego *zawsze jest cechowany przez* krotność, *która posiada* kres górny, *który jest równy* 1
- K.3.1.2 źródło elementu opisującego *zawsze jest nawigowalne*
- K.3.2 *pewien* klasyfikator *zawsze pełni rolę* elementu opisywanego w opisywaniu
- K.3.2.1 źródło elementu opisywanego *zawsze jest cechowany przez* krotność, *która posiada* kres górny, *który jest równy* 1
- K.3.2.2 cel elementu opisywanego *zawsze jest nawigowalny*

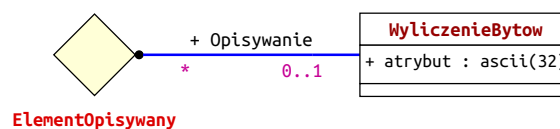
*Atomy semantyczne własności możliwych*

- M.1 binawigowalność słownika *jest konkretyzacją* własności wzorca
- M.1.1 cel elementu opisującego *zawsze jest nawigowalny*
- M.1.2 źródło elementu opisującego *zawsze jest nawigowalne*
- M.2 *n-*ograniczenie użycia słownika *jest konkretyzacją* własności wzorca
- M.2.1 źródło elementu opisującego *zawsze jest cechowany przez* krotność, *która posiada* kres górny, *który jest równy* *n*
- M.2.2 cel elementu opisywanego *zawsze jest cechowany przez* krotność, *która posiada* kres górny, *który jest równy* *n*
- M.3 *n-*minimalna opisywalność słownika *jest konkretyzacją* własności wzorca
- M.3.1 źródło elementu opisującego *zawsze jest cechowany przez* krotność, *która posiada* kres dolny, *który jest równy* *n*
- M.3.2 cel elementu opisywanego *zawsze jest cechowany przez* krotność, *która posiada* kres dolny, *który jest równy* *n*
- M.4 obligatoryjność opisywania *jest konkretyzacją* własności wzorca
- M.4.1 źródło elementu opisującego *zawsze jest cechowany przez* krotność, *która posiada* kres dolny, *który jest równy* 1

- M.4.2 cel elementu opisywanego zawsze jest cechowany przez krotność, która posiada kres górny, który jest równy 1
- M.5 zależność istnienia elementu opisywanego od opisującego jest konkretyzacją własności wzorca
- M.5.1 cel elementu opisującego jest cechowany przez zależność istnienia
- M.5.2 źródło elementu opisywanego jest cechowane przez zależność istnienia
- M.6 uporządkowanie słownika jest konkretyzacją własności wzorca
- M.6.1 cel elementu opisującego jest cechowany przez uporządkowanie
- M.6.2 źródło elementu opisywanego jest cechowane przez uporządkowanie

### Implementacja rozwiązania

Implementacja koniecznych atomów semantycznych wzorca słownik



Rysunek 4.12. Diagram modelu asocjacyjnego wzorca słownik implementującego konieczne atomy semantyczne

$$\diamond \text{ElementOpisywany} [*] \xrightarrow{+\text{Opisywanie}(\square \text{WyliczenieBytow})} [0..1] \square \text{WyliczenieBytow} \quad (4.9)$$

$$\square \text{WyliczenieBytow} \left\{ +\text{atrybut} : \text{ascii}(32) \right\}; \quad (4.10)$$

Rysunek 4.12 przedstawia implementację koniecznych atomów semantycznych wzorca słownika. Poniżej przedstawiono odniesienie kolejnych elementów modelu do realizacji koniecznych atomów semantycznych wzorca:

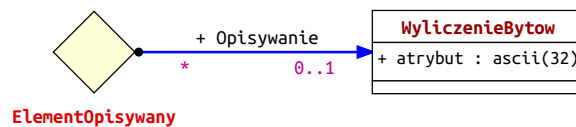
1. Utworzenie modelu asocjacyjnego realizuje konieczny atom semantyczny K.1.
2. Implementacja K.2 została zrealizowana poprzez utworzenie kolekcji  $\square \text{WyliczenieBytow}$ .
3. Implementacja K.2.1 została zrealizowana poprzez utworzenie atrybutu o nazwie *atrybut* w kolekcji  $\square \text{WyliczenieBytow}$ .



4. Implementacja K.3, K.3.1 oraz K.3.2 została zrealizowana poprzez zastosowanie redukcji asocjacji binarnych – opisywanie zostało zrealizowane jako rola *Opisywanie* w asocjacji  $\diamond$ *ElementOpisywany*.
5. Implementacja K.3.1.1 oraz K.3.2.1 została zrealizowana poprzez określenie krotności zakończenia roli *Opisywanie* jako 0..1.
6. Implementacja K.3.1.2 oraz K.3.2.2 realizowana jest ze względu na fakt domyślnej binawigowalności ról w ramach AOM.

Implementacja koniecznych atomów semantycznych wzorca słownik z własnością możliwą: binawigowalność słownika

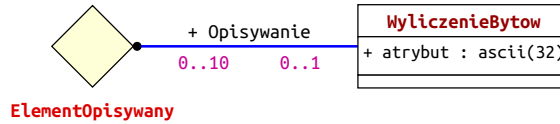
Implementacja wzorca słownika przedstawiona na wyr. 4.9 oraz 4.10 spełnia semantykę własności możliwej binawigowalności słownika. W przypadku implementacji w metamodelu asocjacyjnym, binawigowalność jest własnością domyślną, co powoduje, iż własność możliwa binawigowalność słownika jest domyślnie spełniona przez modele asocjacyjne realizujące ten wzorzec. AOM dopuszcza jednak implementacje, w których występuje uninawigowalność. Przekładając to na przyjęty system pojęciowy, realizacja uninawigowalności stanowiłaby zanegowanie tej własności (rys. 4.13).



Rysunek 4.13. Diagram AML wzorca słownik implementującego konieczne atomy semantyczne wraz z zanegowaniem własności możliwej binawigowalności słownika

$$\diamond \textit{ElementOpisywany} [*] \xrightarrow{+\textit{Opisywanie}(\square \textit{WyliczenieBytow})} [0..1] \square \textit{WyliczenieBytow} \quad (4.11)$$

$$\square \textit{WyliczenieBytow} \left\{ +\textit{atrybut} : \textit{ascii}(32) \right\}; \quad (4.12)$$



Rysunek 4.14. Diagram AML wzorca słownik implementującego konieczne atomy semantyczne oraz własność możliwą:  $n$ -ograniczone użycia dla  $n = 10$

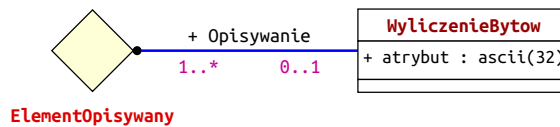
Implementacja koniecznych atomów semantycznych wzorca słownik z własnością możliwą:  $n$ -ograniczone użycie słownika

$$\diamond \text{ElementOpisywany} [0..10] \xrightarrow{+\text{Opisywanie}(\square \text{WyliczenieBytow})} [0..1] \square \text{WyliczenieBytow} \quad (4.13)$$

$$\square \text{WyliczenieBytow} \left\{ +\text{atrybut} : \text{ascii}(32) \right\}; \quad (4.14)$$

Rysunek 4.14 przedstawia realizację wzorca listy wraz z własnością możliwą  $n$ -ograniczenie użycia dla  $n = 10$ . Implementacja zakłada realizację wszystkich koniecznych atomów semantycznych wzorca, wraz z atomami semantycznymi M.2, M.2.1 oraz M.2.2. Implementacja atomu semantycznego M.2 polega na utworzeniu nowego modelu słownika, natomiast implementacja atomów M.2.1 oraz M.2.2 została zrealizowana poprzez określenie krotności po stronie asocjacji w roli *Opisywanie* asocjacji  $\diamond \text{ElementOpisywany}$  jako 0..10.

Implementacja koniecznych atomów semantycznych wzorca słownik z własnością możliwą:  $n$ -minimalna opisywalność słownika



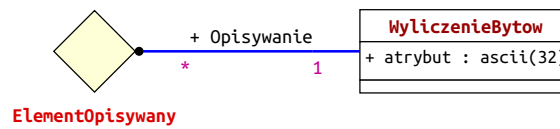
Rysunek 4.15. Diagram modelu asocjacyjnego wzorca słownik implementującego konieczne atomy semantyczne oraz własność możliwą:  $n$ -minimalna opisywalność dla  $n = 1$

$$\diamond \text{ElementOpisywany} [1..*] \xrightarrow{+\text{Opisywanie}(\square \text{WyliczenieBytow})} [0..1] \square \text{WyliczenieBytow} \quad (4.15)$$

$$\square \text{WyliczenieBytow} \left\{ +\text{atrybut} : \text{ascii}(32) \right\}; \quad (4.16)$$

Rysunek 4.15 przedstawia realizację wzorca słownika wraz z własnością możliwą ***n*--minimalna opisywalność słownika** dla  $n = 1$ . Implementacja zakłada realizację wszystkich koniecznych atomów semantycznych wzorca, wraz z atomami semantycznymi M.3, M.3.1 oraz M.3.2. Implementacja atomu semantycznego M.3 polega na utworzeniu nowego modelu słownika, natomiast implementacja M.3.1 oraz M.3.2 została zrealizowana poprzez określenie krotności po stronie asocjacji w roli *Opisywanie* asocjacji  $\diamond ElementOpisywany$  jako  $1..*$ .

Implementacja koniecznych atomów semantycznych wzorca słownik z własnością możliwą: ***obligatoryjność opisywania***



Rysunek 4.16. Diagram modelu asocjacyjnego wzorca słownik implementującego konieczne atomy semantyczne oraz własność możliwą: obligatoryjność opisywania

$$\diamond ElementOpisywany [*] \xrightarrow{+Opisywanie(\square WyliczenieBytow)} [1] \square WyliczenieBytow \quad (4.17)$$

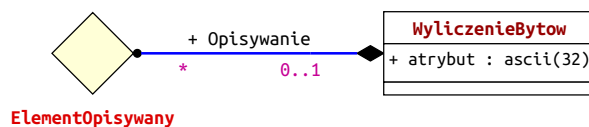
$$\square WyliczenieBytow \left\{ +atrybut : ascii(32) \right\}; \quad (4.18)$$

Rysunek 4.16 przedstawia realizację wzorca słownika wraz z własnością możliwą ***obligatoryjność opisywania***. Implementacja zakłada realizację wszystkich koniecznych atomów semantycznych wzorca, wraz z atomami semantycznymi M.4 oraz M.4.1. Implementacja atomu semantycznego M.4 polega na utworzeniu nowego modelu słownika, natomiast implementacja M.4.1 oraz M.4.2 została zrealizowana poprzez określenie krotności po stronie asocjacji w roli *Opisywanie* asocjacji  $\diamond ElementOpisywany$  jako 1.

Implementacja koniecznych atomów semantycznych wzorca słownik z własnością możliwą: ***zależność istnienia elementu opisywanego od opisującego***

$$\diamond ElementOpisywany [*] \xrightarrow{+Opisywanie(\square WyliczenieBytow)} [0..1] \square WyliczenieBytow \quad (4.19)$$

$$\square WyliczenieBytow \left\{ +atrybut : ascii(32) \right\}; \quad (4.20)$$



Rysunek 4.17. Diagram modelu asocjacyjnego wzorca słownik implementującego konieczne atomy semantyczne oraz własność możliwą: zależność istnienia elementu opisywanego od opisującego

Rysunek 4.17 przedstawia realizację wzorca słownika wraz z własnością możliwą zależność istnienia elementu opisywanego od opisującego. Implementacja zakłada realizację wszystkich koniecznych atomów semantycznych wzorca, wraz z atomami semantycznymi M.5, M.5.1 oraz M.5.2. Implementacja M.5 polega na utworzeniu nowego modelu słownika, natomiast implementacja atomów M.5.1 oraz M.5.2 została zrealizowana poprzez użycie kompozycji w ramach roli *Opisywanie* asocjacji  $\diamond$ *ElementOpisywany* po stronie elementu wiążanego.

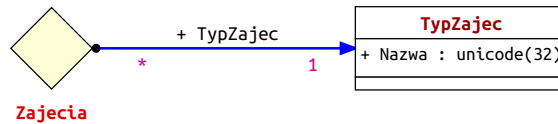
*Implementacja koniecznych atomów semantycznych wzorca słownik z własnością możliwą: uporządkowanie słownika*

Implementacja wzorca słownika przedstawiona na wyr. 4.9 oraz 4.10 spełnia semantykę własności możliwej uporządkowanie słownika, ponieważ realizuje atomy semantyczne M.6, M.6.1 i M.6.2. W metamodelu asocjacyjnym wszystkie zbiorowości, takie jak kolekcje oraz role są uporządkowane. W efekcie, jedynie co jest możliwe to zignorowanie istniejącego porządku.

### Przykład zastosowania

Przykładem zastosowania słownika może być wykorzystanie tego wzorca do zamodelowania typu zajęć na uczelni wyższej. Wyróżnia się kilka rodzajów zajęć, np. wykłady, ćwiczenia, laboratoria, projekty, seminaria. W modelu danych reprezentującym siatkę studiów może znaleźć się potrzeba określenia, jakiego typu będą zajęcia w ramach konkretnego kursu. Typ zajęć pełnić będzie zatem rolę uczestnika WYLICZENIA BYTÓW, a zajęcia ELEMENTU OPISYWANEGO. Uproszczoną implementację fragmentu takiego modelu przedstawiono na rys. 4.18. W ramach implementacji wzorca słownika, wykorzystana została następująca własność możliwa:

- obligatoryjność opisywania, ponieważ od każdego zajęcia wymaga się posiadanie dokładnie jednego, określonego typu tych zajęć.



Rysunek 4.18. Przykład zastosowania wzorca modelowania słownik

### 4.1.3. Autoreferencja

#### Opis problemu

Autoreferencja rozwiązuje problem związany z potrzebą odwołania się pewnego elementu do elementu tego samego typu (tzn. do samego siebie – w sensie intencjonalnym). Zastosowanie tego wzorca pozwala tworzyć rekurencyjne struktury danych, takie jak np. łańcuchy, drzewa, grafy [107].

#### Elementy składowe

Pojęcie stanowiące uczestnika wzorca:

1. ELEMENT AUTOREFERENCYJNY – element, który uczestniczy w związku autoreferencyjnym.

Pojęcie stanowiące powiązanie uczestnika z nim samym:

1. ZWIĄZEK AUTOREFERENCYJNY – powiązanie elementów, które są tego samego typu. Powiązanie to realizuje autoreferencję. W ramach powiązania, elementy pełnią różne role autoreferencyjne (co najmniej dwie), które są charakteryzowane jako źródłowe lub docelowe.

#### Własności

Pojęcia stanowiące własności wzorca:

1. BINAWIGOWALNOŚĆ – własność parametryzowalna względem roli autoreferencji, która określa, czy ELEMENT AUTOREFERENCYJNY posiada informacje o istniejącym połączeniu zwrotnym. Dla ról źródłowych jest to informacja o połączeniu pomiędzy ZWIĄZKIEM AUTOREFERENCYJNYM, a ELEMENTEM AUTOREFERENCYJNYM, natomiast dla ról docelowych – odwrotnie.
2. *n*-ARNOŚĆ – własność wzorca określająca ile referencji zachodzi pomiędzy ELEMENTEM AUTOREFERENCYJNYM w ramach jednego ZWIĄZKU AUTOREFERENCYJNEGO.
3. *n*-MINIMALNOŚĆ ROLI AUTOREFERENCJI – własność parametryzowalna względem roli autoreferencji, która określa ile minimalnie ELEMENTÓW AUTOREFERENCYJNYCH występuje w ramach danej roli autoreferencji.

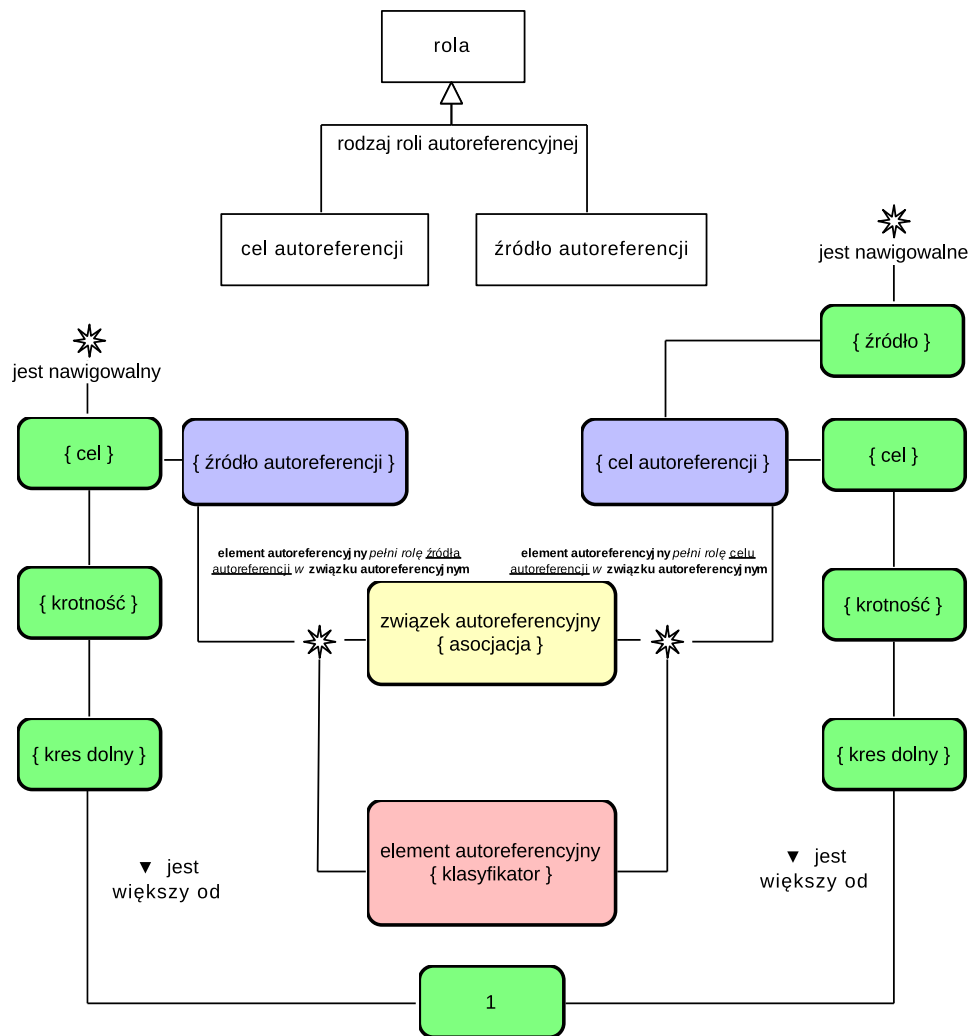
4. *n*-MAKSYMALNOŚĆ ROLI AUTOREFERENCJI – własność parametryzowalna względem roli autoreferencji, która określa ile maksymalnie ELEMENTÓW AUTOREFERENCYJNYCH występuje w ramach danej roli autoreferencji.
5. *n*-POWTARZALNOŚĆ ROLI AUTOREFERENCJI – własność parametryzowalna względem roli autoreferencji, która określa ile razy dany ELEMENT AUTOREFERENCYJNY (w sensie ekstensjonalnym) może wystąpić w ramach danej roli.
6. ZALEŻNOŚĆ ISTNIENIA ELEMENTU OD AUTOREFERENCJI – własność parametryzowalna względem roli autoreferencji, która określa, czy w ujęciu behawioralnym podczas przerywania istnienia ZWIĄZKU AUTOREFERENCYJNEGO, następuje zakończenie czasu życia ELEMENTU AUTOREFERENCYJNEGO.
7. ZALEŻNOŚĆ ISTNIENIA ZWIĄZKU AUTOREFERENCJI OD ELEMENTU – własność parametryzowalna względem roli autoreferencji, która określa, czy w ujęciu behawioralnym podczas przerywania istnienia ELEMENTU AUTOREFERENCYJNEGO, następuje zakończenie czasu życia ZWIĄZKU AUTOREFERENCYJNEGO.
8. UPORZĄDKOWANIE ROLI AUTOREFERENCJI – własność parametryzowalna względem roli autoreferencji, która określa, czy kolejność ELEMENTÓW AUTOREFERENCYJNYCH w ramach tej roli jest istotna.

### Specyfikacja wzorca

Diagram pojęć na rys. 4.19 przedstawia konieczne atomy semantyczne wzorca autoreferencji.

*Konieczne atomy semantyczne*

- K.1 autoreferencja *jest konkretyzacją* koncepcji wzorca
- K.2 autoreferencja *zawsze posiada* element autoreferencyjny, *który jest konkretyzacją klasyfikatora*
- K.3 autoreferencja *zawsze posiada* związek autoreferencyjny, *który jest konkretyzacją asocjacji*
  - K.3.1 element autoreferencyjny *zawsze pełni rolę* źródła autoreferencji *w* związku autoreferencyjnym
    - K.3.1.1 cel źródła autoreferencji *zawsze jest cechowany przez* krotność, *która posiada kres dolny*, *który jest większy lub równy* 1
    - K.3.1.2 cel źródła autoreferencji *zawsze jest nawigowalny*



Rysunek 4.19. Diagram pojęć prezentujący konieczne atomy semantyczne koncepcji wzorca autoreferencja

### K.3.2 element autoreferencyjny *zawsze pełni rolę* celu autoreferencji *w* związku autoreferencyjnym

K.3.2.1 cel celu autoreferencji *zawsze jest cechowany przez* krotność, *która posiada* kres dolny, *który jest większy lub równy* 1

K.3.2.2 źródło celu autoreferencji *zawsze jest* nawigowalny

K.3.3 *każda* inherentna rola *posiadana przez* związek autoreferencyjny *jest konkretyzacją* albo celu autoreferencji albo źródła autoreferencji

*Atomy semantyczne własności możliwych*

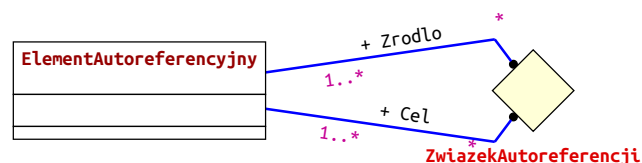
M.1 *n*-arność autoreferencji *dla*  $n > 2$  *jest konkretyzacją* własności wzorca

M.1.1 związek autoreferencyjny *posiada* *dokładnie*  $n$  inherentnych ról

- M.2 binawigowalność roli autoreferencji *r* jest konkretyzacją własności wzorca
- M.2.1 jeżeli *r* jest konkretyzacją źródła autoreferencji to źródło *r* jest nawigowalne lub jeżeli *r* jest konkretyzacją celu autoreferencji to cel *r* jest nawigowalny
- M.3 *n*-minimalność roli autoreferencji *r* jest konkretyzacją własności wzorca
- M.3.1 cel *r* jest cechowany przez krotność, która posiada kres dolny, który jest równy *n*
- M.4 *n*-maksymalność roli autoreferencji *r* jest konkretyzacją własności wzorca
- M.4.1 cel *r* jest cechowany przez krotność, która posiada kres górny, który jest równy *n*
- M.5 *n*-powtarzalność roli autoreferencji *r* jest konkretyzacją własności wzorca
- M.5.1 *r* jest cechowana przez powtarzalność, która jest określana przez *n*
- M.6 zależność istnienia elementu od autoreferencji *r* jest konkretyzacją własności wzorca
- M.6.1 źródło *r* jest cechowane przez zależność czasu istnienia
- M.7 zależność istnienia autoreferencji od elementu *r* jest konkretyzacją własności wzorca
- M.7.1 cel *r* jest cechowany przez zależność czasu istnienia
- M.8 uporządkowanie roli autoreferencji *r* jest konkretyzacją własności wzorca
- M.8.1 cel *r* jest cechowany przez uporządkowanie

### Implementacja rozwiązania

*Implementacja koniecznych atomów semantycznych wzorca autoreferencji*



Rysunek 4.20. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne



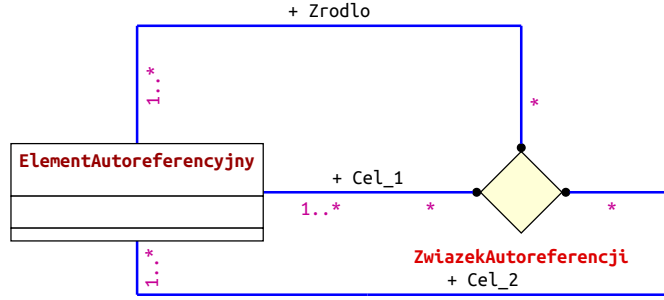
$$\begin{array}{c}
\Diamond ZwiasekAutoreferencji \\
\left( \begin{array}{c}
[*] \xrightarrow{+Zrodlo(\square ElementAutoreferencyjny)} [1..*] \\
[*] \xrightarrow{+Cel(\square ElementAutoreferencyjny)} [1..*]
\end{array} \right) \\
\square ElementAutoreferencyjny;
\end{array} \tag{4.21}$$

Na rysunku 4.20 przedstawiono implementację koniecznych atomów semantycznych wzorca autoreferencji. Poniżej przedstawiono odniesienie kolejnych elementów modelu do realizacji koniecznych atomów semantycznych wzorca.

1. Utworzenie modelu asocjacyjnego realizuje konieczny atom semantyczny K.1.
2. K.2 został zaimplementowany poprzez utworzenie kolekcji  $\square ElementAutoreferencyjny$ . Należy dodatkowo zauważyć, iż równie poprawną implementacją byłoby wykorzystanie instancji koncepcji wzorca bikompozycyjny tandem asocjacja-kolekcja<sup>2</sup> lub asocjacji.
3. K.3 został zrealizowany poprzez utworzenie asocjacji  $\Diamond ZwiasekAutoreferencyjny$ .
4. Implementację K.3.1 osiągnięto poprzez utworzenie roli *Zrodlo* w asocjacji  $\Diamond ZwiasekAutoreferencyjny$ .
5. K.3.1.1 został zrealizowany poprzez określenie krotności roli *Zrodlo* w asocjacji  $\Diamond ZwiasekAutoreferencyjny$  jako 1..\*.
6. Fakt, iż role w metamodelu asocjacyjnym są domyślnie dwukierunkowe, realizuje K.3.1.2.
7. Implementacja K.3.2 polega na utworzeniu roli *Cel* w asocjacji  $\Diamond ZwiasekAutoreferencyjny$ .
8. Implementacja K.3.2.1 opiera się o określenie krotności roli *Cel* w asocjacji  $\Diamond ZwiasekAutoreferencyjny$  jako 1..\*.
9. Implementacja K.3.3 wynika z faktu, iż semantyka źródła autoreferencji oraz celu autoreferencji została nadana dwóm oddzielnym rolom w asocjacji  $\Diamond ZwiasekAutoreferencyjny$ , stanowiącym odrębne od siebie w sensie strukturalnym byty.

*Implementacja koniecznych atomów semantycznych wzorca autoreferencji z własnością możliwą: n-arność autoreferencji dla  $n = 3$ .*

<sup>2</sup> Zob. rozdział 4.2.1



Rysunek 4.21. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne oraz własność możliwą  $n$ -arność autoreferencji dla  $n = 4$

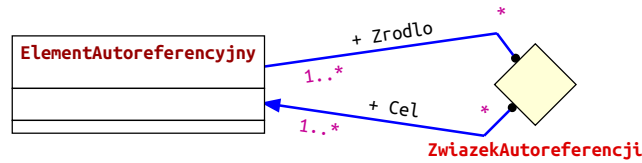
$$\begin{aligned}
 & \diamond \text{ZwiasekAutoreferencji} \\
 & \left( \begin{array}{l}
 [*] \xrightarrow{+Zrodlo(\square \text{ElementAutoreferencyjny})} [1..*] \\
 [*] \xrightarrow{+Cel_1(\square \text{ElementAutoreferencyjny})} [1..*] \\
 [*] \xrightarrow{+Cel_2(\square \text{ElementAutoreferencyjny})} [1..*]
 \end{array} \right) \quad (4.22) \\
 & \square \text{ElementAutoreferencyjny};
 \end{aligned}$$

Rysunek 4.21 przedstawia realizację wzorca autoreferencji wraz z własnością możliwą  $n$ -arność referencji dla  $n = 3$ . Implementacja zakłada zrealizowanie wszystkich atomów semantycznych wzorca oraz dodatkowo atomy semantyczne zakładające spełnienie tej własności możliwej: M.1 oraz M.1.1. Atom semantyczny M.1 został spełniony poprzez utworzenie nowego modelu asocjacyjnego, natomiast M.1.1 został spełniony poprzez zapewnienie możliwości przypisania semantyki źródła autoreferencji oraz celu autoreferencji do więcej, niż jednej roli. W przykładowej implementacji zmnożono role o semantyce celu autoreferencji, mające odzwierciedlenie w postaci ról  $Cel_1$  oraz  $Cel_2$  asocjacji  $\diamond \text{ZwiasekAutoreferencji}$ . Informacja ta, z punktu widzenia modelu asocjacyjnego, wymaga dodatkowego opisu, który uwzględni, jaką rolę w sensie wzorca autoreferencji pełni konkretna rola (*Role*) w sensie modelowania asocjacyjnego.

*Implementacja koniecznych atomów semantycznych wzorca autoreferencji z własnością możliwą:* binawigowalność roli autoreferencji.

$$\begin{aligned}
 & \diamond \text{ZwiasekAutoreferencji} \\
 & \left( \begin{array}{l}
 [*] \xrightarrow{+Zrodlo(\square \text{ElementAutoreferencyjny})} [1..*] \\
 [*] \xrightarrow{+Cel(\square \text{ElementAutoreferencyjny})} [1..*]
 \end{array} \right) \quad (4.23) \\
 & \square \text{ElementAutoreferencyjny};
 \end{aligned}$$

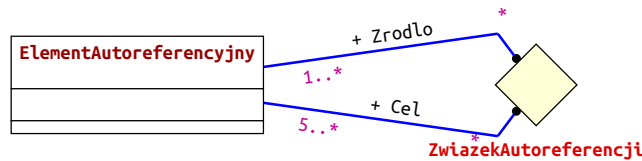
Implementacja wzorca autoreferencji, która realizuje wszystkie konieczne



Rysunek 4.22. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne oraz zanegowaną własność możliwą binawigowalność roli autoreferencji dla  $r = \diamond ZwiasekAutoreferencji \xrightarrow{+Cel} \square ElementAutoreferencyjny$

atomy semantyczne tego wzorca (rys. 4.20) realizuje własność możliwą binawigowalność roli autoreferencyjnej dla każdej z ról. Dla metamodelu asocjacyjnego binawigowalność ról jest własnością domyślną. Należy jednak zauważyć, iż możliwe jest zanegowanie tej własności w kierunku źródło  $\rightarrow$  cel poprzez utworzenie roli uninawigowalnej – implementację taką przedstawiono dla uninawigowalnej roli *Cel* w asocjacji  $\diamond ZwiasekAutoreferencyjny$  (rys. 4.22).

Implementacja koniecznych atomów semantycznych wzorca autoreferencji z własnością możliwą:  $n$ -minimalność roli autoreferencji dla  $n = 5$  oraz  $r = \diamond ZwiasekAutoreferencyjny \xrightarrow{+Cel} \square ElementAutoreferencyjny$ .



Rysunek 4.23. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne oraz własność możliwą  $n$ -minimalność roli autoreferencji dla  $n = 5$  oraz  $r = \diamond ZwiasekAutoreferencyjny \xrightarrow{+Cel} \square ElementAutoreferencyjny$

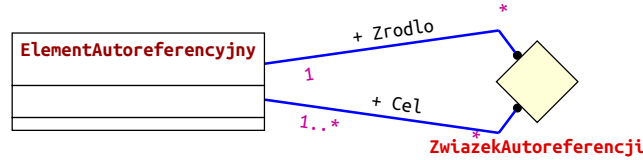
$$\left( \begin{array}{l} [*] \xrightarrow{+Zrodlo(\square ElementAutoreferencyjny)} [1..*] \\ [*] \xrightarrow{+Cel(\square ElementAutoreferencyjny)} [5..*] \end{array} \right) \quad (4.24)$$

$\square ElementAutoreferencyjny$ ;

Rysunek 4.23 przedstawia realizację wzorca autoreferencji wraz z własnością możliwą  $n$ -minimalność roli autoreferencji dla  $n = 5$ . Własność ta została zobrażowana dla  $r$  będącego rolą *Cel* w asocjacji  $\diamond ZwiasekAutoreferencyjny$ . Implementacja zakłada zrealizowanie wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomy semantyczne zakładające spełnienie tej własności możliwej: M.3 oraz M.3.1. Atom semantyczny M.3. został zrealizowany poprzez utworzenie nowego

modelu cechowanego przez tę własność, natomiast M.3.1 został zrealizowany poprzez modyfikację krotności roli *Cel* asocjacji  $\diamond$ *ZwiazekAutoreferencji*, uwzględniając parametr  $n$ , tj. na krotność  $5..*$ .

*Implementacja koniecznych atomów semantycznych wzorca autoreferencji z własnością możliwą:  $n$ -maksymalność roli autoreferencji dla  $n = 1$  oraz*  
 $r = \diamond$ *ZwiazekAutoreferencji*  $\bullet \xrightarrow{+Zrodlo}$   $\square$ *ElementAutoreferencyjny*.



Rysunek 4.24. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne oraz własność możliwą  $n$ -maksymalność roli autoreferencji dla  $n = 1$  oraz  $r = \diamond$ *ZwiazekAutoreferencji*  $\bullet \xrightarrow{+Zrodlo}$   $\square$ *ElementAutoreferencyjny*

$$\left( \begin{array}{l} \diamond \text{ZwiazekAutoreferencji} \\ \left[ \begin{array}{l} [*] \bullet \xrightarrow{+Zrodlo(\square \text{ElementAutoreferencyjny})} [1] \\ [*] \bullet \xrightarrow{+Cel(\square \text{ElementAutoreferencyjny})} [1..*] \end{array} \right] \end{array} \right) \quad (4.25)$$

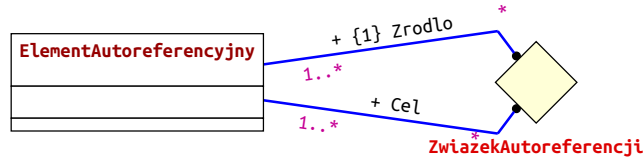
$\square$ *ElementAutoreferencyjny*;

Rysunek 4.24 przedstawia realizację wzorca autoreferencji wraz z własnością możliwą  $n$ -**maksymalność roli autoreferencji** dla  $n = 1$ . Własność ta została zobrażowana dla  $r$  będącego rolą *Zrodlo* w asocjacji  $\diamond$ *ZwiazekAutoreferencji*. Implementacja zakłada zrealizowanie wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomy semantyczne zakładające spełnienie tej własności możliwej: M.4 oraz M.4.1. Atom semantyczny M.4 został zrealizowany poprzez utworzenie nowego modelu cechowanego przez tę własność, natomiast M.4.1 został zrealizowany poprzez modyfikację krotności roli *Cel* w asocjacji  $\diamond$ *ZwiazekAutoreferencji*, uwzględniając parametr  $n$ , tj. 1.

*Implementacja koniecznych atomów semantycznych wzorca autoreferencji z własnością możliwą:  $n$ -powtarzalność roli autoreferencji dla  $n = 1$  oraz  $r =$*   
 $\diamond$ *ZwiazekAutoreferencji*  $\bullet \xrightarrow{\{1\}+Zrodlo}$   $\square$ *ElementAutoreferencyjny*.

$$\left( \begin{array}{l} \diamond \text{ZwiazekAutoreferencji} \\ \left[ \begin{array}{l} [*] \bullet \xrightarrow{\{1\}+Zrodlo(\square \text{ElementAutoreferencyjny})} [1..*] \\ [*] \bullet \xrightarrow{+Cel(\square \text{ElementAutoreferencyjny})} [1..*] \end{array} \right] \end{array} \right) \quad (4.26)$$

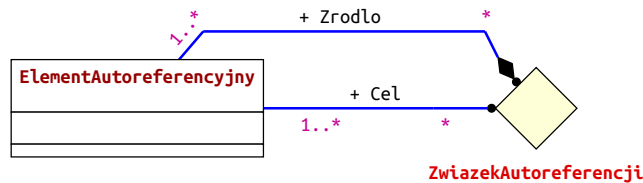
$\square$ *ElementAutoreferencyjny*;



Rysunek 4.25. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne oraz własność możliwą  $n$ -powtarzalność roli autoreferencji dla  $n = 1$  oraz  $r = \diamond ZwiasekAutoreferencji \xrightarrow{\{1\}+Zrodlo} \square ElementAutoreferencyjny$

Na rysunku 4.25 przedstawiono realizację wzorca autoreferencji wraz z własnością możliwą  $n$ -powtarzalność roli autoreferencji dla  $n = 1$ . Własność ta została przedstawiona dla roli  $r$  będącego rolą  $Zrodlo$  w asocjacji  $\diamond ZwiasekAutoreferencji$ . Implementacja uwzględnia zrealizowanie wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomy semantyczne zakładające spełnienie tej własności możliwej: M.5 oraz M.5.1. Atom semantyczny M.5 został zrealizowany poprzez utworzenie nowego modelu cechowanego przez tę własność, natomiast M.5.1 został zrealizowany poprzez modyfikację roli  $Zrodlo$  w asocjacji  $\diamond ZwiasekAutoreferencji$  w taki sposób, aby określić wartość unikalności jako 1.

Implementacja koniecznych atomów semantycznych wzorca autoreferencji z własnością możliwą: zależność istnienia elementu od autoreferencji dla  $r = \diamond ZwiasekAutoreferencji \xrightarrow{+Zrodlo} \square ElementAutoreferencyjny$ .



Rysunek 4.26. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne oraz własność możliwą zależność istnienia elementu od autoreferencji dla  $r = \diamond ZwiasekAutoreferencji \xrightarrow{+Zrodlo} \square ElementAutoreferencyjny$

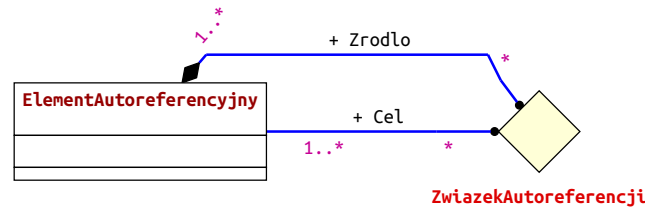
$$\left( \begin{array}{l} \diamond ZwiasekAutoreferencji \\ \begin{array}{l} +Zrodlo(\square ElementAutoreferencyjny) [1..*] \\ +Cel(\square ElementAutoreferencyjny) [1..*] \end{array} \end{array} \right) \quad (4.27)$$

$\square ElementAutoreferencyjny$ ;

Rysunek 4.26 przedstawia realizację wzorca autoreferencji wraz z własnością możliwą zależność istnienia elementu od autoreferencji, która została spa-

rametryzowana względem roli  $r$  dla  $r$  będącego rolą  $Zrodlo$  w asocjacji  $\diamond ZwiazekAutoreferencji$ . Implementacja zakłada zrealizowanie wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomy semantyczne zakładające spełnienie tej własności możliwej: M.6 oraz M.6.1. Model posiadający wzorzec charakteryzowany przez tę własność realizuje atom semantyczny M.6. Implementacja M.6.1 została zrealizowana poprzez określenie roli  $Zrodlo$  w asocjacji  $\diamond ZwiazekAutoreferencji$  jako komponowanej przez asocjacje.

Implementacja koniecznych atomów semantycznych wzorca autoreferencji z własnością możliwą: zależność istnienia autoreferencji od elementu dla  $r = \diamond ZwiazekAutoreferencji \xrightarrow{+Zrodlo} \square ElementAutoreferencyjny$ .



Rysunek 4.27. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne oraz własność możliwą zależność istnienia autoreferencji od elementu dla  $r = \diamond ZwiazekAutoreferencji \xrightarrow{+Zrodlo} \square ElementAutoreferencyjny$

$$\left( \begin{array}{c} \diamond ZwiazekAutoreferencji \\ \left[ \begin{array}{c} [*] \xrightarrow{+Zrodlo(\square ElementAutoreferencyjny)} [1..*] \\ [*] \xrightarrow{+Cel(\square ElementAutoreferencyjny)} [1..*] \end{array} \right] \end{array} \right) \quad (4.28)$$

$\square ElementAutoreferencyjny$ ;

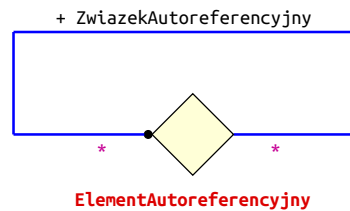
Rysunek 4.27 przedstawia realizację wzorca autoreferencji wraz z własnością możliwą zależność istnienia autoreferencji od elementu, która została sparametryzowana względem roli  $r$  dla  $r$  będącego rolą  $Zrodlo$  w asocjacji  $\diamond ZwiazekAutoreferencji$ . Implementacja zakłada zrealizowanie wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomy semantyczne zakładające spełnienie tej własności możliwej: M.7 oraz M.7.1. Model posiadający wzorzec charakteryzowany przez tę własność realizuje atom semantyczny M.7. Implementacja M.7.1 została zrealizowana poprzez określenie roli  $Zrodlo$  w asocjacji  $\diamond ZwiazekAutoreferencji$  jako komponowanej przez element wiązany.

Implementacja koniecznych atomów semantycznych wzorca autoreferencji z własnością możliwą: uporządkowanie autoreferencji

Implementacja wzorca autoreferencji przedstawiona na rys. 4.20 spełnia semantykę własności możliwej uporządkowanie autoreferencji, ponieważ realizuje atomy semantyczne M.8 oraz M.8.1.

*Implementacja wzorca autoreferencji charakterystyczna dla metamodelu asocjacyjnego oparta o autoreferencyjną rolę*

Przedstawione powyżej opracowane implementacje wzorca autoreferencji zawierają odwzorowanie elementu autoreferencyjnego jako kolekcję  $\square ElementAutoreferencyjny$ . W ramach modelowania asocjacyjnego silnie opartego o związki, bardzo często stosowaną konstrukcją autoreferencyjną jest autoreferencja oparta o asocjację. Jest to bardzo silna konstrukcja, niewystępująca w innych metamodelach pozwalająca konstruować związki zwrotne, w ramach których rola spełnia funkcję autoreferencyjności. Przykład takiej implementacji przedstawiono na rys. 4.28 i wyr. 4.29. Poniżej zostanie przedstawiona implementacja poszczególnych koniecznych atomów semantycznych dla takiej, asocjacyjnej wersji autoreferencji.



Rysunek 4.28. Diagram modelu asocjacyjnego wzorca autoreferencji opartego o autoreferencyjną rolę

$$\begin{array}{c}
 \diamond ElementAutoreferencyjny \\
 [*] \xrightarrow{+ZwiazekAutoreferencyjny(\diamond ElementAutoreferencyjny)} [*] \quad (4.29) \\
 \diamond ElementAutoreferencyjny
 \end{array}$$

1. Utworzenie odrębnego modelu asocjacyjnego realizuje konieczny atom semantyczny K.1.
2. K.2 został zaimplementowany poprzez utworzenie asocjacji  $\diamond ElementAutoreferencyjny$ .
3. K.3, K.3.1 oraz K.3.1.2 zostały zrealizowane poprzez utworzenie roli *ZwiazekAutoreferencyjny* w asocjacji  $\diamond ElementAutoreferencyjny$ .
4. K.3.1.1 został zaimplementowany w sposób automatyczny, ze względu na fakt, iż w AOM każda rola w sensie ekstensjonalnym (obiekt roli) odpowiada dokładnie jednemu obiektowi asocjacji. Oznacza to natomiast, iż podczas takiej implementacji powstaje nadmiarowa semantyka związana z tym, iż:

- a) cel źródła autoreferencji jest cechowany przez krotność, która posiada kres górny równy 1,
- b) źródło źródła autoreferencji jest cechowane przez krotność, która posiada kres górny i kres dolny równe 1.

W konsekwencji asocjacyjna implementacja autoreferencji opartej o autoreferencyjną rolę wymusza, że w ramach jednej instancji związku autoreferencyjnego dokładnie jeden obiekt asocjacji będzie musiał stanowić jej źródło.

5. Implementację K.3.2 osiągnięto poprzez określenie celu tej roli jako asocjację  $\diamond$ *ElementAutoreferencyjny*.
6. Nie jest możliwe przy pomocy ograniczeń metamodelu zapewnienie, aby każda instancja modelu spełniała atom semantyczny K.3.2.1. Jednakże, nie istnieją żadne ograniczenia, aby ten atom semantyczny mógł być spełniony.
7. Implementacja K.3.2.2 realizowana jest domyślnie w sposób automatyczny. Wynika z domyślnej obustronnej nawigowalności ról.
8. Implementacja K.3.3 wynika z faktu, iż role w metamodelu asocjacyjnym są wewnętrznie binarne i zawsze muszą mieć określone cel (asocjację lub kolekcję pełniącą daną rolę) oraz właściciela (asocjację posiadającą daną rolę).

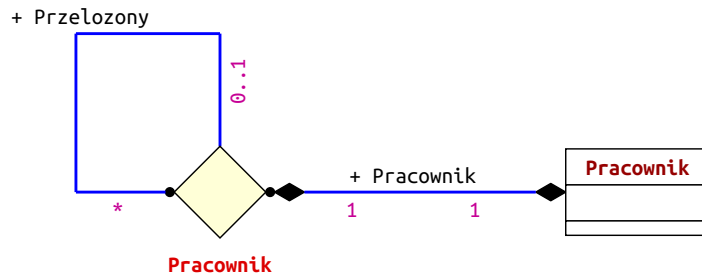
### Przykład zastosowania

Przykładem zastosowania wzorca autoreferencji w modelowaniu asocjacyjnym może być przykład związku *podwładny-przełożony* pomiędzy pracownikami. Przy założeniu, że pracownik posiada reprezentację w postaci BACT lub asocjacji, możliwa jest do skonstruowania autoreferencyjna rola, która wskazuje na obiekt asocjacji reprezentujący przełożonego (rys. 4.29). W ramach takiej konstrukcji asocjacja  $\diamond$ *Pracownik* pełni funkcję ELEMENTU AUTOREFERENCYJNEGO, natomiast rola  $\diamond$ *Pracownik*  $\xrightarrow{+Przełożony}$   $\diamond$ *Pracownik* pełni rolę ZWIĄZKU AUTOREFERENCYJNEGO. Model dodatkowo ma następujące własności możliwe:

- binawigowalność źródła autoreferencji w implementacji korzystającej z autoreferencyjnej roli jest własnością domyślną i jedyną możliwą,
- binawigowalność celu autoreferencji w celu możliwości nawigowania od celu do źródła, tj. od przełożonego do podporządkowanych mu pracowników,
- 1-maksymalność celu autoreferencji w celu ograniczenia liczby obiektów asocjacji reprezentujących przełożonego do maksymalnie jednego.

Należy dodatkowo zauważyć, iż model asocjacyjny przedstawiony na rys. 4.29 nie posiada możliwości wyrażenia ograniczenia związanego z acyklicznością struktury drzewiastej na poziomie ekstensjonalnym. Oznacza to, że model zawierający cykle w części ekstensjonalnej zostałby uznany za poprawny (zgodny z metamodelem),





Rysunek 4.29. Przykład zastosowania wzorca autoreferencji

choć niepoprawny ze względu na założenia dziedzinowe. Szczegółowo zagadnienie to omówiono w rozdziale 3.2 publikacji [107].

#### 4.1.4. Pęk ról

##### Opis problemu

Wzorzec pęku ról rozwiązuje problem polegający na powiązaniu więcej niż dwóch elementów danych przy pomocy związku, przy czym każdy z tych elementów może pełnić różne role w ramach tego związku. Jednocześnie, konieczne jest zapewnienie możliwości, aby każdy z elementów pęku ról mógł mieć różną strukturę. Powiązaną z pękiem ról techniką jest modelowanie oparte o związki  $n$ -arne [46, 89, 122, 130] lub hipersieci [81].

##### Elementy składowe

Pojęcia stanowiące uczestników wzorca:

1. ELEMENT ZWIĄZANY<sub>N</sub> – element, który pełni rolę wchodzącą w skład WIĄZKI. Jest to uczestnik generyczny<sup>3</sup>, tzn. w skład wzorca wchodzi co najwyżej N ELEMENTÓW ZWIĄZANYCH, gdzie N jest równe liczbie ról w ramach WIĄZKI.

Pojęcie stanowiące powiązanie między uczestnikami:

1. WIĄZKA – element wiążący, który składa się z dokładnie N elementów pęku, przy czym N jest większe od 2.

##### Własności

1. OPISANIE WIĄZKI – własność, która określa, czy wymagane są dodatkowe dane, które określają wiązkę.

<sup>3</sup> Przez pojęcie określanie terminem *uczestnik generyczny* rozumie się fakt, iż stanowi on element szablonu wzorca, w którym możliwe jest multiplikowanie wystąpień tego elementu na poziomie intensjonalnym.

2. *n*-POWTARZALNOŚĆ ELEMENTU PĘKU – własność parametryzowalna względem elementu pęku, która określa ile razy dany ELEMENT ZWIĄZANY (w sensie ekstensjonalnym) może wystąpić w ramach danego elementu pęku.
3. *n*-MINIMALNOŚĆ ELEMENTU PĘKU – własność parametryzowalna względem elementu pęku, która określa ile minimalnie ELEMENTÓW ZWIĄZANYCH występuje w ramach danego elementu pęku.
4. *n*-MAKSYMALNOŚĆ ELEMENTU PĘKU – własność parametryzowalna względem elementu pęku, która określa ile maksymalnie ELEMENTÓW ZWIĄZANYCH występuje w ramach danego elementu pęku.
5. *n*-OGRANICZONA PRZYNALEŻNOŚĆ DO WIĄZKI – własność parametryzowalna względem elementu pęku, która określa, w ilu WIĄZKACH maksymalnie dany ELEMENT ZWIĄZANY może być związany.
6. ZALEŻNOŚĆ ISTNIENIA WIĄZKI OD ELEMENTU PĘKU – własność parametryzowalna względem elementu pęku, która określa, czy w ujęciu behawioralnym podczas przerywania istnienia ELEMENTU ZWIĄZANEGO, następuje zakończenie czasu życia WIĄZKI.
7. ZALEŻNOŚĆ ISTNIENIA ELEMENTU PĘKU OD WIĄZKI – własność parametryzowalna względem elementu pęku, która określa, czy w ujęciu behawioralnym podczas przerywania istnienia WIĄZKI, następuje zakończenie czasu życia ELEMENTU ZWIĄZANEGO.
8. BINAWIGOWALNOŚĆ ELEMENTU PĘKU – własność parametryzowalna względem elementu pęku, która określa, czy ELEMENT ZWIĄZANY posiada informacje o WIĄZKACH, w ramach których jest związany.
9. UPORZĄDKOWANIE ELEMENTU PĘKU – własność parametryzowalna względem elementu pęku, która określa, czy w jego ramach ELEMENTY ZWIĄZANE posiadają ustalony porządek.

Diagram pojęć na rys. 4.30 przedstawia konieczne atomy semantyczne wzorca pęku ról.

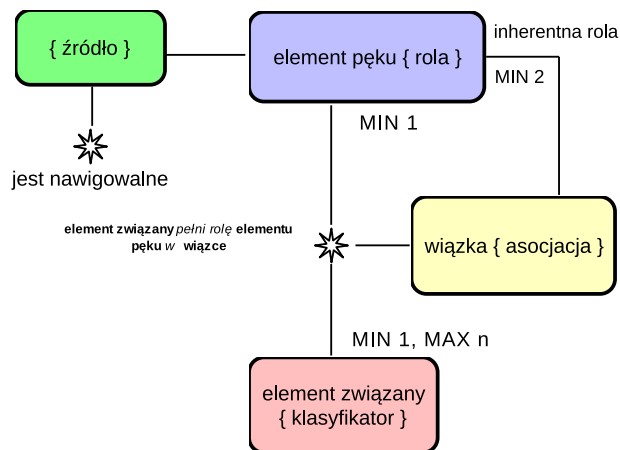
### Specyfikacja wzorca

*Konieczne atomy semantyczne*

K.1 pęk ról *jest konkretyzacją* koncepcji wzorca

K.2 pęk ról *zawsze posiada przynajmniej jeden oraz co najwyżej  $n$  elementów związanych, które są instancjami klasyfikatora*

K.3 pęk ról *zawsze posiada wiązkę, która jest konkretyzacją asocjacji*



Rysunek 4.30. Diagram pojęć prezentujący konieczne atomy semantyczne koncepcji wzorca pęku ról

K.3.1 element związany *zawsze pełni przynajmniej jedną rolę* elementu pęku w wiązce

K.3.1.1 źródło elementu pęku *jest nawigowalne*

K.3.2 wiązka *zawsze posiada dokładnie n inherentnych ról* oraz n *zawsze jest większe od 2*

*Atomy semantyczne własności możliwych*

M.1 opisanie wiązki *jest konkretyzacją własności wzorca*

M.1.1 opis wiązki *jest konkretyzacją typu bytu*

M.1.2 opis wiązki *pełni rolę opisywania wiązki w wiązce*

M.1.2.1 źródło opisywania wiązki *jest cechowane przez krotność, która posiada kres górny oraz kres dolny, który jest równy 1*

M.1.2.2 cel opisywania wiązki *jest cechowany przez krotność, która posiada kres górny oraz kres dolny, który jest równy 1*

M.1.2.3 źródło opisywania wiązki *jest nawigowalne*

M.1.2.4 cel opisywania wiązki *jest nawigowalny*

M.1.2.5 źródło opisywania wiązki *jest cechowane przez zależność czasu życia*

M.1.2.6 cel opisywania wiązki *jest cechowany przez zależność czasu życia*

M.1.3 opis wiązki *posiada przynajmniej jeden inherentny atrybut*

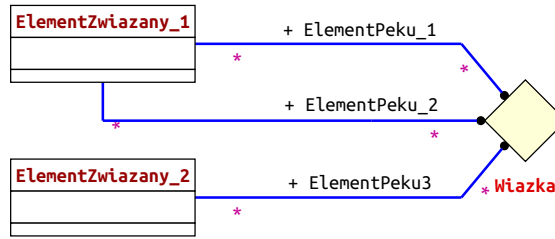
M.2 m-powtarzalność elementu pęku *p jest konkretyzacją własności wzorca*

- M.2.1  $p$  jest cechowana przez powtarzalność, która jest określana przez  $m$
- M.3  $m$ -minimalność elementu pęku  $p$  jest konkretyzacją własności wzorca
- M.3.1 cel  $p$  jest cechowany przez krotność, która posiada kres dolny, który jest równy  $m$
- M.4  $m$ -maksymalność elementu pęku  $p$  jest konkretyzacją własności wzorca
- M.4.1 cel  $p$  jest cechowany przez krotność, która posiada kres górny, który jest równy  $m$
- M.5  $m$ -ograniczona przynależność do wiązki elementu pęku  $p$  jest konkretyzacją własności wzorca
- M.5.1 źródło  $p$  jest cechowane przez krotność, która posiada kres górny, który jest równy  $m$
- M.6 zależność istnienia wiązki od elementu pęku  $p$  jest konkretyzacją własności wzorca
- M.6.1 cel  $p$  jest cechowany przez zależność istnienia
- M.7 zależność istnienia elementu pęku od wiązki  $p$  jest konkretyzacją własności wzorca
- M.7.1 źródło  $p$  jest cechowane przez zależność istnienia
- M.8 binawigowalność elementu pęku  $p$  jest konkretyzacją własności wzorca
- M.8.1 cel  $p$  jest nawigowalny
- M.9 uporządkowanie elementu pęku  $p$  jest konkretyzacją własności wzorca
- M.9.1  $p$  jest cechowany przez porządek

### **Implementacja rozwiązania**

*Implementacja koniecznych atomów semantycznych wzorca pęku ról*

Rysunek 4.31 przedstawia implementację koniecznych atomów semantycznych wzorca pęku ról. Poniżej przedstawiono odniesienie kolejnych elementów modelu do realizacji koniecznych atomów semantycznych wzorca.

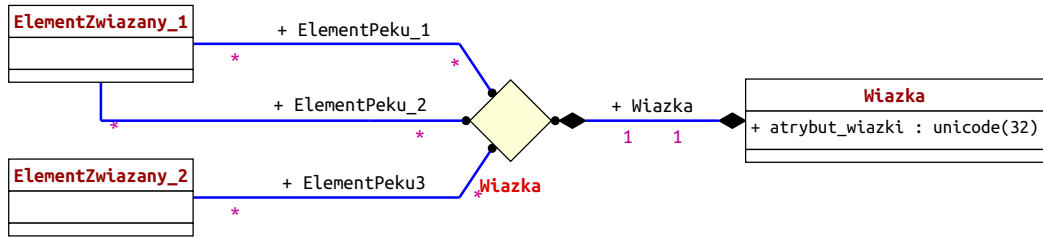


Rysunek 4.31. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne

$$\diamond Wiazka \left( \begin{array}{l} [*] \bullet \frac{+ElementPeku_1(\square ElementZwiazany_1)}{[*] \square ElementZwiazany_1} \\ [*] \bullet \frac{+ElementPeku_2(\square ElementZwiazany_1)}{[*] \square ElementZwiazany_1} \\ [*] \bullet \frac{+ElementPeku_3(\square ElementZwiazany_2)}{[*] \square ElementZwiazany_2} \end{array} \right); \quad (4.30)$$

1. Utworzenie modelu asocjacyjnego realizuje konieczny atom semantyczny K.1.
2. K.2 został zaimplementowany poprzez utworzenie kolekcji  $\square ElementZwiazany_1$  oraz  $\square ElementZwiazany_2$ . Kolekcje te reprezentują rozłączne w sensie typu elementy związane, wchodzące w skład pęku ról. Należy dodatkowo zauważyć, iż równie poprawna implementacja wzorca mogłaby zawierać dowolną liczbę elementów związanych.
3. K.3. został zrealizowany poprzez utworzenie asocjacji  $\diamond Wiazka$ .
4. K.3.1 został zaimplementowany poprzez utworzenie ról  $ElementPeku_1$ ,  $ElementPeku_2$  oraz  $ElementPeku_3$  w asocjacji  $\diamond Wiazka$ . Każda z tych ról reprezentuje pojęcie elementu pęku w specyfikacji wzorca.
5. K.3.1.1. został zrealizowany poprzez fakt, iż role w metamodelu asocjacyjnym są domyślnie dwukierunkowe.
6. K.3.2. realizuje fakt, iż asocjacja  $\diamond Wiazka$  posiada trzy role, których jest właścicielem. Liczba ta nie jest mniejsza od 2, ani od liczby kolekcji reprezentujących związane elementy.

Implementacja koniecznych atomów semantycznych wzorca pęku ról z własnością możliwą: [opisanie wiązki](#)



Rysunek 4.32. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz własność możliwą: opisanie wiązki

$$\diamond Wiazka \left( \begin{array}{l}
 \begin{array}{l}
 [*] \xrightarrow{+ElementPeku_1(\square ElementZwiazany_1)} [*] \square ElementZwiazany_1 \\
 [*] \xrightarrow{+ElementPeku_2(\square ElementZwiazany_1)} [*] \square ElementZwiazany_1 \\
 [*] \xrightarrow{+ElementPeku_3(\square ElementZwiazany_2)} [*] \square ElementZwiazany_2
 \end{array} \\
 \end{array} \right); \\
 [1] \xrightarrow{+Wiazka(\square Wiazka)} [1] \square Wiazka
 \tag{4.31}$$

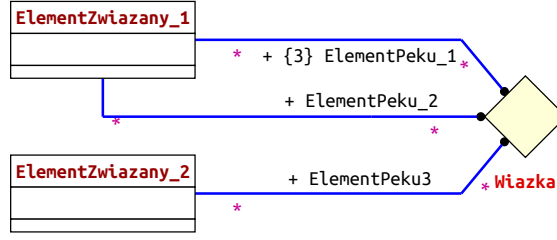
Implementacja wzorca pęku ról, która realizuje wszystkie konieczne atomy semantyczne tego wzorca (rys. 4.32) realizuje własność możliwą opisanie wiązki. Implementacja uwzględnia realizację wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomy semantyczne definiujące tę własność możliwą. Atom M.1. został spełniony poprzez utworzenie nowego modelu asocjacyjnego realizującego ten wzorec. Atom M.1.1. zrealizowano poprzez utworzenie kolekcji  $\square Wiazka$ , która pełni rolę opisu wiązki. Atomy semantyczne M.1.2.1-6 opisują wymagania dotyczące wzorca BACT. Dlatego, w implementacji założono wykorzystanie tej struktury, nadając *Wiazce* charakter binawigowalnego tandemu. Atom semantyczny M.1.3 zrealizowano tworząc w kolekcji  $\square Wiazka$  jeden atrybut o nazwie *+atrybut\_wiazki*.

*Implementacja koniecznych atomów semantycznych wzorca pęku ról z własnością możliwą: m-powtarzalność elementu pęku* dla  $m = 3$  oraz  $p =$

$$\diamond Wiazka \xrightarrow{\{3\}+ElementPeku_1} \square ElementZwiazany_1$$

$$\diamond Wiazka \left( \begin{array}{l}
 \begin{array}{l}
 [*] \xrightarrow{\{3\}+ElementPeku_1(\square ElementZwiazany_1)} [*] \square ElementZwiazany_1 \\
 [*] \xrightarrow{+ElementPeku_2(\square ElementZwiazany_1)} [*] \square ElementZwiazany_1 \\
 [*] \xrightarrow{+ElementPeku_3(\square ElementZwiazany_2)} [*] \square ElementZwiazany_2
 \end{array} \\
 \end{array} \right); \\
 \tag{4.32}$$

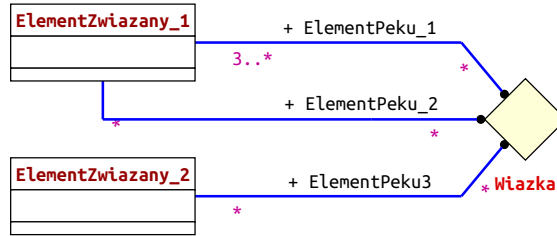
Na diagramie 4.33 pokazano model asocjacyjny wzorca pęku ról, który obrazuje własność możliwą 3-powtarzalność dla roli *ElementPeku\_1* w asocjacji  $\diamond Wiazka$ .



Rysunek 4.33. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz własność możliwą:  $m$ -powtarzalność elementu pęku dla  $m = 3$  oraz  $p = \diamond Wiazka \xrightarrow{\{3\}+ElementPeku_1} \square ElementZwiazany_1$

Przedstawiona implementacja uwzględnia realizację wszystkich koniecznych atomów semantycznych wzorca oraz atomy semantyczne definiujące tę własność możliwą. Atom M.2 został spełniony poprzez utworzenie modelu asocjacyjnego realizującego tę konfigurację wzorca. Atom M.2.1 został osiągnięty poprzez wyspecyfikowanie wartości unikalności równej 3 w ramach roli  $ElementPeku_1$  w asocjacji  $\diamond Wiazka$ .

Implementacja koniecznych atomów semantycznych wzorca pęku ról z własnością możliwą:  $m$ -minimalność elementu pęku dla  $m = 3$  oraz  $p = \diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$



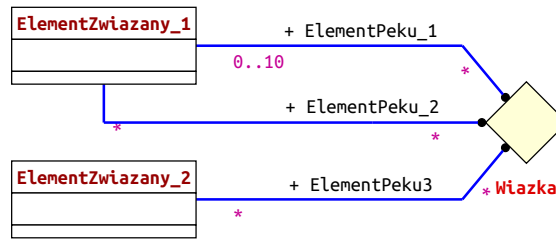
Rysunek 4.34. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz własność możliwą:  $m$ -minimalność elementu pęku dla  $m = 3$  oraz  $p = \diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$

$$\diamond Wiazka \left\{ \begin{array}{l} [*] \xrightarrow{+ElementPeku_1(\square ElementZwiazany_1)} [3..*] \square ElementZwiazany_1 \\ [*] \xrightarrow{+ElementPeku_2(\square ElementZwiazany_1)} [*] \square ElementZwiazany_1 \\ [*] \xrightarrow{+ElementPeku_3(\square ElementZwiazany_2)} [*] \square ElementZwiazany_2 \end{array} \right\}; \quad (4.33)$$

Implementację konfiguracji pęku ról realizującej własność możliwą  $m$ -minimalność elementu pęku pokazano na diagramie znajdującym się na rys. 4.34. Implementacja ta zakłada realizację wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomów semantycznych zdefiniowanych w ramach tej wła-

sności możliwej. Przyjęto  $m = 3$ , a pod  $p$  podstawiono rolę  $\diamond ElementPeku_1$  asocjacji  $\diamond Wiazka$ . Atom semantyczny M.3 został zrealizowany poprzez wyodrębnienie nowego modelu asocjacyjnego. Natomiast atom semantyczny M.3.1 zrealizowano poprzez określenie krotności roli  $ElementPeku_1$  asocjacji  $\diamond Wiazka$  po stronie elementu wiązanego jako  $3..*$ .

Implementacja koniecznych atomów semantycznych wzorca pęku ról z własnością możliwą:  $m$ -maksymalność elementu pęku dla  $m = 10$  oraz  $p = \diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$



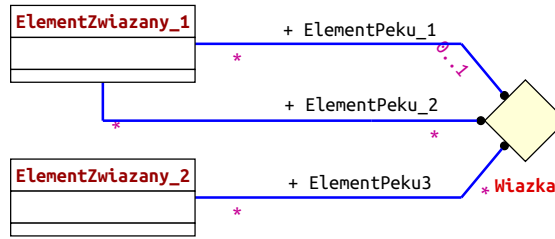
Rysunek 4.35. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz własność możliwą:  $m$ -maksymalność elementu pęku dla  $m = 10$  oraz  $p = \diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$

$$\diamond Wiazka \left\{ \begin{array}{l} [*] \xrightarrow{+ElementPeku_1(\square ElementZwiazany_1)} [0..10] \square ElementZwiazany_1 \\ [*] \xrightarrow{+ElementPeku_2(\square ElementZwiazany_1)} [*] \square ElementZwiazany_1 \\ [*] \xrightarrow{+ElementPeku_3(\square ElementZwiazany_2)} [*] \square ElementZwiazany_2 \end{array} \right\}; \quad (4.34)$$

Implementację konfiguracji pęku ról realizującej własność możliwą  $m$ -maksymalność elementu pęku pokazano na diagramie znajdującym się na rys. 4.35. Implementacja ta zakłada realizację wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomów semantycznych zdefiniowanych w ramach tej własności możliwej. Przyjęto za  $m = 10$ , a pod  $p$  podstawiono rolę  $\diamond ElementPeku_1$  asocjacji  $\diamond Wiazka$ . Atom semantyczny M.4 został zrealizowany poprzez wyodrębnienie nowego modelu asocjacyjnego. Natomiast atom semantyczny M.4.1 zrealizowano poprzez określenie krotności roli  $ElementPeku_1$  asocjacji  $\diamond Wiazka$  po stronie elementu wiązanego jako  $0..10$ .

Implementacja koniecznych atomów semantycznych wzorca pęku ról z własnością możliwą:  $m$ -ograniczoną przynależność do wiązki elementu pęku dla  $m = 1$  oraz  $p = \diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$



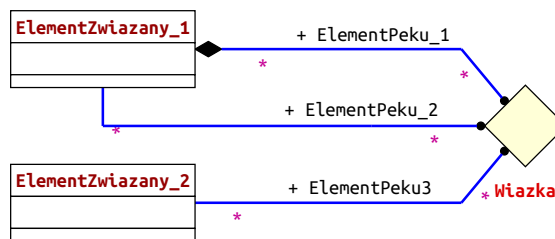


Rysunek 4.36. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz własność możliwą:  $m$ -ograniczoną przynależność do wiązki elementu pęku dla  $m = 1$  oraz  $p = \diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$

$$\diamond Wiazka \left( \begin{array}{l} [0..1] \xrightarrow{+ElementPeku_1(\square ElementZwiazany_1)} [*] \square ElementZwiazany_1 \\ [*] \xrightarrow{+ElementPeku_2(\square ElementZwiazany_1)} [*] \square ElementZwiazany_1 \\ [*] \xrightarrow{+ElementPeku_3(\square ElementZwiazany_2)} [*] \square ElementZwiazany_2 \end{array} \right); \quad (4.35)$$

Implementację konfiguracji pęku ról realizującej własność możliwą  $m$ -ograniczona przynależność do wiązki elementu pęku pokazano na diagramie znajdującym się na rys. 4.36. Implementacja ta zakłada realizację wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomów semantycznych określonych w ramach tej własności możliwej. Przyjęto za  $m = 1$ , a pod  $p$  podstawiono rolę  $\diamond ElementPeku_1$  asocjacji  $\diamond Wiazka$ . Atom semantyczny M.5 został zrealizowany poprzez wyodrębnienie nowego modelu asocjacyjnego. Natomiast atom semantyczny M.5.1 zrealizowano poprzez określenie krotności roli  $ElementPeku_1$  asocjacji  $\diamond Wiazka$  po stronie właściciela jako 0..1.

Implementacja koniecznych atomów semantycznych wzorca pęku ról z własnością możliwą: zależność istnienia wiązki od elementu pęku dla  $p = \diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$



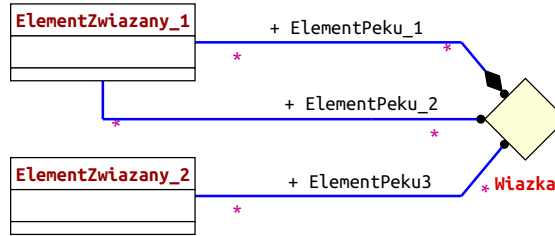
Rysunek 4.37. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz własność możliwą: zależność istnienia wiązki od elementu pęku dla  $p = \diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$

$$\diamond Wiazka \left\{ \begin{array}{l} [*] \xrightarrow{+ElementPeku_1(\square ElementZwiazany_1)} [*] \square ElementZwiazany_1 \\ [*] \xrightarrow{+ElementPeku_2(\square ElementZwiazany_1)} [*] \square ElementZwiazany_1 \\ [*] \xrightarrow{+ElementPeku_3(\square ElementZwiazany_2)} [*] \square ElementZwiazany_2 \end{array} \right\}; \quad (4.36)$$

Implementację konfiguracji pęku ról realizującej własność możliwą zależność istnienia wiązki od elementu pęku pokazano na diagramie znajdującym się na rys. 4.37. Implementacja ta zakłada realizację wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomów semantycznych określonych w ramach tej własności możliwej. Pod  $p$  podstawiono rolę  $\diamond ElementPeku_1$  asocjacji  $\diamond Wiazka$ . Atom semantyczny M.6 został zrealizowany poprzez wyodrębnienie nowego modelu asocjacyjnego. Natomiast atom semantyczny M.6.1 zrealizowano poprzez dodanie kompozycji roli  $ElementPeku_1$  asocjacji  $\diamond Wiazka$  po stronie elementu wiązanego.

Implementacja koniecznych atomów semantycznych wzorca pęku ról z własnością możliwą: zależność istnienia elementu pęku od wiązki dla  $p =$

$$\diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$$



Rysunek 4.38. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz własność możliwą: zależność istnienia od wiązki elementu

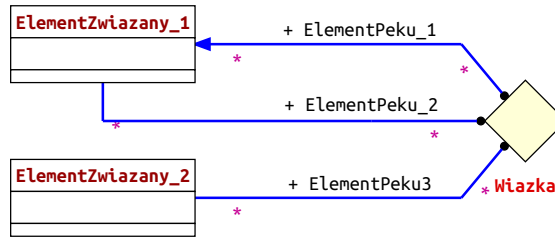
$$\text{pęku dla } p = \diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$$

$$\diamond Wiazka \left\{ \begin{array}{l} [*] \xrightarrow{+ElementPeku_1(\square ElementZwiazany_1)} [*] \square ElementZwiazany_1 \\ [*] \xrightarrow{+ElementPeku_2(\square ElementZwiazany_1)} [*] \square ElementZwiazany_1 \\ [*] \xrightarrow{+ElementPeku_3(\square ElementZwiazany_2)} [*] \square ElementZwiazany_2 \end{array} \right\}; \quad (4.37)$$

Implementację konfiguracji pęku ról realizującej własność możliwą zależność istnienia elementu pęku od wiązki pokazano na diagramie znajdującym się na rys. 4.38. Implementacja ta zakłada realizację wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomów semantycznych określonych

w ramach tej własności możliwej. Pod  $p$  podstawiono rolę  $\diamond ElementPeku_1$  asocjacji  $\diamond Wiazka$ . Atom semantyczny M.7 został zrealizowany poprzez wyodrębnienie nowego modelu asocjacyjnego, natomiast atom semantyczny M.7.1 zrealizowano poprzez dodanie kompozycji roli  $ElementPeku_1$  asocjacji  $\diamond Wiazka$  po stronie właściciela.

*Implementacja koniecznych atomów semantycznych wzorca pęku ról z własnością możliwą:* binawigowalność elementu pęku



Rysunek 4.39. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz zanegowaną własność możliwą binawigowalność elementu pęku dla roli  $\diamond Wiazka$

$$\diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$$

$$\diamond Wiazka \left\{ \begin{array}{l} [*] \xrightarrow{+ElementPeku_2(\square ElementZwiazany_1)} [*] \square ElementZwiazany_1 \\ [*] \xrightarrow{+ElementPeku_3(\square ElementZwiazany_2)} [*] \square ElementZwiazany_2 \\ [*] \xrightarrow{+ElementPeku_1(\square ElementZwiazany_1)} [*] \square ElementZwiazany_1 \end{array} \right\}; \quad (4.38)$$

Implementacja wzorca pęku ról, która realizuje wszystkie konieczne atomy semantyczne tego wzorca (rys. 4.31) realizuje własność możliwą binawigowalności elementu pęku dla każdej z ról. Dla metamodelu asocjacyjnego binawigowalność ról jest własnością domyślną. Należy jednak zauważyć, iż możliwe jest zanegowanie tej własności w kierunku źródło  $\rightarrow$  cel poprzez utworzenie roli uninawigowalnej – implementację taką przedstawiono dla uninawigowalnej roli  $ElementPeku_1$  w asocjacji  $\diamond Wiazka$  (rys. 4.39).

*Implementacja koniecznych atomów semantycznych wzorca pęku ról z własnością możliwą:* uporządkowanie elementu pęku

Implementacja wzorca pęku ról przedstawiona na rys. 4.31 spełnia semantykę własności możliwej uporządkowanie elementu pęku dla każdej z ról, ponieważ realizuje atomy semantyczne M.9 oraz M.9.1.

## Przykład zastosowania

Przykładem zastosowania wzorca pęku ról może być model zajęć na uczelni wyższej, w ramach którego można wyróżnić szereg następujących ELEMENTÓZWIAZANYCH: prowadzący zajęcia, grupa studentów, przedmiot, sala dydaktyczna oraz termin w harmonogramie tygodniowym. Same zajęcia pełnią rolę WIĄZKI. Na rys. 4.40 pokazano diagram AML reprezentujący model zajęć na uczelni wyższej zrealizowany jako wzorec pęku ról. Implementacja wzorca obejmuje własności możliwe:

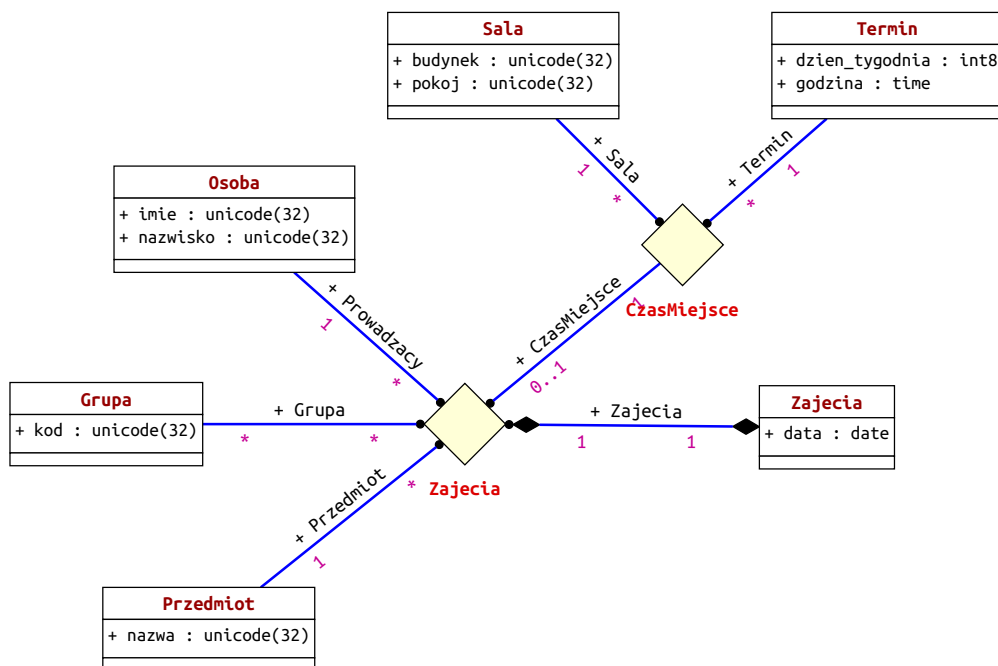
- opisanie wiązki,
- 1-maksymalność elementu pęku dla ról  $\diamond Zajecia \xrightarrow{+Przedmiot} \square Przedmiot$  oraz  $\diamond Zajecia \xrightarrow{+Prowadzacy} \square Osoba$ ,
- 1-minimalność elementu pęku dla ról  $\diamond Zajecia \xrightarrow{+Przedmiot} \square Przedmiot$  oraz  $\diamond Zajecia \xrightarrow{+Prowadzacy} \square Osoba$ ,
- 1-ograniczoną przynależność do wiązki elementu pęku dla roli  $\diamond Zajecia \xrightarrow{+CzasMiejsce} \diamond CzasMiejsce$ ,
- uporządkowanie elementu pęku dla każdej z ról,
- binawigowalność elementu pęku dla każdej z ról.

Warto zwrócić na zastosowany w modelu zabieg, mający na celu redukcję elementów pęku. Jednym z uczestników asocjacji  $\diamond Zajecia$  jest asocjacja:  $\diamond CzasMiejsce$ . Asocjacja  $\diamond CzasMiejsce$  reprezentuje związek konkretnego miejsca (sali) oraz konkretnego terminu (określonego jako dzień tygodnia i godzina). Obiekty asocjacji  $\diamond CzasMiejsce$  reprezentują określone punkty w cyklicznej czasoprzestrzeni, w których odbywają się zajęcia – stąd też dopiero klasyfikator tych punktów uczestniczy w asocjacji  $\diamond Zajecia$ .

### 4.1.5. Sieć uogólniona

#### Opis problemu

Problem rozwiązywany przez wzorec sieci uogólnionej polega na możliwości utworzenia modeli opartych o różnego rodzaju konstrukcje, które polegają na uogólnieniu grafowego podejścia do reprezentacji danych. Koncepcja wzorca sieci uogólnionej bazuje na zastosowaniu szeregu własności modelowania grafowego zauważonych w [79] do problematyki modelowania konceptualnego. Przyczyną wyodrębnienia sieci uogólnionej jako osobnego wzorca jest przeniesienie struktury danych przechowywanej w ramach instancji tego wzorca na wyższy poziom abstrakcji. Przyjęta abstrakcja polega na uogólnieniu krawędzi oraz węzła do postaci elementu wiązki.



Rysunek 4.40. Przykład użycia wzorca pęku ról

Zagadnienie modelowania struktur o charakterze sieciowym, takich jak grafy i hipergrafy [14, 148] oraz ubergrafy [82] wykorzystywane jest do reprezentacji szeregu problemów, struktur i zagadnień w nauce i jej zastosowaniach [105, 123, 151].

### Elementy składowe

Pojęcia stanowiące uczestników wzorca:

1. ELEMENT WIĄZANY – pojęcie stanowiące abstrakcję uogólnienia pojęć WĘZŁA oraz KRAWĘDZI
2. WĘZŁEŁ – pojęcie stanowiące abstrakcję elementu reprezentującego dane,
3. KRAWĘDŹ – pojęcie stanowiące abstrakcję elementu reprezentującego związku pomiędzy danymi.

Pojęcie stanowiące powiązanie pomiędzy uczestnikami:

1. ZŁĄCZE – pojęcie pełniące rolę związku łączącego element związany z wiążącym.

### Własności

Pojęcia stanowiące własności możliwe wzorca *sieci*:

1. BINARNOŚĆ POWIĄZAŃ – własność wzorca, która ogranicza możliwość realizowania struktur hipergrafowych do struktur uwzględniających jedynie binarne krawędzie. Innymi słowy, jedna krawędź może łączyć tylko dwa węzły.

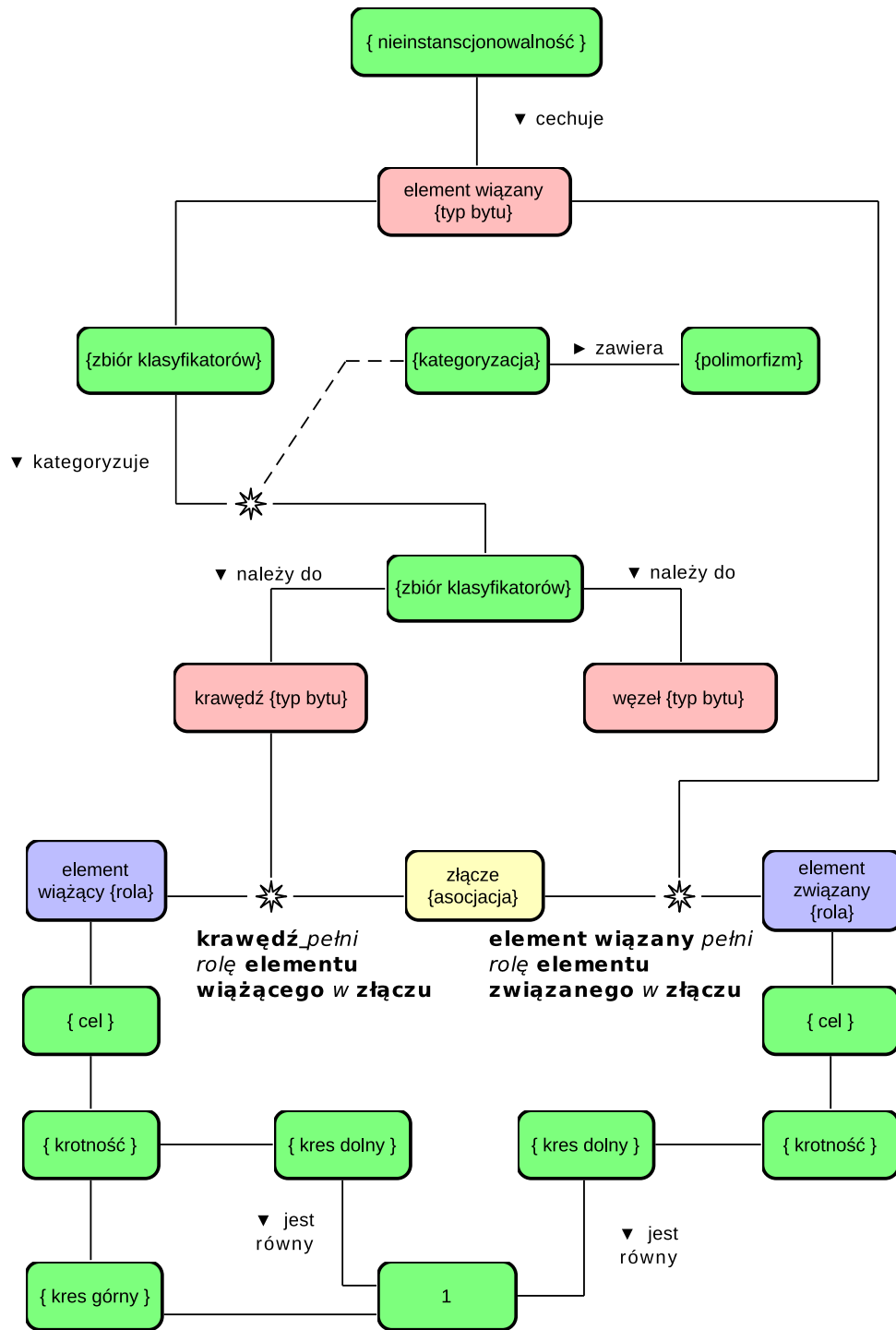
2. WĘZŁOWE OGRANICZENIE ZŁĄCZA – własność wzorca, która ogranicza typ elementu związanego do węzła, ograniczając możliwość realizacji struktur ultragrafowych.
3. ETYKIETOWALNOŚĆ WĘZŁÓW – własność wzorca określająca, czy jest możliwe opisanie węzła dodatkowym zbiorem atrybutów.
4. ETYKIETOWALNOŚĆ KRAWĘDZI – własność wzorca określająca, czy jest możliwe opisanie krawędzi dodatkowym zbiorem atrybutów.
5. PIERWSZOKLASOWOŚĆ – własność wzorca określająca, czy sieci jako takie mogą być elementami krawędzi. Własność sprawia, iż konceptualizowane sieci będą zgodne z ideą sieci wydzielonych [15, 67].
6. UNINAWIGOWALNOŚĆ – własność wzorca określająca, że każda KRAWĘDŹ posiada zdefiniowany kierunek przechodzenia pomiędzy ELEMENTAMI WIĄZANYMI.

### Specyfikacja wzorca

Diagram pojęć na rys. 4.41 przedstawia konieczne atomy semantyczne wzorca sieci uogólnionej.

*Konieczne atomy semantyczne*

- K.1 sieć uogólniona jest konkretyzacją koncepcji wzorca
- K.2 sieć uogólniona zawsze posiada węzeł, który jest konkretyzacją typu bytu
- K.3 sieć uogólniona zawsze posiada krawędź, która jest konkretyzacją typu bytu
- K.4 sieć uogólniona zawsze posiada element wiązany, który jest konkretyzacją typu bytu
- K.5 element wiązany kategoryzuje zbiór klasyfikatorów, którego elementami są węzeł oraz krawędź
  - K.5.1 kategoryzacja ta zawiera polimorfizm
- K.6 sieć uogólniona zawsze posiada złącze, które jest konkretyzacją asocjacji
  - K.6.1 krawędź zawsze pełni rolę elementu wiążącego w złączu
  - K.6.2 element wiązany zawsze pełni rolę elementu związanego w złączu
  - K.6.3 cel elementu wiążącego jest cechowany przez krotność, która posiada kres górny oraz kres dolny, które są równe 1
- K.7 element wiązany jest cechowany przez nieinstancjonowalność



Rysunek 4.41. Diagram pojęć prezentujący konieczne atomy semantyczne koncepcji wzorca sieci uogólnionej

*Atomy semantyczne własności możliwych*

M.1 binarność powiązań sieci uogólnionej jest konkretyzacją własności wzorca

M.1.1 każda krawędź pełni rolę elementu wiążącego w dokładnie 2 złączach

M.2 węzłowe ograniczenie złącza jest konkretyzacją własności wzorca

- M.2.1 każda instancja klasyfikowana przez element wiązany pełniący rolę element związanego w złączu jest klasyfikowana przez węzeł
- M.3 etykietowalność węzłów sieci uogólnionej jest konkretyzacją własności wzorca
- M.3.1 węzeł posiada przynajmniej jeden inherentny atrybut
- M.4 etykietowalność krawędzi sieci uogólnionej jest konkretyzacją własności wzorca
- M.4.1 krawędź posiada przynajmniej jeden inherentny atrybut
- M.5 pierwszoklasowość sieci uogólnionej jest konkretyzacją własności wzorca
- M.5.1 sieć uogólniona posiada sieć, która jest konkretyzacją typu bytu
- M.5.1.1 sieć należy do zbioru klasyfikatorów, który jest kategoryzowany przez element wiązany
- M.5.2 sieć uogólniona posiada posiadanie elementów przez sieć, która jest konkretyzacją asocjacji
- M.5.2.1 sieć pełni rolę właściciela w posiadaniu elementów przez sieć
- M.5.2.1.1 źródło właściciela jest cechowane przez zależność czasu istnienia
- M.5.2.1.2 źródło i cel właściciela są cechowane przez krotność, która posiada kres górny oraz kres dolny, które są równe 1
- M.5.2.2 element wiązany pełni rolę własności w posiadaniu elementów przez sieć
- M.5.2.2.1 cel własności jest cechowany przez zależność czasu istnienia
- M.5.2.2.2 źródło własności jest cechowane przez krotność, która posiada kres górny oraz kres dolny, które są równe 1
- M.6 uninawigowalność krawędzi sieci uogólnionej jest konkretyzacją własności wzorca
- M.6.1 każde złącze cechowane jest przez nawigowalność
- M.6.2 jeżeli źródło roli posiadanej przez złącze jest nawigowalne, to cel tej roli jest nienawigowalny
- M.6.3 jeżeli źródło roli posiadanej przez złącze jest nienawigowalne, to cel tej roli jest nawigowalny



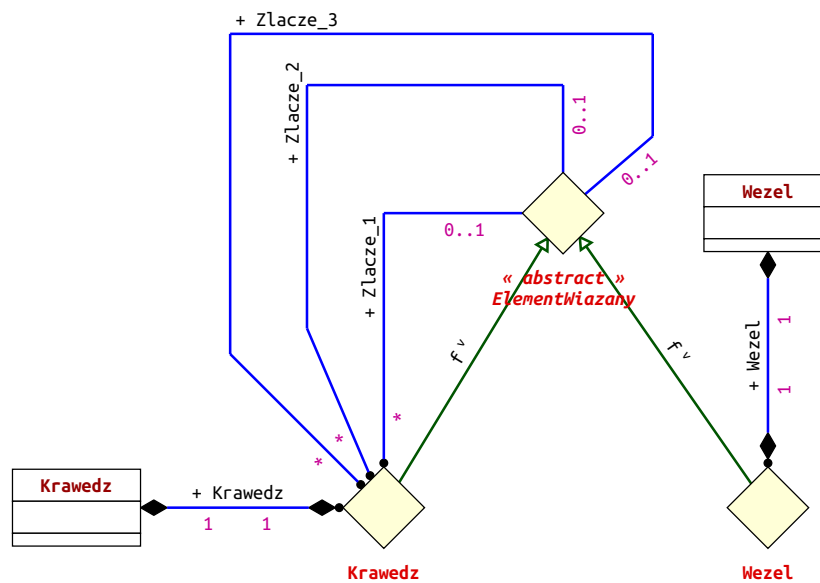
M.6.4 każda krawędź w przynajmniej jednym złączeniu pełni rolę elementu wiążącego, którego źródło jest nawigowalne oraz w przynajmniej jednym złączeniu pełni rolę elementu wiążącego, którego źródło jest nienawigowalne

### Implementacja rozwiązania

Implementacja koniecznych atomów semantycznych sieci uogólnionej

Na rysunkach 4.42 oraz 4.43 przedstawiono implementację koniecznych atomów semantycznych wzorca sieci uogólnionej. Rozwiązanie przedstawione na rys. 4.42 dotyczy sposobu implementacji opartego o stałą, maksymalną arność powiązań. W tym przykładzie złącze zostało zrealizowane przy pomocy asocjacyjnego pojęcia roli. Ze względu na konieczność przedstawienia konkretnej struktury, przykład przedstawia trzy konkretne złącza:  $Zlacze_1$ ,  $Zlacze_2$ ,  $Zlacze_3$ . Oznacza to, że struktury opisane przez ten model będą mogły wykorzystywać maksymalnie trzy złącza.

W przypadku rozwiązania polegającego na zastosowaniu asocjacji<sup>4</sup> możliwe jest dynamiczne tworzenie nowych złączy (konkretne złącza zostały przeniesione na poziom ekstensjonalny).



Rysunek 4.42. Diagram modelu asocjacyjnego wzorca sieci uogólnionej implementującego konieczne atomy semantyczne – wersja uwzględniająca stałą, maksymalną arność powiązań

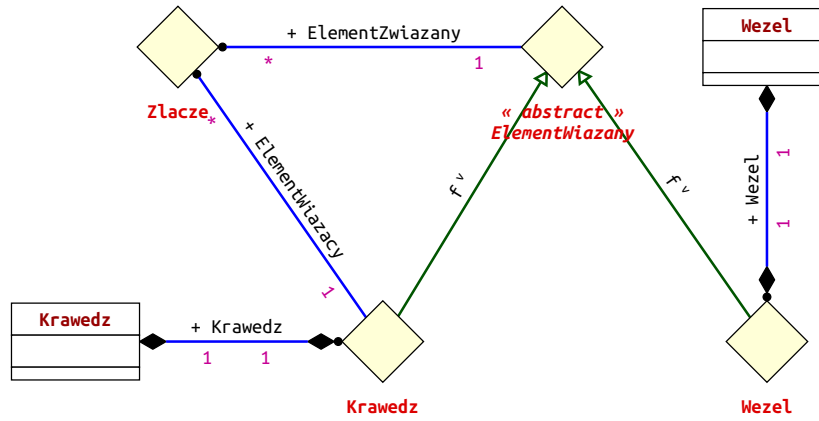
$$\diamond Krawedz \left\{ \begin{array}{l} [1] \xrightarrow{+Krawedz(\square Krawedz)} [1] \square Krawedz \\ [*] \xrightarrow{+Zlacze_1} [0..1] \diamond ElementWiazany^\emptyset \\ [*] \xrightarrow{+Zlacze_2} [0..1] \diamond ElementWiazany^\emptyset \\ [*] \xrightarrow{+Zlacze_3} [0..1] \diamond ElementWiazany^\emptyset \end{array} \right\}; \quad (4.39)$$

<sup>4</sup> Pojęcie modelu asocjacyjnego – Assoc

$$\diamond Wezel [1] \xrightarrow{+Wezel(\square Wezel)} [1] \square Wezel \quad (4.40)$$

$$\diamond ElementWiazany^\emptyset \quad (4.41)$$

$$\left\{ \begin{array}{l} \diamond Krawedz \\ \diamond Wezel \end{array} \right\} \xrightarrow{f^v} \diamond ElementWiazany^\emptyset \quad (4.42)$$



Rysunek 4.43. Diagram modelu asocjacyjnego wzorca sieci uogólnionej implementującego konieczne atomy semantyczne – wersja uwzględniająca dowolną arność powiązań

$$\diamond Krawedz [1] \xrightarrow{+Krawedz(\square Krawedz)} [1] \square Krawedz \quad (4.43)$$

$$\diamond Wezel [1] \xrightarrow{+Wezel(\square Wezel)} [1] \square Wezel \quad (4.44)$$

$$\diamond ElementWiazany^\emptyset \quad (4.45)$$

$$\left\{ \begin{array}{l} \diamond Krawedz \\ \diamond Wezel \end{array} \right\} \xrightarrow{f^v} \diamond ElementWiazany^\emptyset \quad (4.46)$$

$$\diamond Zlacze \left\{ \begin{array}{l} [*] \xrightarrow{+ElementWiazacy} [1] \diamond Krawedz \\ [*] \xrightarrow{+ElementZwiazany} [1] \diamond ElementWiazany^\emptyset \end{array} \right\}; \quad (4.47)$$

Poniżej przedstawiono odniesienie kolejnych elementów modelu do realizacji koniecznych atomów semantycznych wzorca, które są wspólne dla obu implementacji.

1. Implementacja K.1 została zrealizowana poprzez utworzenie odrębnego modelu asocjacyjnego implementującego ten wzorzec.

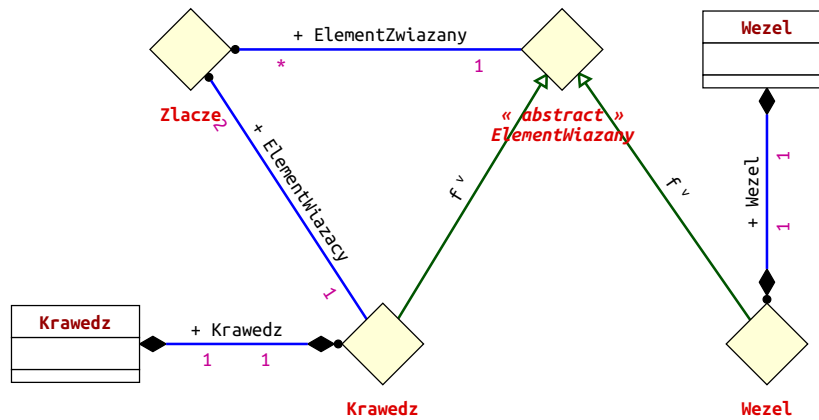
2. Implementacja K.2 została zrealizowana poprzez zainstancjonowanie wzorca BACT (4.2.1) w postaci kolekcji  $\square$ *Wezel* powiązanego rolą *Wezel* z asocjacją  $\diamond$ *Wezel* bikompozycyjną rolą o jednostkowych krotnościach.
3. Implementacja K.3 została zrealizowana analogicznie — poprzez wykorzystanie wzorca BACT.
4. Implementacja K.4 została zrealizowana poprzez utworzenie asocjacji  $\diamond$ *ElementWiazany*.
5. Implementacja K.5 została zrealizowana poprzez utworzenie związków dziedziczenia pomiędzy asocjacją  $\diamond$ *ElementWiazany* a asocjacją  $\diamond$ *Krawedz* oraz  $\diamond$ *Wezel*. W związkach tych  $\diamond$ *ElementWiazany* stanowi generalizację.
6. Implementacja atomu semantycznego K.5.1 została zrealizowana poprzez wyspecyfikowanie trybu dziedziczenia jako  $f^v$ , to znaczy włączenia dziedziczenia praw do pełnienia ról.
7. Implementacja atomu semantycznego K.7 została zrealizowana poprzez określenie asocjacji  $\diamond$ *ElementWiazany* jako abstrakcyjnej.

W kolejnych punktach przedstawiono odmienne sposoby rozwiązania w obu implementacjach.

W przypadku modelu z diagramu 4.42 kolejne konieczne atomy semantyczne zostały opatrzone symbolem a), natomiast dla modelu przedstawionego na diagramie 4.43 jest to symbol b).

7. a) Implementacja atomu K.6 została zrealizowana poprzez utworzenie zestawu ról *Zlacze<sub>1</sub>*, *Zlacze<sub>2</sub>*, *Zlacze<sub>3</sub>* w asocjacji  $\diamond$ *Krawedz*.  
b) Implementacja atomu K.6 polegało na utworzeniu asocjacji  $\diamond$ *Zlacze*.
8. a) Implementacja atomu K.6.1 polegała na określeniu właściciela ról *Zlacze<sub>1</sub>*, *Zlacze<sub>2</sub>*, *Zlacze<sub>3</sub>* jako asocjację  $\diamond$ *Krawedz*.  
b) Implementacja atomu K.6.1 polegała na utworzeniu roli *ElementWiazacy* w asocjacji  $\diamond$ *Zlacze* i określeniu jej celu jako  $\diamond$ *Krawedz*.
9. a) Implementacja K.6.2 została zrealizowana poprzez określenie asocjacji  $\diamond$ *ElementWiazany* jako celu dla zestawu ról *Zlacze<sub>1</sub>*, *Zlacze<sub>2</sub>*, *Zlacze<sub>3</sub>*.  
b) Implementacja K.6.2 została zrealizowana poprzez utworzenie roli *ElementZwiazany* w asocjacji  $\diamond$ *Zlacze* i określeniu jej celu jako  $\diamond$ *Krawedz*.
10. a) Implementacja K.6.3 jest niemożliwa do bezpośredniej i pełnej realizacji przy założonej strukturze.  
b) Implementacja K.6.3 została zrealizowana poprzez określenie krotności roli *ElementWiazacy* w asocjacji  $\diamond$ *Zlacze* po stronie elementu wiązanego jako 1.

Implementacja koniecznych atomów semantycznych wzorca sieci uogólnionej z własnością możliwą: binarność powiązań sieci uogólnionej



Rysunek 4.44. Diagram modelu asocjacyjnego wzorca sieci uogólnionej implementującego konieczne atomy semantyczne oraz własność możliwą: binarność powiązań sieci uogólnionej

$$\diamond Krawedz [1] \xrightarrow{+Krawedz(\square Krawedz)} [1] \square Krawedz \quad (4.48)$$

$$\diamond Wezel [1] \xrightarrow{+Wezel(\square Wezel)} [1] \square Wezel \quad (4.49)$$

$$\diamond ElementWiazany^\emptyset \quad (4.50)$$

$$\left\{ \begin{array}{l} \diamond Krawedz \\ \diamond Wezel \end{array} \right\} \xrightarrow{f^v} \diamond ElementWiazany^\emptyset \quad (4.51)$$

$$\diamond Zlacze \left\{ \begin{array}{l} [2] \xrightarrow{+ElementWiazacy} [1] \diamond Krawedz \\ [*] \xrightarrow{+ElementZwiazany} [1] \diamond ElementWiazany^\emptyset \end{array} \right\}; \quad (4.52)$$

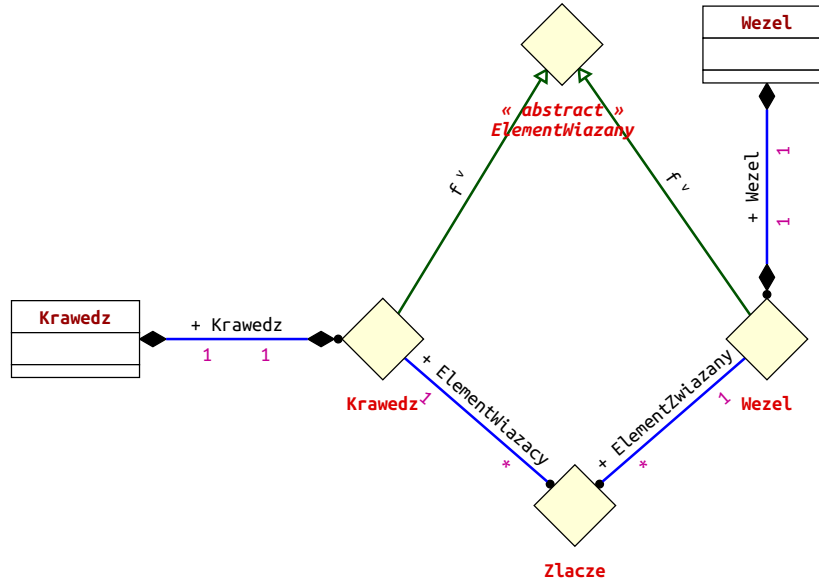
$$\square Wezel \langle \rangle; \quad (4.53)$$

$$\square Krawedz \langle \rangle; \quad (4.54)$$

Implementację konfiguracji wzorca sieci uogólnionej realizującej własność możliwą binarność powiązań sieci uogólnionej przedstawiono na diagramie AML znajdującym się na rys. 4.44 oraz w postaci wyrażeń AFN 4.48-4.54. Implementacja ta zakłada realizację wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomów semantycznych określonych w ramach tej własności możliwej.

Atomy semantyczne M.1 oraz M.1.1 zostały zaimplementowane poprzez określenie w nowoutworzonym modelu krotności roli *ElementWiazacy* w asocjacji  $\diamond$ Zlacze po stronie właściciela roli jako 2.

Implementacja koniecznych atomów semantycznych wzorca sieci uogólnionej z własnością możliwą: węzłowe ograniczenie złącza



Rysunek 4.45. Diagram modelu asocjacyjnego wzorca sieci uogólnionej implementującego konieczne atomy semantyczne oraz własność możliwą: węzłowe ograniczenie złącza

$$\diamond Krawedz [1] \xrightarrow{+Krawedz(\square Krawedz)} [1] \square Krawedz \quad (4.55)$$

$$\diamond Wezel [1] \xrightarrow{+Wezel(\square Wezel)} [1] \square Wezel \quad (4.56)$$

$$\diamond ElementWiazany^\emptyset \quad (4.57)$$

$$\left\{ \begin{array}{l} \diamond Krawedz \\ \diamond Wezel \end{array} \right\} \xrightarrow{f^v} \diamond ElementWiazany^\emptyset \quad (4.58)$$

$$\diamond Zlacze \left\langle \begin{array}{l} [*] \xrightarrow{+ElementWiazacy(\diamond Krawedz)} [1] \diamond Krawedz \\ [*] \xrightarrow{+ElementZwiazany(\diamond Wezel)} [1] \diamond Wezel \end{array} \right\rangle; \quad (4.59)$$

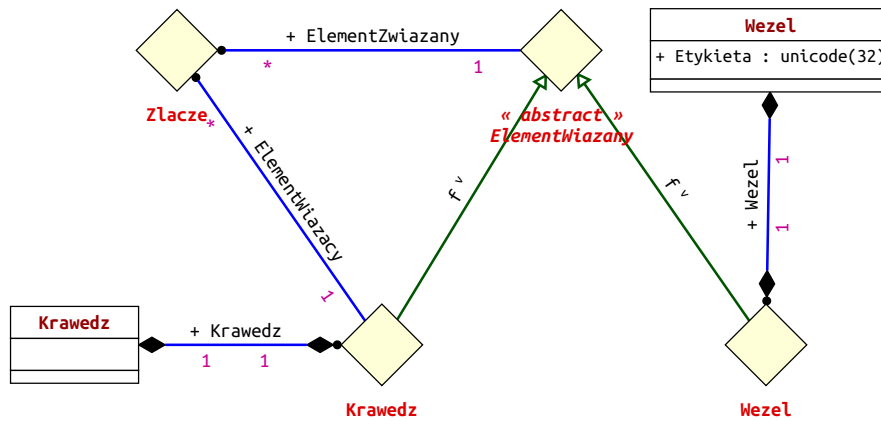
$$\square Wezel \langle \rangle; \quad (4.60)$$

$$\square Krawedz \langle \rangle; \quad (4.61)$$

Implementację konfiguracji wzorca sieci uogólnionej realizującej własność możliwą węzłowe ograniczenie złącza przedstawiono na diagramie AML znajdującym się na rys. 4.45 oraz w postaci wyrażeń AFN 4.55-4.61. Implementacja ta zakłada realizację wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomów semantycznych określonych w ramach tej własności możliwej.

Atomy semantyczne M.2 oraz M.2.1 zostały zaimplementowane poprzez utworzenie odrębnego modelu oraz usunięcie roli *ElementZwiazany* z asocjacji  $\diamond Zlacze$  oraz utworzenie nowej roli w tej samej asocjacji o tej samej nazwie i takiej samej charakterystyce, jednakże określając cel roli jako  $\diamond Wezel$ .

Implementacja koniecznych atomów semantycznych wzorca sieci uogólnionej z własnością możliwą: etykietowalność węzłów sieci uogólnionej



Rysunek 4.46. Diagram modelu asocjacyjnego wzorca sieci uogólnionej implementującego konieczne atomy semantyczne oraz własność możliwą: etykietowalność węzłów sieci uogólnionej

$$\diamond Krawedz [1] \xrightarrow{+Krawedz(\square Krawedz)} [1] \square Krawedz \quad (4.62)$$

$$\diamond Wezel [1] \xrightarrow{+Wezel(\square Wezel)} [1] \square Wezel \quad (4.63)$$

$$\diamond ElementWiazany \quad (4.64)$$

$$\left\{ \begin{array}{l} \diamond Krawedz \\ \diamond Wezel \end{array} \right\} \xrightarrow{f^v} \diamond ElementWiazany^\emptyset \quad (4.65)$$

$$\diamond Zlacze \left\langle \begin{array}{l} [*] \xrightarrow{+ElementWiazacy} [1] \diamond Krawedz \\ [*] \xrightarrow{+ElementZwiazany} [1] \diamond ElementWiazany \end{array} \right\rangle; \quad (4.66)$$

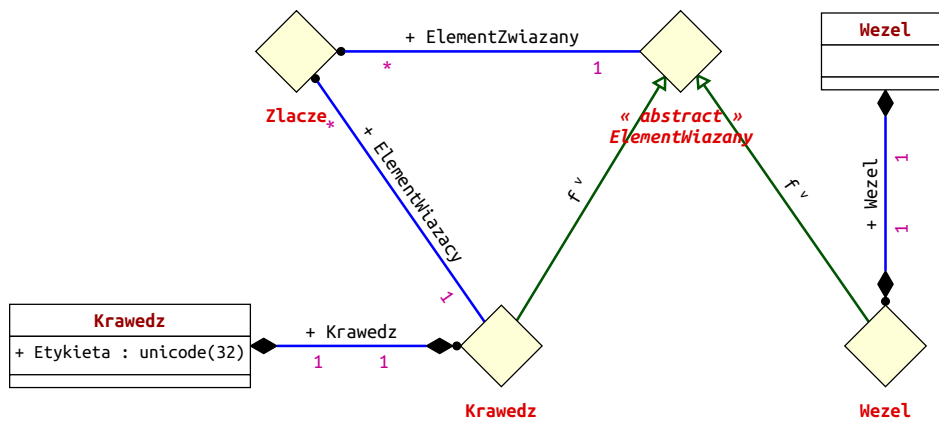
$$\square Wezel \left\langle +Etykieta : unicode(1) \right\rangle; \quad (4.67)$$

$$\square Krawedz \langle \rangle; \quad (4.68)$$

Implementację konfiguracji wzorca sieci uogólnionej realizującej własność możliwą etykietowalność węzłów sieci uogólnionej przedstawiono na diagramie AML znajdującym się na rys. 4.46 oraz w postaci wyrażeń AFN 4.62-4.68. Implementacja ta zakłada realizację wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomów semantycznych określonych w ramach tej własności możliwej.

Atomy semantyczne M.3 oraz M.3.1 zostały zaimplementowane poprzez dodanie atrybutu *Etykieta : unicode(32)* do kolekcji  $\square Wezel$  w ramach nowego modelu, reprezentującego konfigurację wzorca.

Implementacja koniecznych atomów semantycznych wzorca sieci uogólnionej z własnością możliwą: etykietowalność krawędzi sieci uogólnionej



Rysunek 4.47. Diagram modelu asocjacyjnego wzorca sieci uogólnionej implementującego konieczne atomy semantyczne oraz własność możliwą: etykietowalność krawędzi sieci uogólnionej

$$\diamond Krawedz [1] \xrightarrow{+Krawedz(\square Krawedz)} [1] \square Krawedz \quad (4.69)$$

$$\diamond Wezel [1] \xrightarrow{+Wezel(\square Wezel)} [1] \square Wezel \quad (4.70)$$

$$\diamond ElementWiazany^{\emptyset} \quad (4.71)$$

$$\left\{ \begin{array}{l} \diamond Krawedz \\ \diamond Wezel \end{array} \right\} \xrightarrow{f^v} \diamond ElementWiazany^{\emptyset} \quad (4.72)$$

$$\diamond Zlacze \left\langle \begin{array}{l} [*] \xrightarrow{+ElementWiazacy} [1] \diamond Krawedz \\ [*] \xrightarrow{+ElementZwiazany} [1] \diamond ElementWiazany^{\emptyset} \end{array} \right\rangle; \quad (4.73)$$

$$\square Wezel \langle \rangle; \quad (4.74)$$

$$\square Krawedz \langle +Etykieta : unicode(32) \rangle; \quad (4.75)$$

Implementację konfiguracji wzorca sieci uogólnionej realizującej własność możliwą etykietowalność krawędzi sieci uogólnionej przedstawiono na diagramie AML znajdującym się na rys. 4.47 oraz w postaci wyrażeń AFN 4.69-4.75. Implementacja ta zakłada realizację wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomów semantycznych określonych w ramach tej własności możliwej.

Atomy semantyczne M.4 oraz M.4.1 zostały zaimplementowane poprzez dodanie atrybutu *Etykieta : unicode(32)* do kolekcji  $\square Krawedz$  w ramach nowego modelu, reprezentującego konfigurację wzorca.

*Implementacja koniecznych atomów semantycznych wzorca sieci uogólnionej z własnością możliwą: pierwszoklasowość sieci uogólnionej*

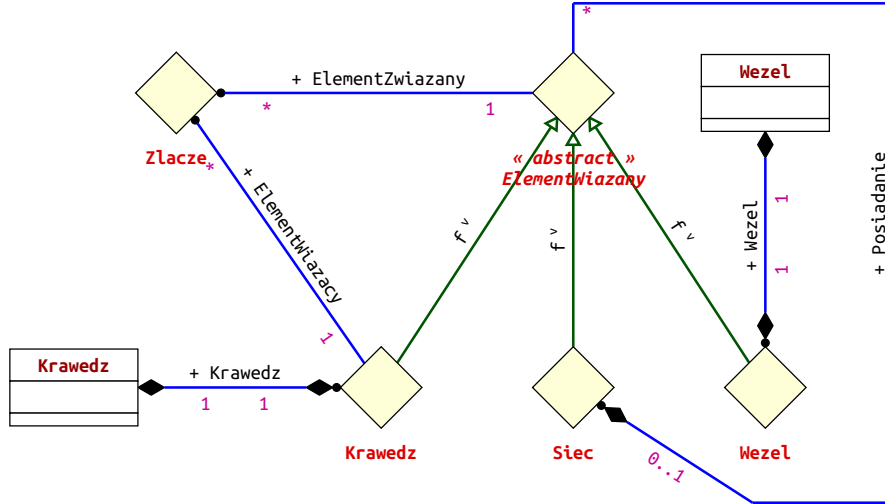
$$\diamond Krawedz [1] \xrightarrow{+Krawedz} [1] \square Krawedz \quad (4.76)$$

$$\diamond Wezel [1] \xrightarrow{+Wezel} [1] \square Wezel \quad (4.77)$$

$$\diamond ElementWiazany \quad (4.78)$$

$$\left\{ \begin{array}{l} \diamond Krawedz \\ \diamond Wezel \\ \diamond Siec \end{array} \right\} \xrightarrow{f^v} \diamond ElementWiazany^{\emptyset} \quad (4.79)$$





Rysunek 4.48. Diagram modelu asocjacyjnego wzorca sieci uogólnionej implementującego konieczne atomy semantyczne oraz własność możliwą: pierwszoklasowość sieci uogólnionej

$$\diamond Zlacze \left\langle \begin{array}{l} [*] \xrightarrow{+ElementWiazacy} [1] \diamond Krawedz \\ [*] \xrightarrow{+ElementZwiazany} [1] \diamond ElementWiazany \end{array} \right\rangle; \quad (4.80)$$

$$\diamond Siec [0..1] \xrightarrow{+Posiadanie} [*] \diamond ElementWiazany \quad (4.81)$$

$$\square Wezel \langle \rangle; \quad (4.82)$$

$$\square Krawedz \langle \rangle; \quad (4.83)$$

Implementację konfiguracji wzorca sieci uogólnionej realizującej własność możliwą pierwszoklasowość sieci uogólnionej przedstawiono na diagramie AML znajdującym się na rys. 4.48 oraz w postaci wyrażeń AFN 4.76-4.83. Implementacja ta zakłada realizację wszystkich koniecznych atomów semantycznych wzorca oraz dodatkowo atomów semantycznych określonych w ramach tej własności możliwej.

Atom semantyczny M.5 został zrealizowany poprzez utworzenie odrębnego modelu asocjacyjnego. Atom M.5.1 zrealizowano poprzez utworzenie asocjacji  $\diamond Siec$ . Atom M.5.1.1 został zrealizowany poprzez utworzenie generalizacji pomiędzy asocjacją  $\diamond Siec$  oraz  $\diamond ElementWiazany^\emptyset$ . Atomy semantyczny M.5.2, M.5.2.1 i M.5.2.1-2 zrealizowano poprzez utworzenie roli  $+Posiadanie$  w asocjacji  $\diamond Siec$  o krotnościach 0..1 po stronie właściciela oraz \* po stronie elementu związanego, celem roli jest asocjacja  $\diamond ElementWiazany^\emptyset$ .

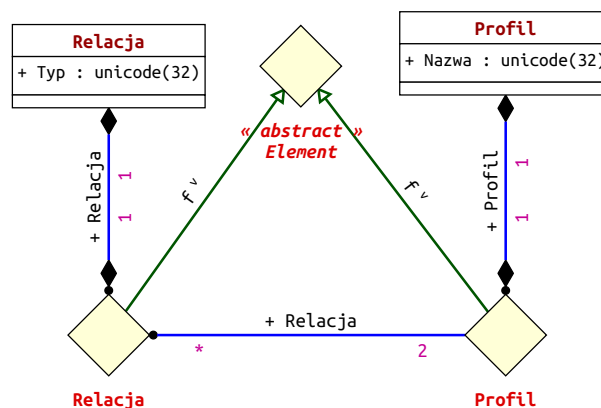
Implementacja koniecznych atomów semantycznych wzorca sieci uogólnionej z własnością możliwą: uninawigowalność

Implementacja własności możliwej uninawigowalność jest niemożliwa do bezpośredniej realizacji przy użyciu metamodelu asocjacyjnego. Informacja w zakresie koniecznych atomów semantycznych tej własności musi zostać zamodelowana przy pomocy istniejących mechanizmów metamodelu asocjacyjnego, a ścisła semantyka tej własności możliwej musi zostać określona w warstwie modelu, a następnie zaimplementowana w warstwie logiki systemu.

### Przykład zastosowania

Przykładem zastosowania wzorca sieci uogólnionej może być sieć społecznościowa, której WĘZŁAMI są profile osób, natomiast KRAWĘDZIAMI utrzymywanie pewnej relacji (np. znajomości, relacji rodzinnych, itp.) pomiędzy profilami. Asocjacyjny model takiej sieci pokazano na rys. 4.49. Rodzaj relacji stanowić może etykietę dla krawędzi. Przyjęto, iż w ramach modelu klasycznych sieci społecznościowych, powiązania między profilami są binarne. Dodatkowo, elementy, które mogą być związane, ograniczone są do profili, bowiem nie przewiduje się relacji pomiędzy innymi relacjami. Implementacja wzorca obejmuje własności możliwe:

- binarność powiązań sieci uogólnionej,
- węzłowe ograniczenie złącza,
- etykietowalność węzłów sieci uogólnionej,
- etykietowalność krawędzi sieci uogólnionej.



Rysunek 4.49. Przykład zastosowania wzorca modelowania sieci uogólnionej

## 4.2. Wzorce dedykowane

W niniejszym podrozdziale opisano wzorce modelowania, które są dedykowane metamodelowi asocjacyjnemu.

### 4.2.1. Bikompozycyjny Tandem Asocjacja-Kolekcja

#### Opis problemu

Do kluczowych postulatów modelowania asocjacyjnego należą rozdział danych od związków oraz zapewnienie, iż każda kategoria modelowania pełnić będzie tylko jedną funkcję. Postulaty te spowodowały, iż asocjacja posiada zdolność wiązania, a kolekcja przechowywania danych. Fakt ten powoduje powstanie zagadnienia związanego z realizacją pojęcia, które powinno spełniać oba wymagania, tj. zarówno przechowywać pewne dane, jak również je wiązać.

Nazwa wzorca: BACT – *bikompozycyjny tandem asocjacja-kolekcja* jest nieprzypadkowa. Termin *bikompozycyjny* odnosi się do określenia kompozycji łączącej kolekcję z asocjacją – nadając jej charakter obustronnej. Oznacza to, iż ani aspekt łączenia danych nie może istnieć bez aspektu przechowywania, ani odwrotnie. Termin *tandem* wskazuje na nieseparowalną kooperację pomiędzy oboma aspektami wchodzących w skład wzorca (asocjacją i kolekcją) oraz oznacza, że wskazują jeden, funkcjonalnie złożony byt [74, 77].

#### Elementy składowe

Pojęcia stanowiące uczestników wzorca:

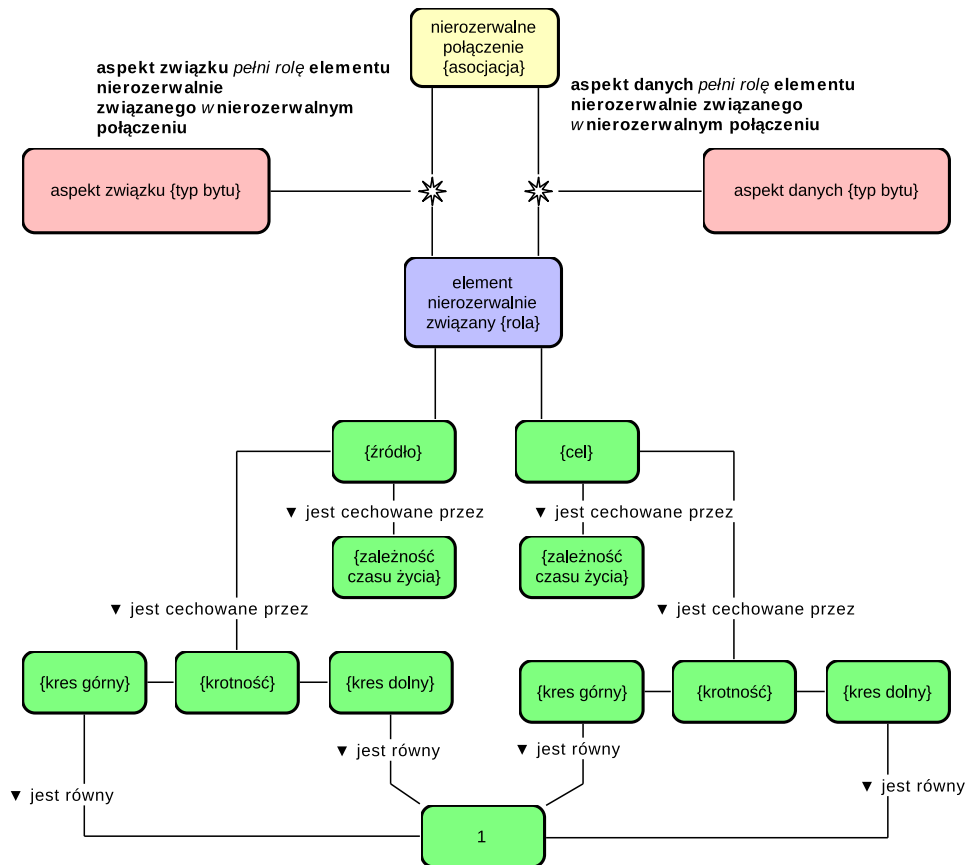
1. ASPEKT ZWIĄZKU – pojęcie pełniące rolę aspektu związanego z łączeniem danych przez modelowany byt.
2. ASPEKT DANYCH – pojęcie pełniące rolę aspektu związanego z przechowywaniem danych przez modelowany byt.

Pojęcia stanowiące powiązanie między uczestnikami:

1. NIEROZERWALNE POŁĄCZENIE – element łączący jeden konkretny ASPEKT ZWIĄZKU z jednym konkretnym ASPEKTEM DANYCH w nierozdzielny sposób.

#### Specyfikacja wzorca

Na diagramie pojęć (rys. 4.50) przedstawiono pojęcia występujące w ramach specyfikacji koniecznych atomów semantycznych wzorca BACT. Poniżej wyspecyfikowano konieczne atomy semantyczne konieczne do zrealizowania w ramach implementacji wzorca. *Konieczne atomy semantyczne*



Rysunek 4.50. Diagram pojęć prezentujący konieczne atomy semantyczne wzorca BACT

K.1 BACT *jest konkretyzacją koncepcji wzorca*

K.2 BACT *zawsze posiada aspekt danych, który jest konkretyzacją typu bytu*

K.3 BACT *zawsze posiada aspekt związku, który jest konkretyzacją typu bytu*

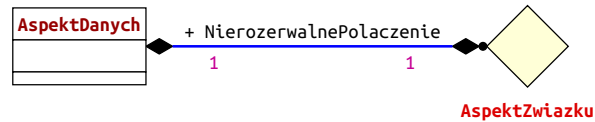
K.4 nierozierwalne połączenie *jest konkretyzacją asocjacji*

K.4.1 aspekt związku *zawsze pełni rolę elementu nierozierwalnie związanego w nierozierwalnym połączeniu*

K.4.2 aspekt danych *zawsze pełni rolę elementu nierozierwalnie związanego w nierozierwalnym połączeniu*

K.4.2.1 źródło i cel elementu nierozierwalnie związanego *są cechowane przez zależność czasu życia*

K.4.2.2 cel i źródło elementu nierozierwalnie związanego *są cechowane przez krotność, która posiada kres górny i kres dolny, które są równe 1*



Rysunek 4.51. Diagram AML reprezentujący implementację wzorca BACT

### Implementacja rozwiązania

$$\diamond \textit{AspektZwiazku} [1] \xrightarrow{+ \textit{NerozerwalnePolaczenie}} [1] \square \textit{AspektDanych} \quad (4.84)$$

$$\square \textit{AspektDanych} \{ \}; \quad (4.85)$$

Implementację wzorca BACT w metamodelu asocjacyjnym przedstawiono na rys. 4.51 oraz w postaci wyrażeń AFN 4.84 oraz 4.85. Wszystkie wyspecyfikowane atomy semantyczne zostały w pełni zrealizowane:

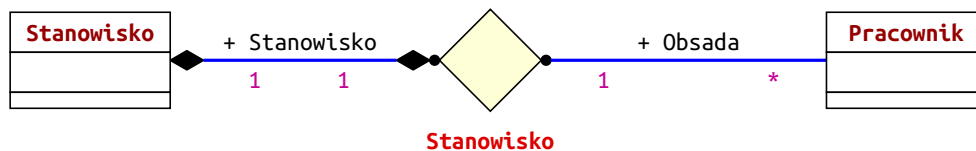
1. Implementacja K.1 polega na utworzeniu nowego modelu asocjacyjnego.
2. Implementacja K.2 została zrealizowana poprzez utworzenie kolekcji  $\square \textit{AspektDanych}$ .
3. Implementacja K.3 polegała na utworzeniu asocjacji  $\diamond \textit{AspektZwiazku}$ .
4. Atom semantyczny K.4 został zrealizowany przez utworzenie roli *NerozerwalnePolaczenie* w asocjacji  $\diamond \textit{AspektZwiazku}$ .
5. Implementacja K.4.1 została zrealizowana poprzez określenie właściciela roli *NerozerwalnePolaczenie* jako asocjacji  $\diamond \textit{AspektZwiazku}$ .
6. Implementacja K.4.2 została zrealizowana poprzez określenie celu roli *NerozerwalnePolaczenie* z asocjacji  $\diamond \textit{AspektZwiazku}$  jako kolekcji  $\square \textit{AspektDanych}$ .
7. Implementacja atomu semantycznego K.4.2.1 polegała na umiejscowieniu kompozycji po obu stronach roli *NerozerwalnePolaczenie* asocjacji  $\diamond \textit{AspektZwiazku}$ .
8. Implementacja atomu semantycznego K.4.2.2 polegała na określeniu krotności 1 po obu stronach roli *NerozerwalnePolaczenie* asocjacji  $\diamond \textit{AspektZwiazku}$ .

### Przykład zastosowania

BACT jest wzorcem niezwykle pierwotnym i znajduje potencjalne zastosowanie zawsze tam, gdzie modelowane są elementy rzeczywistości stanowiące byty.

Przykładowe zastosowanie wzorca w domenie przedsiębiorstwa może obejmować zamodelowanie stanowisk zajmowanych przez pracowników. Stanowisko, samo w sobie jest bytem, lecz ma również aspekt związku, który łączy pracowników zajmujących to stanowisko. Stanowisko pełni zatem rolę zarówno ASPEKTU ZWIĄZKU, jak również ASPEKTU DANYCH.

Przykład obrazuje diagram AML na rys. 4.52. W modelu przedstawionym na tym przykładzie, asocjacja  $\diamond$  *Stanowisko* odpowiada ASPEKTOWI ZWIĄZKU, z kolei kolekcja  $\square$  *Stanowisko* ASPEKTOWI DANYCH. Rola  $\diamond$  *Stanowisko*  $\xrightarrow{+Stanowisko}$   $\square$  *Stanowisko* stanowi NIEROZERWALNEPOLACZENIE. Kolekcja  $\square$  *Pracownik* oraz rola  $\diamond$   $\xrightarrow{+Obsada}$   $\square$  *Pracownik* nie stanowią elementów wzorca i zostały przedstawione na diagramie w celu nakreślenia kontekstu.



Rysunek 4.52. Przykład zastosowania wzorca BACT

#### 4.2.2. Abstrakcyjny Cel Roli

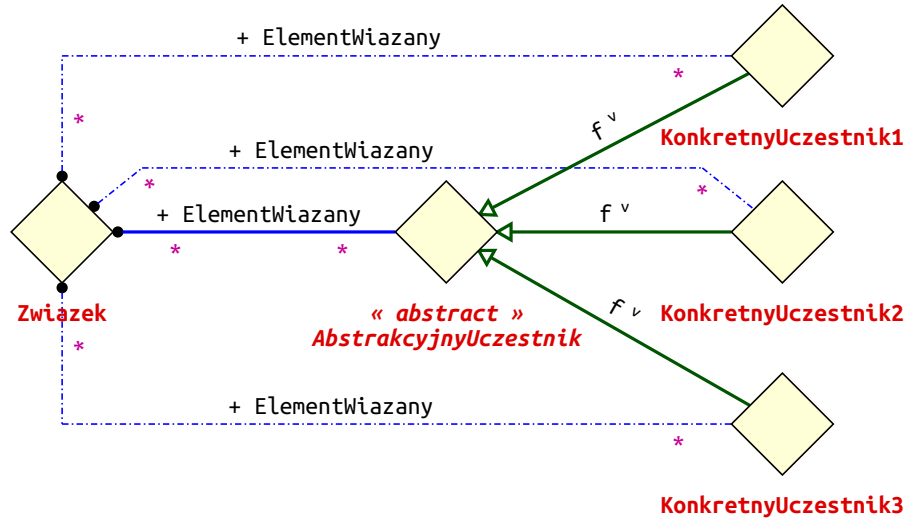
##### Opis problemu

Jednym z problemów modelowania jest tworzenie związków, w których biorą udział elementy różnych typów, pełniąc tę samą rolę. Rozwiązanie tego problemu w ramach metamodelu asocjacyjnego może polegać na wykorzystaniu mechanizmu przenoszenia praw do pełnienia ról zapewniającego polimorfizm w zakresie uczestnictwa w związku. W ramach tworzonego modelu, naturalne wydaje się być zatem zdefiniowanie węzła bazowego<sup>5</sup>, który znajduje się na końcu roli, jako węzeł abstrakcyjny.

Użycie tego wzorca upraszcza strukturę, tworząc uogólniony węzeł bazowy, mogący być celem roli. Rola w tym wzorcu tworzona jest tylko raz, natomiast prawa do jej pełnienia wynikają z dziedziczenia i tworzone są automatycznie. Szablon rozwiązania korzystającego z nakreślonej koncepcji został nazwany Abstrakcyjnym Celem Roli – Abstract Role Destination (ARD).

<sup>5</sup> Pojęcie pierwotne metamodelu asocjacyjnego: (*BaseNode*). Stanowi on generalizację asocjacji i kolekcji.





Rysunek 4.54. Diagram AML modelu implementującego konieczne atomy semantyczne wzorca abstrakcyjnego celu roli. Uczestnicy są asocjacjami.

K.4 abstrakcyjny cel roli *zawsze posiada* związek, *który jest konkretyzacją* asocjacji

K.4.1 abstrakcyjny uczestnik *zawsze pełni rolę* elementu wiązanego w związku

K.5 abstrakcyjny uczestnik *kategoryzuje* zbiór klasyfikatorów, *którego elementem jest* każdy konkretny uczestnik

K.5.1 kategoryzacja *ta zawiera* polimorfizm

### Implementacja rozwiązania

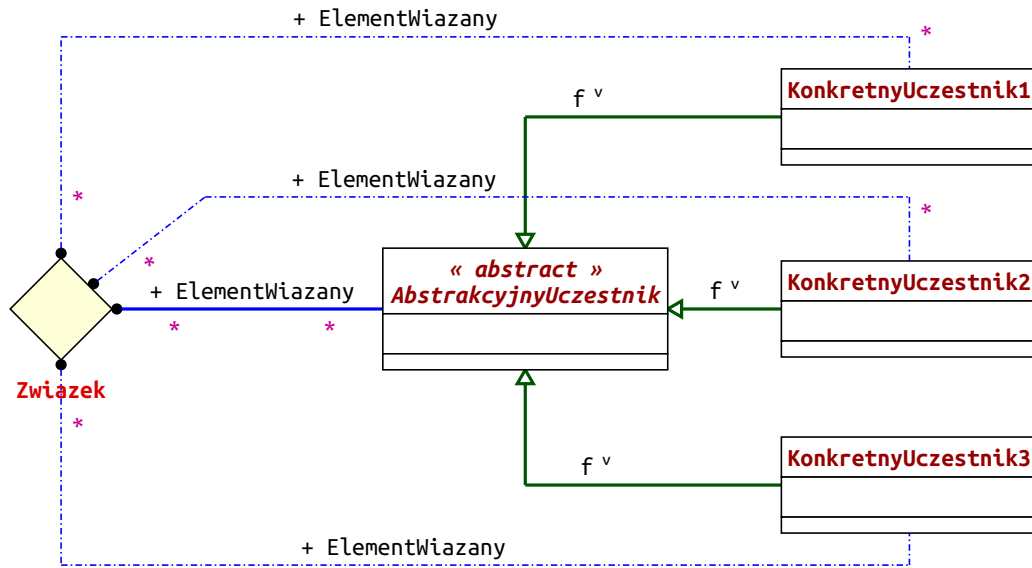
Implementację wzorca w metamodelu asocjacyjnym przedstawiono na dwa sposoby. Pierwszy sposób dotyczy zamodelowania abstrakcyjnego i konkretnego uczestnika jako asocjacje – implementację tę przedstawiono w ramach diagramu 4.54 i zbioru wyrażeń AFN 4.86-4.91. W drugim przypadku wykorzystano do tego kolekcję (rys. 4.55, wyr. 4.92-4.97).

$$\diamond \text{Związek} [*] \xrightarrow{+ElementWiazany} [*] \diamond \text{AbstrakcyjnyUczestnik} \quad (4.86)$$

$$\diamond \text{AbstrakcyjnyUczestnik} \quad (4.87)$$

$$\left\{ \begin{array}{l} \diamond \text{KonkretnyUczestnik2} \\ \diamond \text{KonkretnyUczestnik3} \\ \diamond \text{KonkretnyUczestnik1} \end{array} \right\} \xrightarrow{f^v} \diamond \text{AbstrakcyjnyUczestnik}^{\emptyset} \quad (4.88)$$





Rysunek 4.55. Diagram AML modelu implementującego konieczne atomy semantyczne wzorca abstrakcyjnego celu roli. Uczestnicy są kolekcjami.

$$\diamond \text{KonkretnyUczestnik}_1 \quad (4.89)$$

$$\diamond \text{KonkretnyUczestnik}_2 \quad (4.90)$$

$$\diamond \text{KonkretnyUczestnik}_3 \quad (4.91)$$

$$\diamond \text{Zwiazek} [*] \xrightarrow{+\text{ElementWiazany}} [*] \square \text{AbstrakcyjnyUczestnik} \quad (4.92)$$

$$\square \text{AbstrakcyjnyUczestnik} \langle \rangle ; \quad (4.93)$$

$$\left\{ \begin{array}{l} \square \text{KonkretnyUczestnik}_1 \\ \square \text{KonkretnyUczestnik}_2 \\ \square \text{KonkretnyUczestnik}_3 \end{array} \right\} \xrightarrow{f^v} \square \text{AbstrakcyjnyUczestnik}^\emptyset \quad (4.94)$$

$$\square \text{KonkretnyUczestnik}_1 \langle \rangle ; \quad (4.95)$$

$$\square \text{KonkretnyUczestnik}_2 \langle \rangle ; \quad (4.96)$$

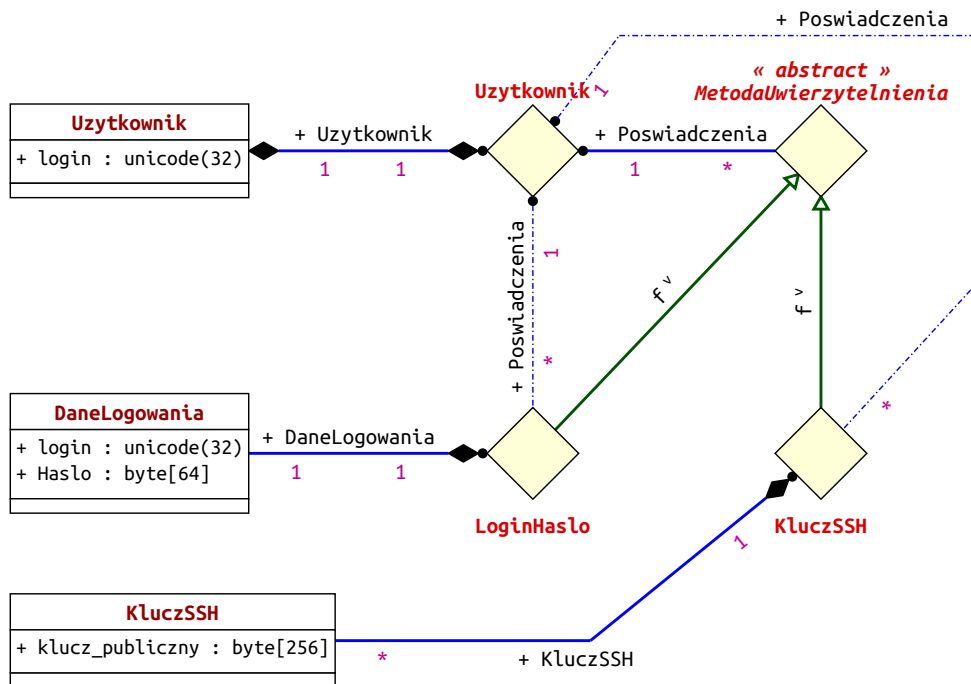
$$\square \text{KonkretnyUczestnik}_3 \langle \rangle ; \quad (4.97)$$

1. Atom semantyczny K.1 został zrealizowany poprzez utworzenie dedykowanych modeli reprezentujących wzorzec.
2. Atom K.2 został zrealizowany poprzez utworzenie asocjacji  $\diamond \text{AbstrakcyjnyUczestnik}^\emptyset$  / kolekcji  $\square \text{AbstrakcyjnyUczestnik}^\emptyset$ .
3. Atom K.2.1 zrealizowano poprzez określenie jej jako abstrakcyjnej.
4. Atom K.3 został zrealizowany poprzez zdefiniowanie szeregu asocjacji  $\diamond \text{KonkretnyUczestnik}_k$  / kolekcji  $\square \text{KonkretnyUczestnik}_k$  dla  $k \in \{1, 2, 3\}$ .
5. W obu przypadkach atom semantyczny K.4 został zrealizowany poprzez utworzenie asocjacji  $\diamond \text{Zwiazek}$ .
6. Atom K.4.1 zrealizowano przez określenie w asocjacji  $\diamond \text{Zwiazek}$  roli *ElementWiazany* i umieszczenie na jej końcu asocjacji  $\diamond \text{AbstrakcyjnyUczestnik}$  / kolekcji  $\square \text{AbstrakcyjnyUczestnik}$ .
7. K.5 zrealizowano poprzez zdefiniowanie generalizacji dla każdej z asocjacji  $\diamond \text{KonkretnyUczestnik}_k$  / kolekcji  $\square \text{KonkretnyUczestnik}_k$  dla  $k \in \{1, 2, 3\}$  do asocjacji  $\diamond \text{AbstrakcyjnyUczestnik}$  / kolekcji  $\square \text{AbstrakcyjnyUczestnik}$ .
8. Atom semantyczny K.5.1 zrealizowano poprzez określenie trybu dziedziczenia jako  $f^v$ , tj. uruchomienie jedynie dziedziczenia praw do pełnienia ról.

### Przykład zastosowania

Przykładem zastosowania Abstrakcyjnego Celu Roli może być fragment modelu zarządzania dostępem do zasobów oparty na różnych metodach poświadczeń (rys. 4.56). Model reprezentuje sytuację, w ramach której użytkownik może posiadać kilka strukturalnie odmiennych metod do uwierzytelnienia się w systemie, np. korzystając z loginu oraz hasła, bądź klucza SSH.

Rolę ABSTRAKCYJNEGO UCZESTNIKA w takim modelu pełnić może asocjacja  $\diamond \text{MetodaUwierzytelniania}^\emptyset$ , stanowiąca cel roli dla asocjacji  $\diamond \text{Uzytkownik}$ , która pełni rolę ZWIAZKU. Dzięki mechanizmowi dziedziczenia praw do pełnienia ról, asocjacje, które dziedziczą po  $\diamond \text{MetodaUwierzytelniania}^\emptyset$ , w sposób polimorficzny mają możliwość pełnienia roli reprezentującej posiadanie przez użytkownika konkretnych poświadczeń:  $\diamond \text{Uzytkownik} \xrightarrow{+\text{Poswiadczenia}} \diamond \text{MetodaUwierzytelniania}^\emptyset$ . Co za tym idzie, asocjacje  $\diamond \text{LoginHaslo}$  oraz  $\text{KluczSSH}$  pełnią rolę KONKRETNEGO UCZESTNIKA wzorca.



Rysunek 4.56. Przykład wykorzystania wzorca modelowania Abstrakcyjnego Celu Roli

### 4.2.3. Abstrakcyjny Właściciel Roli

#### Opis problemu

Tworzenie modeli opartych o związki bardzo często zakłada sytuację, w ramach której związki dzielą podobną charakterystykę, tj. mają bardzo podobną strukturę. Oczywistym sposobem zamodelowania takiej sytuacji jest utworzenie szeregu asocjacji o podobnej strukturze. Nie jest to, jednak, rozwiązanie optymalne z punktu widzenia minimalizacji struktur danych, a co za tym idzie, z punktu widzenia siły wyrazu samego modelu. Aby poradzić sobie z problemem związków o zbliżonej charakterystyce, należy rozważyć, czy jest możliwość wyodrębnienia części wspólnych tych związków i utworzenie abstrakcyjnego związku będącego kategorią nadrzędną względem modelowanych asocjacji. Szablonowe rozwiązanie korzystające z nakreślonej koncepcji zostało nazwane Abstrakcyjnym Właścicielem Roli – Abstract Role Owner (ARO). Technikę modelowania opartą o związki złożone, również w aspekcie kategoryzacyjnym opisano w pracy [73]. Jej studium przypadku w zakresie modelowania struktur reprezentacji wiedzy zostało przedstawione w ramach pracy [72].

#### Elementy składowe

Pojęcie pełniące rolę uczestnika wzorca:

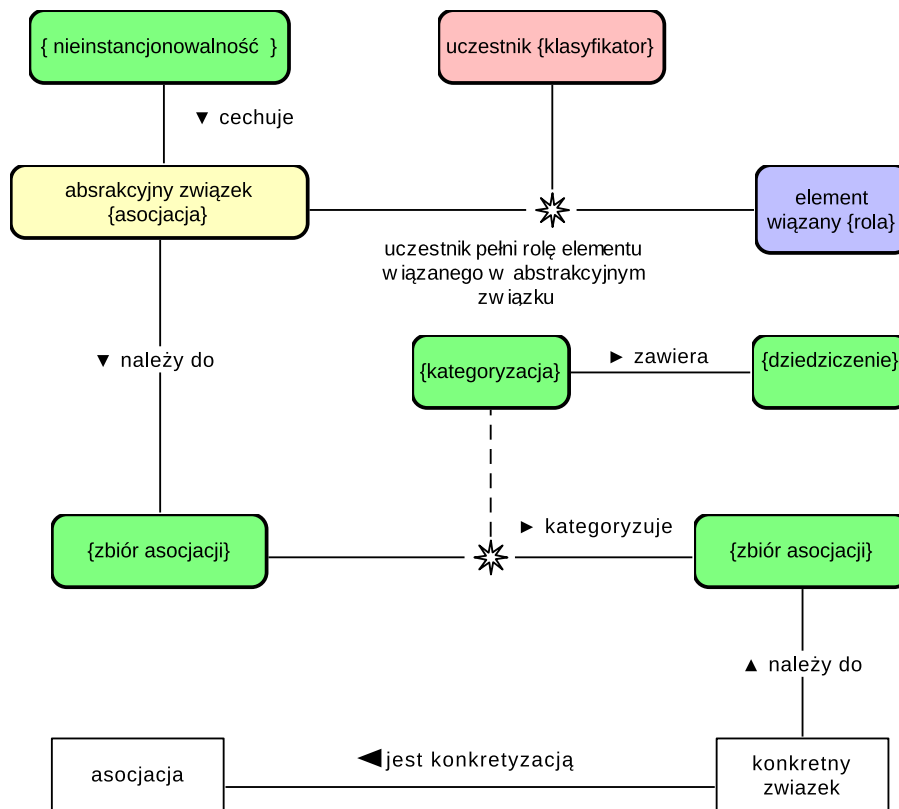
1. UCZESTNIK – pojęcie, które modeluje element biorący udział w definicji związku poprzez pełnienie w nim roli.

Pojęcia stanowiące związki pomiędzy uczestnikami:

1. ABSTRAKCYJNY ZWIĄZEK – pojęcie, które pełni rolę abstrakcji wspólnej charakterystyki konkretnych związków,
2. KONKRETNY ZWIĄZEK<sub>k</sub> – pojęcie, które pełni rolę związku definiującego pewne, określone powiązania.

### Specyfikacja wzorca

Rysunek 4.57 przedstawia diagram pojęć zawierający pojęcia i konieczne atomy semantyczne wzorca. Poniżej przedstawiono konieczne atomy semantyczne w ramach specyfikacji wzorca.

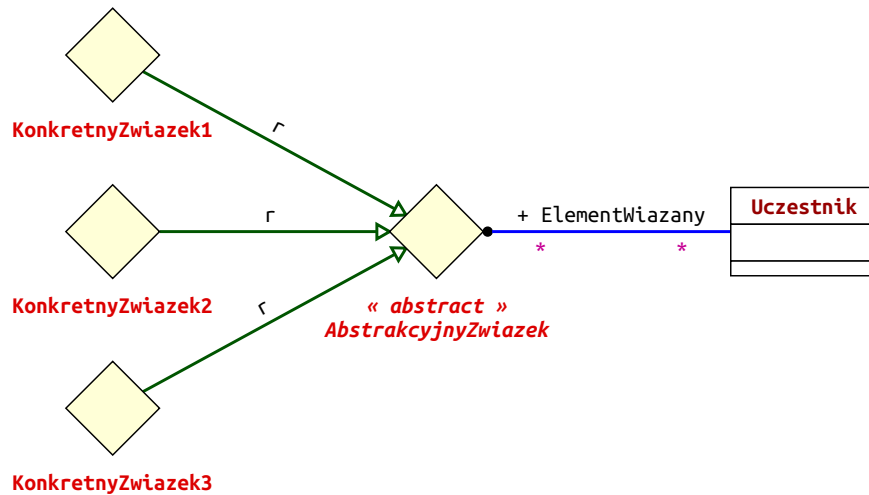


Rysunek 4.57. Diagram pojęć prezentujący konieczne atomy semantyczne koncepcji wzorca abstrakcyjnego właściciela roli

#### Konieczne atomy semantyczne

K.1 abstrakcyjny właściciel roli *jest konkretyzacją* koncepcji wzorca

K.2 abstrakcyjny właściciel roli *zawsze posiada* uczestnika, *który jest konkretyzacją* klasyfikatora



Rysunek 4.58. Diagram AML modelu implementującego konieczne atomy semantyczne wzorca abstrakcyjnego właściciela roli.

K.3 abstrakcyjny właściciel roli *zawsze posiada* abstrakcyjny związek, *który jest konkretyzacją* asocjacji

K.3.1 abstrakcyjny związek *jest cechowany przez* nieinstancjonowalność

K.3.2 uczestnik *zawsze pełni rolę* elementu wiążanego w abstrakcyjnym związku

K.4 abstrakcyjny właściciel roli *zawsze posiada* *przynajmniej jeden* konkretny związek, *który jest konkretyzacją* asocjacji

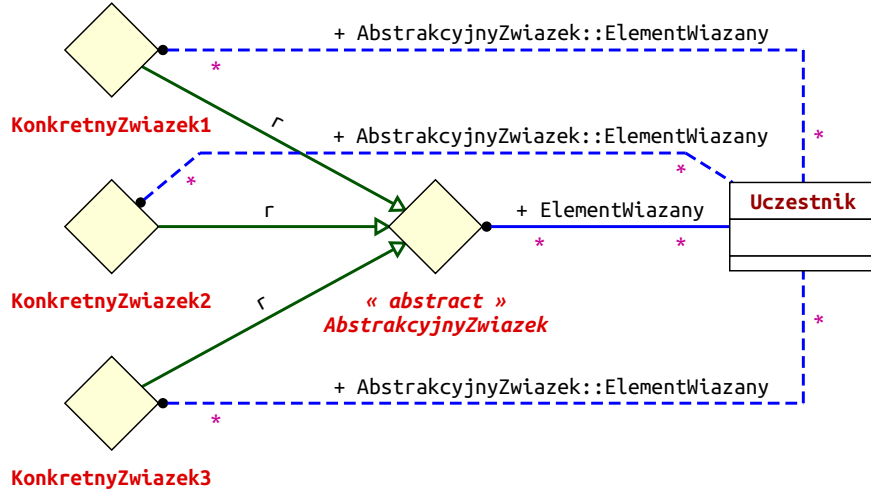
K.5 abstrakcyjny związek *kategoryzuje* zbiór asocjacji, *którego elementem jest* *każdy* konkretny związek

K.5.1 kategoryzacja *ta zawiera* dziedziczenie

### Implementacja rozwiązania

Na rys. 4.58 oraz 4.59 i wyrażeniach AFN 4.98-4.103 pokazano implementację wzorca w metamodelu asocjacyjnym. Oba diagramy prezentują ten sam model. Diagram na rys. 4.59 wzbogaca diagram o dodatkowe elementy prezentacyjne, pokazujące wszystkie role, również te odziedziczone w wyniku implementacji atomu semantycznego K.5.1.

$$\diamond \text{AbstrakcyjnyZwiazek} [*] \xrightarrow{+ \text{ElementWiazany}} [*] \square \text{Uczestnik} \quad (4.98)$$



Rysunek 4.59. Diagram AML modelu implementującego konieczne atomy semantyczne wzorca abstrakcyjnego właściciela roli. Na diagramie pokazano odziedziczone role.

$$\left\{ \begin{array}{l} \diamond \text{KonkretnyZwiazek1} \\ \diamond \text{KonkretnyZwiazek3} \\ \diamond \text{KonkretnyZwiazek2} \end{array} \right\} \xrightarrow{r} \diamond \text{AbstrakcyjnyZwiazek}^{\emptyset} \quad (4.99)$$

$$\diamond \text{KonkretnyZwiazek}_1 [*] \xrightarrow{+ \text{AbstrakcyjnyZwiazek}::\text{ElementWiazany} [*]} \square \text{Uczestnik} \quad (4.100)$$

$$\diamond \text{KonkretnyZwiazek}_2 [*] \xrightarrow{+ \text{AbstrakcyjnyZwiazek}::\text{ElementWiazany} [*]} \square \text{Uczestnik} \quad (4.101)$$

$$\diamond \text{KonkretnyZwiazek}_3 [*] \xrightarrow{+ \text{AbstrakcyjnyZwiazek}::\text{ElementWiazany} [*]} \square \text{Uczestnik} \quad (4.102)$$

$$\square \text{Uczestnik} \langle \rangle; \quad (4.103)$$

W ramach implementacji zrealizowano wszystkie atomy semantyczne określone w specyfikacji w poniższy sposób:

1. Atom semantyczny K.1 zrealizowano poprzez utworzenie nowego modelu reprezentującego wzorec.
2. K.2 został zrealizowany poprzez utworzenie kolekcji  $\square \text{Uczestnik}$ .

3. Atom semantyczny K.3 został zrealizowany poprzez utworzenie asocjacji  $\diamond AbstrakcyjnyZwiazek^\emptyset$ .
4. Atom K.3.1 zrealizowano poprzez określenie asocjacji  $\diamond AbstrakcyjnyZwiazek^\emptyset$  jako abstrakcyjnej.
5. Atom K.3.2 zrealizowano poprzez utworzenie roli *ElementWiazany* w asocjacji  $\diamond AbstrakcyjnyZwiazek$  i określenie jej celu jako  $\square Uczestnik$ .
6. Atom K.4 zrealizowano przez utworzenie szeregu asocjacji  $\diamond KonkretnyZwiazek_k$ , gdzie  $k \in \{1, 2, 3\}$ .
7. K.5 zrealizowano poprzez zdefiniowanie generalizacji dla każdej z asocjacji  $\diamond KonkretnyZwiazek_k$  dla  $k \in \{1, 2, 3\}$  do asocjacji  $\diamond AbstrakcyjnyZwiazek$ .
8. Atom semantyczny K.5.1 został utworzony poprzez określenie trybu dziedziczenia jako *r*, tj. uruchomienie jedynie dziedziczenia ról w ramach generalizacji asocjacji.

### Przykład zastosowania

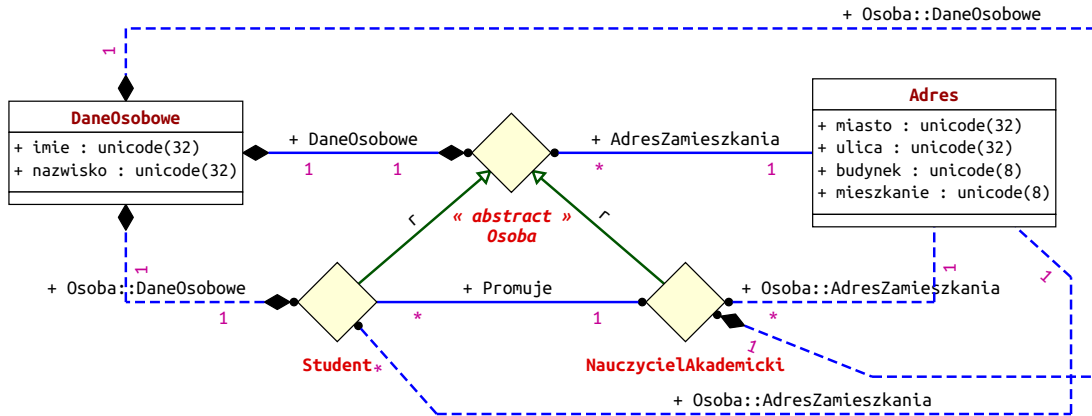
Przedstawiony przykład wzorca Abstrakcyjnego Właściciela Roli dotyczy fragmentu modelu uczelni wyższej dotyczącego zagadnień reprezentacji osób rozróżnialnych w ramach tej dziedziny (rys. 4.60). Model zawiera reprezentacje dwóch typów osób: studentów oraz nauczycieli akademickich. Dla każdej z osób model pozwala na zdefiniowanie oraz przechowywanie informacji o danych osobowych oraz o adresie. Dodatkowo, model daje możliwość wyrażenia faktu, iż nauczyciel akademicki może być promotorem studentów.

Realizacja wzorca opiera się o asocjację  $\diamond Osoba^\emptyset$ , która pełni rolę ABSTRAKCYJNEGO ZWIĄZKU. UCZESTNIKAMI tego związku są kolekcje, które znajdują się na końcu ról posiadanych przez  $\diamond Osoba^\emptyset$ , tj.  $\square DaneOsobowe$  oraz  $\square Adres$ . Mechanizm dziedziczenia ról związany z określonym trybem generalizacji pomiędzy  $\diamond Osoba^\emptyset$  a asocjacjami  $\diamond Student$  oraz  $\diamond NauczycielAkademicki$  (pełniących rolę KONKRETNYCH ZWIĄZKÓW) zapewnia, iż dziedziczalne role  $\diamond Osoba^\emptyset$   $\xrightarrow{+DaneOsobowe}$   $\square DaneOsobowe$  i  $\diamond Osoba^\emptyset$   $\xrightarrow{+AdresZamieszkania}$   $\square Adres$  zostały odziedziczone przez specjalizacje. Niezależnie, każda ze specjalizacji może posiadać własne role oraz własne uczestnictwo w rolach, co zobrazowano rolą  $\diamond NauczycielAkademicki$   $\xrightarrow{+Promuje}$   $\diamond Student$ .

#### 4.2.4. Kaskada asocjacji

##### Opis problemu

Podczas modelowania opartego o precyzyjne definiowanie związków niejednokrotnie istnieje potrzeba tworzenia budowy powiązań o charakterze hierarchicznym.



Rysunek 4.60. Przykład wykorzystania wzorca modelowania Abstrakcyjnego Właściciela Roli

Przez hierarchię rozumie się tutaj tworzenie wielopoziomowych związków. Poziomy te pozwalają uchwycić aspekty związków. Pewne związki nie mają charakteru jednorodnego, a uczestników tych związków nie można określić w sposób zupełny, gdyż odnoszą się jedynie do niektórych ich aspektów. Oznacza to, że stanowią one pewne semantycznie rozróżnialne rodzaje powiązania uczestnika w ramach związku. Przykładem takiego aspektowego związku może być związek zależności pomiędzy osobami. Aspekt takiej zależności może reprezentować zależność: służbową, funkcyjną, nieformalną, wynikającą z hierarchii stopni, etc.

### Elementy składowe

Pojęcie stanowiące uczestnika wzorca:

1. UCZESTNIK KASKADY – pojęcie, które modeluje element biorący udział w definicji związku poprzez pełnienie w nim roli.

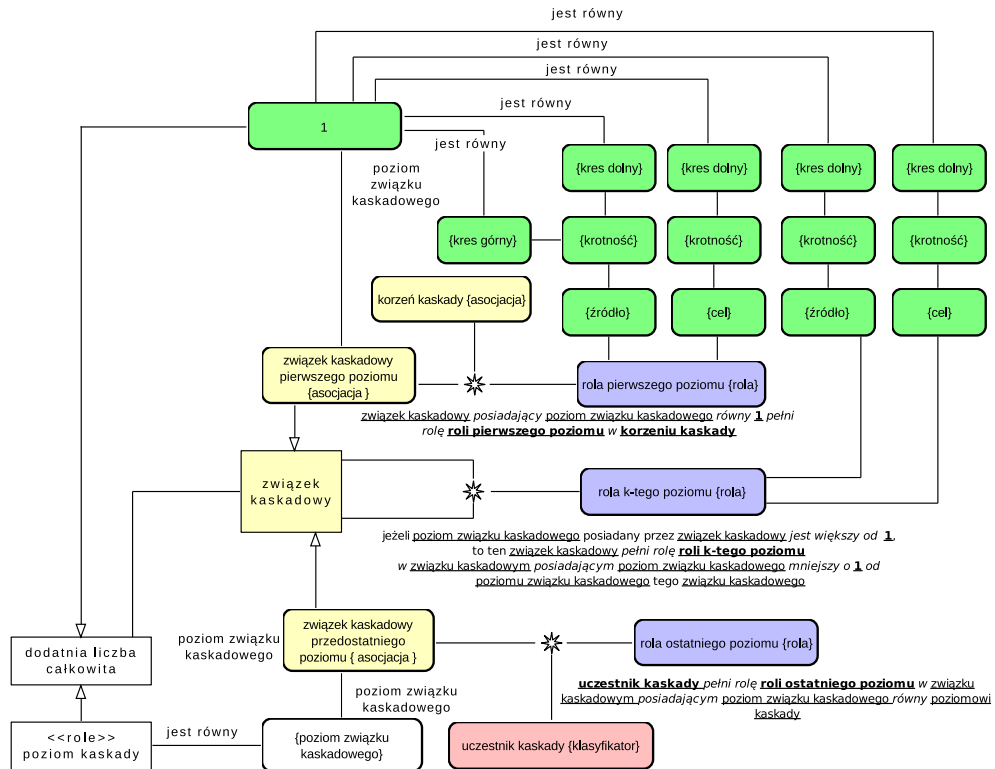
Pojęcia stanowiące powiązania pomiędzy uczestnikami:

1. KORZEŃ KASKADY – pojęcie pełniące rolę abstrakcji korzenia związku kaskadowego, tj. stanowiącego punkt wyjściowy hierarchii kolejnych aspektów.
2. ZWIĄZEK KASKADOWY  $k$ -TEGO POZIOMU – pojęcie pełniące rolę abstrakcji elementu hierarchii związku definiującego właściciela roli zawierającej aspekt na  $k$ -tym poziomie, gdzie  $k \in \mathbb{N}^+$  oraz  $k \leq n$ , przy czym  $n$  oznacza wysokość hierarchii kaskady.

### Specyfikacja wzorca

Na rysunku 4.61 przedstawiono diagram pojęć zawierający pojęcia oraz konieczne atomy semantyczne wzorca, wyspecyfikowane poniżej. W celu zachowania generycz-





Rysunek 4.61. Diagram pojęć przedstawiający strukturę pojęciową wzorca kaskady asocjacji

nego charakteru związku w ramach półformalnej specyfikacji, zdefiniowano dodatkowe pojęcia odnoszące się do kaskadowych struktur wielopoziomowych.

### poziom związku kaskadowego

Typ pojęcia: rola

Pojęcie ogólne: dodatnia liczba całkowita

Definicja: poziom, na którym znajduje się pewien element kaskady

Uwaga: Pojęcie służy do określenia, na którym poziomie znajduje się dany element kaskady. Dla korzenia kaskady przyjmuje się niejawnie, iż poziom ten jest równy zero.

### poziom kaskady

Typ pojęcia: rola

Pojęcie ogólne: dodatnia liczba całkowita

Definicja: liczba poziomów, który posiada kaskada asocjacji.

## *Konieczne atomy semantyczne*

K.1 kaskada asocjacji *jest konkretyzacją* koncepcji wzorca

K.1.1 kaskada asocjacji *posiada* poziom kaskady

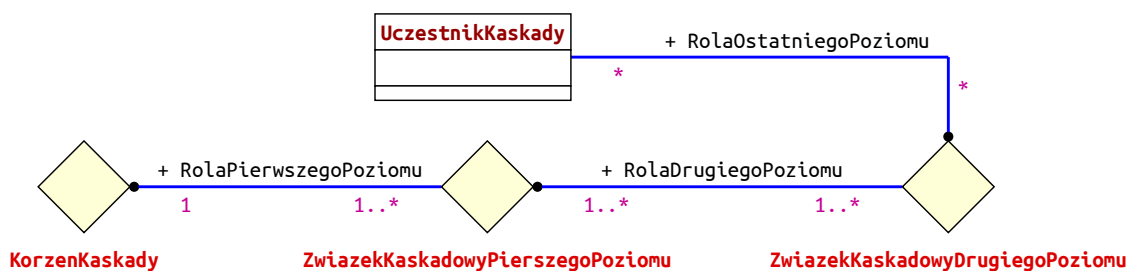
K.2 kaskada asocjacji *zawsze posiada* uczestnika kaskady, *który jest konkretyzacją* klasyfikatora

- K.3 kaskada asocjacji *zawsze posiada* *przynajmniej jeden* związek kaskadowy, *który jest konkretyzacją* asocjacji
- K.3.1 *każdy* związek kaskadowy *posiada* poziom związek kaskadowego
- K.3.1.1 poziom związek kaskadowego *danego* związku kaskadowego *zawsze jest* *równy lub mniejszy od* poziomu kaskady kaskady asocjacji, *która go posiada*
- K.3.2 *jeżeli* poziom związek kaskadowego *posiadany przez* związek kaskadowy *jest większy od* 1 *to ten* związek kaskadowy *pełni rolę* roli  $k$ -tego poziomu *w* związku kaskadowym *posiadającym* poziom związek kaskadowego *mniejszy o* 1 *od* poziomu związku kaskadowego *tego* związku kaskadowego
- K.3.2.1 źródło roli  $k$ -tego poziomu *jest cechowane przez* krotność, *która posiada* kres dolny *równy* 1
- K.3.2.2 cel roli  $k$ -tego poziomu *jest cechowany przez* krotność, *która posiada* kres dolny *równy* 1
- K.3.3 uczestnik kaskady *pełni rolę* roli ostatniego poziomu *w* związku kaskadowym *posiadającym* poziom związek kaskadowego *równy* poziomowi kaskady
- K.4 kaskada asocjacji *zawsze posiada* korzeń kaskady, *który jest konkretyzacją* asocjacji
- K.4.1 związek kaskadowy *posiadający* poziom związek kaskadowego *równy* 1 *pełni rolę* roli pierwszego poziomu *w* korzeniu kaskady
- K.4.1.1 źródło roli pierwszego poziomu *jest cechowane przez* krotność, *która posiada* kres górny *oraz* kres dolny *równe* 1
- K.4.1.2 cel roli pierwszego poziomu *jest cechowany przez* krotność, *która posiada* kres dolny *równy* 1

### Implementacja rozwiązania

Na diagramie znajdującym się na rys. 4.62 oraz w wyr. 4.104-4.107 przedstawiono przykładową implementację kaskady asocjacji o dwóch poziomach. W dalszej kolejności przedstawiony zostanie opis realizacji koniecznych atomów semantycznych w ramach specyfikacji wzorca.

1. Atom K.1 zrealizowano poprzez utworzenie odrębnego modelu zawierającego przykładową implementację wzorca.
2. Atom K.1.1 został zrealizowany poprzez określenie poziomu kaskady dla tej przykładowej implementacji jako 2.
3. K.2 został zrealizowany poprzez utworzenie kolekcji  $\square$ *UczestnikKaskady*.
4. K.3 zrealizowano poprzez utworzenie asocjacji  $\diamond$ *ZwiazekKaskadowyPierwszegoPoziomu* oraz  $\diamond$ *ZwiazekKaskadowyDrugiegoPoziomu*.
5. K.3.1 oraz K.3.1.1 zrealizowano przez nadanie asocjacji  $\diamond$ *ZwiazekKaskadowyPierwszegoPoziomu* semantyki posiadania poziomu związku kaskadowego równego jeden, natomiast asocjacji  $\diamond$ *ZwiazekKaskadowyDrugiegoPoziomu* semantyki posiadania związku kaskadowego równego dwa.
6. Atom semantyczny K.3.2 zrealizowano poprzez utworzenie roli *RolaDrugiegoPoziomu* w asocjacji  $\diamond$ *ZwiazekKaskadowyPierwszegoPoziomu*
7. Atom semantyczny K.3.2.1 oraz K.3.2.2 zrealizowano poprzez określenie krotności roli wymienionej powyżej jako 1..\* zarówno po stronie związku, jak również elementu wiązanego.
8. K.3.3 został zrealizowany poprzez utworzenie roli *RolaOstatniegoPoziomu* w asocjacji  $\diamond$ *ZwiazekKaskadowyDrugiegoPoziomu*.
9. K.4 zrealizowano poprzez utworzenie asocjacji  $\diamond$ *KorzenKaskady*.
10. K.4.1 zrealizowano poprzez utworzenie roli *RolaPierwszegoPoziomu* w asocjacji  $\diamond$ *KorzenKaskady* oraz określenie jej celu jako  $\diamond$ *ZwiazekKaskadowyPierwszegoPoziomu*.
11. Atom semantyczny K.4.1.1 zrealizowano przez określenie krotności roli jako 1 po stronie związku.
12. Atom semantyczny K.4.1.2 zrealizowano przez określenie krotności roli jako 1..\* po stronie elementu wiązanego.



Rysunek 4.62. Diagram AML reprezentujący implementację wzorca kaskady asocjacji

$$\begin{array}{c}
\diamond \text{ZwiazekKaskadowyPierszegoPoziomu} \\
[1] \xrightarrow{+RolaDrugiegoPoziomu(\diamond \text{ZwiazekKaskadowyDrugiegoPoziomu})} [1..*] \\
\diamond \text{ZwiazekKaskadowyDrugiegoPoziomu}
\end{array} \quad (4.104)$$

$$\begin{array}{c}
\diamond \text{KorzenKaskady} \\
[1..*] \xrightarrow{+RolaPierwszegoPoziomu(\diamond \text{ZwiazekKaskadowyPierszegoPoziomu})} [1..*] \\
\diamond \text{ZwiazekKaskadowyPierszegoPoziomu}
\end{array} \quad (4.105)$$

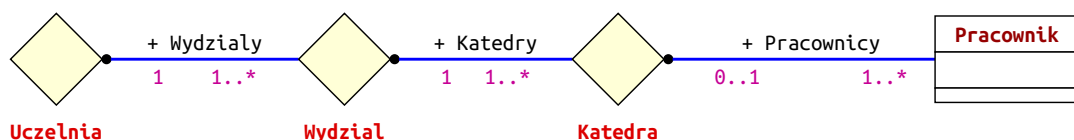
$$\begin{array}{c}
\diamond \text{ZwiazekKaskadowyDrugiegoPoziomu} \\
[*] \xrightarrow{+RolaOstatniegoPoziomu(\square \text{UczestnikKaskady})} [*] \\
\square \text{UczestnikKaskady}
\end{array} \quad (4.106)$$

$$\square \text{UczestnikKaskady} \langle \rangle ; \quad (4.107)$$

### Przykład zastosowania

Wzorzec kaskady asocjacji znajduje swoje zastosowanie w modelowaniu struktur reprezentujących związki o charakterze hierarchicznym. Przykładem takiej struktury może być typowa struktura uczelni, która składa się z wydziałów, składających się z katedr, do których z kolei przypisani są pracownicy. Przykład modelu asocjacyjnego opartego o wzorzec kaskady asocjacji dla takiej struktury przedstawiono na rys. 4.63.

W ramach modelu rolę uczestnika wzorca KORZENIA KASKADY pełni asocjacja  $\diamond \text{Uczelnia}$ . Kolejno, asocjacje  $\diamond \text{Wydzial}$  i  $\diamond \text{Katedra}$  są związkami kaskadowymi następnych poziomów. Kolekcja  $\square \text{Pracownik}$ , z kolei, stanowi UCZESTNIKA KASKADY.



Rysunek 4.63. Diagram AML obrazujący przykład zastosowania wzorca kaskady asocjacji

### **4.3. Podsumowanie**

W niniejszym rozdziale wyspecyfikowano dziewięć wzorców modelowania, tj. listę, słownik, autoreferencję, pęk ról, sieć uogólnioną, BACT, ARD, ARO oraz kaskadę asocjacji. Wzorce te przenoszą koncepcje myślowe często występujące podczas budowy modeli asocjacyjnych. Charakteryzują się one zarówno dużą pojemnością semantyczną, ale przede wszystkim dużą siłą wyrazu. Część wzorców ma charakter generyczny, tj. ich struktura zależy od konkretnego zastosowania, które determinuje ostateczną postać struktury. Niezależnie, część z wzorców posiada szereg własności, które wpływają na suplementację struktur dodatkowymi ograniczeniami semantycznymi, a co za tym idzie, syntaktycznymi. Opisy wzorców opatrzone uproszczonymi przykładami dziedzinowymi, natomiast w rozdziale 8.2 opisano wystąpienia tych wzorców w pełniejszym, bardziej rozbudowanym kontekście – w modelu systemu reprezentacji wiedzy.

## 5. Translacja semantyki modeli asocjacyjnych

W ramach pracy opracowano autorską metodę translacji semantyki, która ma charakter ogólny, tj. może zostać dostosowana do wykorzystania dla dowolnych metamodeli danych. Na potrzebę translacji semantyki modeli asocjacyjnych do innych konceptualnych modeli danych, przedstawiono tę metodę w kontekście metamodelu asocjacyjnego.

Należy zauważyć, iż opracowana metoda nie skupia się na zagadnieniach automatycznej transformacji struktur danych w sensie syntaktycznym, tzn. nie dotyczy technik przekształcania modeli w rozumieniu inżynierii opartej o modele<sup>1</sup> (zob. rozdz. 2.6), lecz definiuje proces odwzorowywania semantyki modelu wyekstrahowanej z metamodelu.

### 5.1. Koncepcja translacji

W niniejszej sekcji przedstawiono koncepcję procesu translacji semantyki modeli danych. Poniżej przedstawiono zarys tej koncepcji, określający poszczególne przekształcenia. Przekształcenia te opisano w sposób szczegółowy w kolejnych podsekcjach.

Przekształcenia wchodzące w skład procesu definicji translacji semantyki między metamodelem źródłowym a docelowym:

1. Ekstrakcja semantyki metamodelu źródłowego.
  - a) Identyfikacja **kategorii gramatycznych** metamodelu źródłowego.
  - b) Zdefiniowanie **zbioru pojęć** odpowiadających zidentyfikowanym kategoriom gramatycznym metamodelu źródłowego, przy użyciu systemu pojęć CLoM.
  - c) Zbudowanie **atomów semantycznych** metamodelu źródłowego na bazie zdefiniowanych pojęć poprzez uchwycenie ich istotnych charakterystyk tj. właściwości i ograniczeń, przy użyciu systemu pojęć CLoM.
  - d) Identyfikacja **elementarnych konstrukcji gramatycznych** w postaci prostych<sup>2</sup> powiązań między kategoriami gramatycznymi metamodelu źródłowego.

---

<sup>1</sup> ang. *Model-Driven Engineering*

<sup>2</sup> W SBVR: pojedynczy *verb concept* z dowolną liczbą ról.

- e) Zbudowanie **atomów semantycznych** metamodelu źródłowego na bazie zidentyfikowanych elementarnych konstrukcji gramatycznych, przy użyciu systemu pojęć CLoM.
- 2. Budowa cząsteczek semantycznych metamodelu źródłowego.
  - a) Zbudowanie **złożonych konstrukcji semantycznych** na bazie atomów semantycznych.
- 3. Utworzenie cząsteczek semantycznych metamodelu docelowego dla cząsteczek semantycznych metamodelu źródłowego.
  - a) Wybór **kategorii gramatycznych** metamodelu docelowego na podstawie wiedzy eksperckiej osoby budującej zasady translacji.
  - b) Zdefiniowanie **pojęć** odpowiadających wybranym kategoriom gramatycznym metamodelu docelowego przy użyciu systemu pojęć CLoM.
  - c) Zbudowanie **atomów semantycznych** metamodelu docelowego na bazie zdefiniowanych pojęć poprzez uchwycenie ich istotnych charakterystyk tj. właściwości i ograniczeń, przy użyciu systemu pojęć CLoM.
  - d) Identyfikacja **elementarnych konstrukcji gramatycznych** w postaci prostych powiązań między kategoriami gramatycznymi metamodelu docelowego.
  - e) Zbudowanie **atomów semantycznych** metamodelu docelowego na bazie zidentyfikowanych elementarnych konstrukcji gramatycznych, przy użyciu systemu pojęć CLoM.
  - f) **Przyporządkowanie cząsteczek semantycznych** metamodelu źródłowego cząsteczkom semantycznym metamodelu docelowego i zdefiniowanie reguł mapowania.

Przekształcenia wchodzące w skład procesu wykonania translacji pomiędzy modelem źródłowym a docelowym:

1. ekstrakcja semantyki modelu źródłowego,
2. dobór wzorców mapowania,
3. utworzenie reprezentacji semantycznej modelu docelowego,
4. utworzenie modelu docelowego.

Oznaczenia metamodeli:

$\mathcal{M}$  – metamodel,

$\mathcal{M}_{AOM}$  – metamodel asocjacyjny (konkretny metamodel),

$m \triangleleft \mathcal{M}$  – model jest zgodny z metamodelem  $\mathcal{M}$ ,

$\mathcal{M}^s$  – metamodel źródłowy (w kontekście translacji),

$\mathcal{M}^d$  – metamodel docelowy (w kontekście translacji).

Oznaczenia związane z artefaktami rozważanymi w procesie translacji oraz ich zbiorami:

$e_k^s$ , gdzie:  $k \in \mathbb{N}^+$  – element  $e$  związany z metamodelem źródłowym (w kontekście translacji),

$e_k^d$ , gdzie:  $k \in \mathbb{N}^+$  – element  $e$  związany z metamodelem docelowym (w kontekście translacji),

$\mathcal{K}_{\mathcal{M}}$  – zbiór kategorii metamodelu  $\mathcal{M}$ ,

$\mathcal{C}_{\mathcal{M}}$  – zbiór pojęć jednostkowych, wyrażonych w systemie pojęć zdefiniowanym w ramach niniejszej pracy, odpowiadający kategoriom metamodelu  $\mathcal{M}$ ,

$\mathcal{P}_{\mathcal{M}}$  – zbiór cząsteczek semantycznych określonych dla metamodelu  $\mathcal{M}$ ,

$\Pi_{\mathcal{M}_1 \rightarrow \mathcal{M}_2}$  – zbiór wzorców mapowania pomiędzy metamodelami  $\mathcal{M}_1$  i  $\mathcal{M}_2$ .

Oznaczenia związane ze zbiorami atomów semantycznych metamodeli:

$\mathcal{A}_{\mathcal{M}}^{(1)}$  – zbiór monadycznych atomów semantycznych określony dla  $\mathcal{M}$ . *Monadyczny atom semantyczny* to taki atom semantyczny, który posiada odwołanie tylko do jednego pojęcia odpowiadającemu kategorii metamodelu. Przykład: [aom:BaseNode](#) jest abstrakcyjny. Monadyczne atomy semantyczne służą do reprezentowania właściwości lub charakterystyki pojęć. Atomy semantyczne, które posiadają odwołania do więcej niż jednego pojęcia również mogą mieć charakter monadyczny, w sytuacji, gdy dokładnie jedno pojęcie stanowi reprezentację kategorii metamodelu.

$\mathcal{A}_{\mathcal{M}}^{(+)}$  – zbiór poliadycznych atomów semantycznych określony dla  $\mathcal{M}$ . *Poliadyczny atom semantyczny* jest atomem semantycznym posiadającym odwołanie do co najmniej dwóch pojęć. Przykład: [aom:Assoc](#) posiada [aom:Role](#).

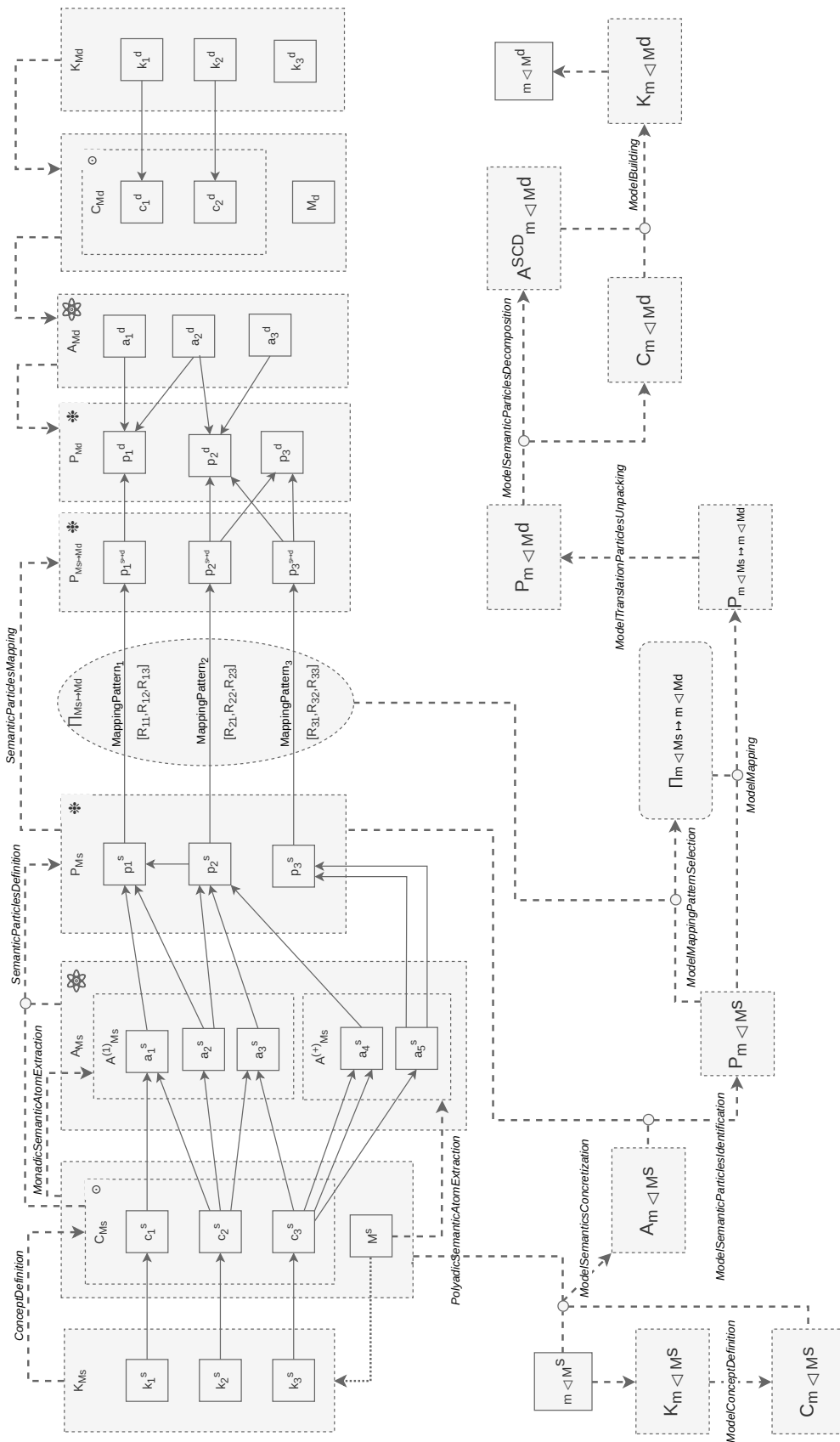
$\mathcal{A}_{\mathcal{M}}$  – zbiór wszystkich atomów semantycznych metamodelu  $\mathcal{M}$ .

Na rysunku 5.1 przedstawiono schematyczny przegląd koncepcji metody obrazującej przepływy i odwzorowania poszczególnych artefaktów oraz ich zbiorów na kolejnych etapach procesu translacji semantyki.

### 5.1.1. Ekstrakcja semantyki metamodelu źródłowego

Ekstrakcja semantyki metamodelu jest przekształceniem, którego wynikiem jest opis semantyki metamodelu w postaci  $\mathcal{A}_{\mathcal{M}}$ . Przekształcenie to wykonywane jest jednokrotnie, w procesie definiowania translacji między parą metamodeli. W ogólnym przypadku, semantyka metamodeli ma postać niejawną, tj. zapisaną w postaci języka naturalnego, wyrażeń formalnych, etc. Ideą ekstrakcji semantyki jest uchwycenie jej w sposób jawny, zgodny ze zdefiniowanym w niniejszej pracy systemem





Rysunek 5.1. Schematyczna transformacja artefaktów translacyjnych

pojęć CLoM. Aby wyekstrahować semantykę  $\mathcal{M}^s$ , wymagana jest konceptualizacja jego kategorii syntaktycznych, tj. odwzorowanie tych kategorii w CLoM. Następnie, należy zidentyfikować poszczególne charakterystyki tych kategorii oraz powiązania pomiędzy kategoriami (elementarne konstrukcje) i również wyrazić je w systemie pojęć, w formie atomów semantycznych operujących na pojęciach reprezentujących kategorie  $\mathcal{M}^s$ . Atomy semantyczne mogą mieć charakter możliwościowy, tj. mogą opisywać potencjalną semantykę możliwą do zaistnienia w ramach modelu.

Ekstrakcja semantyki  $\mathcal{M}^s$  określona jest przez trzy przekształcenia:

- definicja pojęć (*ConceptDefinition* wyr. 5.1)– przekształcenie, które ma za zadanie wydobyć z  $\mathcal{M}^s$  pojęcia reprezentujące kategorie tego metamodelu. Przekształca ono zbiór kategorii semantycznych  $\mathcal{M}^s$  w zbiór pojęć jednostkowych reprezentujących te kategorie.

$$\{k_1^s, k_2^s, \dots, k_{kn}^s | k_i^s \in \mathcal{K}_{\mathcal{M}^s}\} \xrightarrow{\text{ConceptDefinition}} \{c_1^s, c_2^s, \dots, c_{kn}^s | c_j^s \in \mathcal{C}_{\mathcal{M}^s}\} \quad (5.1)$$

- ekstrakcja monadycznych atomów semantycznych (*MonadicSemanticAtomExtraction*, wyr. 5.2) – przekształcenie mające na celu wydobyć z  $\mathcal{M}^s$  jego istotnych charakterystyk tj. właściwości i ograniczeń odnoszących się do jednego pojęcia reprezentującego kategorię metamodelu. Przekształca ono krotkę składającą się ze zbioru pojęć reprezentujących kategorię  $\mathcal{M}^s$  oraz samego  $\mathcal{M}^s$  w  $\mathcal{A}_{\mathcal{M}^s}^{(1)}$ . Sam metamodel uczestniczy w tym przekształceniu, gdyż zawiera on semantykę, która w ramach tego przekształcenia jest wydobywana.

$$(\{c_1^s, c_2^s, \dots, c_{kn}^s | c_i^s \in \mathcal{C}_{\mathcal{M}^s}\}, \mathcal{M}^s) \xrightarrow{\text{MonadicSemanticAtomExtraction}} \{a_1^s, a_2^s, \dots, a_{an}^s | a_j^s \in \mathcal{A}_{\mathcal{M}^s}^{(1)}\} \quad (5.2)$$

- ekstrakcja poliadycznych atomów semantycznych (*PolyadicSemanticAtomExtraction*, wyr. 5.3) – przekształcenie mające na celu wydobyć z  $\mathcal{M}^s$  związków pomiędzy pojęciami reprezentującymi jego kategorie. Sposób przekształcenia jest analogiczny jak w przypadku ekstrakcji monadycznych atomów semantycznych.

$$(\{c_1^s, c_2^s, \dots, c_{kn}^s | c_i^s \in \mathcal{C}_{\mathcal{M}^s}\}, \mathcal{M}^s) \xrightarrow{\text{PolyadicSemanticAtomExtraction}} \{a_1^s, a_2^s, \dots, a_{an}^s | a_j^s \in \mathcal{A}_{\mathcal{M}^s}^{(+)}\} \quad (5.3)$$

Pełny zbiór atomów semantycznych  $\mathcal{M}^s$  określa suma zbiorów atomów semantycznych monadycznych i poliadycznych (wyr. 5.4).

$$\mathcal{A}_{\mathcal{M}^s} = \mathcal{A}_{\mathcal{M}^s}^{(1)} \cup \mathcal{A}_{\mathcal{M}^s}^{(+)} \quad (5.4)$$

### 5.1.2. Budowa cząsteczek semantycznych metamodelu źródłowego

Cząsteczka semantyczna jest trójką:

$$* = (\mathcal{A}^*, \mathcal{I}^*, \mathcal{C}^*)$$

gdzie:

$\mathcal{A}^*$  – zbiór atomów semantycznych opisujących semantykę cząsteczki,

$\mathcal{I}^*$  – zbiór wewnętrznych cząsteczek wbudowanych stanowiących elementy składowe cząsteczki,

$\mathcal{C}^*$  – zbiór pojęć i ich podstawień wymaganych do zasilenia cząsteczki.

Cząsteczka określana jest dla danego metamodelu na bazie jego semantyki, wzorców modelowania oraz innych konstrukcji, które stanowią budulce wzorców mapowania. W przypadku, gdy cząsteczka semantyczna jest **cząsteczką prostą**, oznacza to, że jej zbiór skomponowanych cząsteczek semantycznych  $\mathcal{I}^*$  jest zbiorem pustym. W przypadku, gdy cząsteczka semantyczna jest **cząsteczką złożoną**, oznacza to, że jej zbiór cząsteczek semantycznych  $\mathcal{I}^*$  jest zbiorem niepustym. Cząsteczki złożone mogą być **cząsteczkami czysto złożonymi**, kiedy ich zbiór atomów semantycznych  $\mathcal{A}^*$  jest pusty.

Wbudowane cząsteczki semantyczne mogą wykluczać pewne możliwe atomy semantyczne, bądź wymagać ich zaistnienia. Operację usunięcia danego możliwego atomu semantycznego ze specyfikacji cząsteczki wbudowanej określono jako *dezaktywację*, a wymaganie jej zaistnienia jako *aktywację*.

Zbiór pojęć cząsteczki semantycznej stanowi zbiór, który determinuje sposób konkretyzacji danej cząsteczki semantycznej. W najprostszym przypadku, stanowi on sumę pojęć występujących w każdym z atomów w zbiorze abstrakcyjnych atomów semantycznych metamodelu oraz zbiorze skomponowanych cząsteczek semantycznych. Pojęcia w zbiorze pojęć cząsteczki semantycznej mogą jednak zostać poddane doprecyzowaniu na dwa, opisane poniżej, sposoby.

**Konkretyzacja** zachodzi w celu ustalenia roli pojęcia w kontekście danej cząsteczki semantycznej. Pozwala na określenie potrzeby skonkretyzowania semantyki danego pojęcia niezależnie dla kilku pojęć reprezentujących tę samą kategorię. Konkretyzacja może zostać określona globalnie, w zakresie całej cząsteczki semantycznej lub w zakresie wybranego atomu semantycznego, bądź zbioru atomów semantycznych.

**Kategoryzacja** pozwala na określenie, iż atomy semantyczne grupowane w ramach cząsteczki określają specjalizowane pojęcie metamodelu. Pozwala to odwołać się

do atomów semantycznych specyfikowanych na poziomie kategorii nieterminalnych, przypisując ją pojęciom reprezentującym kategorie terminalne. Kategoryzacja może być określona globalnie lub odnosić się do wybranego atomu semantycznego, bądź zbioru atomów semantycznych.

Idea *atomów semantycznych*, jak sama nazwa wskazuje, jest ich atomowość. Oznacza to, że posiadają one niepodzielny charakter oraz reprezentują znaczenie informacji jednostkowej. Grupując atomy semantyczne można tworzyć złożone konstrukcje semantyczne obejmujące powiązane ze sobą elementy znaczenia, które powinny być rozpatrywane łącznie w zagadnieniach translacyjnych. Ze względu na to, iż konstrukcje te grupują atomy semantyczne, zostały określone jako *cząsteczki semantyczne*. Należy dodatkowo zauważyć, iż w procesie tym jeden atom semantyczny może wchodzić w skład wielu cząsteczek semantycznych – wiele konstrukcji znaczeniowych może wykorzystywać tę samą informację jednostkową. Ma to szczególnie zastosowanie w zakresie traktowania wzorców modelowania jako cząsteczek semantycznych. Takie same, z punktu widzenia struktury, konstrukcje mogą realizować odmienne koncepcje semantyczne, w szczególności w zakresie metamodeli danych o małej pojemności semantycznej.

Podczas tego etapu następuje przekształcenie zbioru atomów semantycznych metamodelu źródłowego w zbiór, którego elementami są cząsteczki semantyczne  $\mathcal{M}^s$ . Każda cząsteczka semantyczna jest zbiorem atomów semantycznych. Proces tworzenia cząsteczek semantycznych należy zakończyć dopiero w sytuacji, gdy każdy z atomów semantycznych będzie elementem co najmniej jednej cząsteczki semantycznej.

Budowa cząsteczek semantycznych  $\mathcal{M}^s$  polega na przekształceniu *SemanticParticlesDefinition* (wyr. 5.5). Odbywa się to poprzez ewentualne łączenie poszczególnych, powiązanych ze sobą atomów semantycznych (wyr. 5.6) w taki sposób, aby realizowały one razem pewną koncepcję semantyczną. Dla przykładu, pewną cząsteczkę semantyczną stanowi zbiór atomów semantycznych AOM opisujący charakterystykę dziedziczenia z uwzględnieniem wszystkich trybów dziedziczenia oraz uczestników tego związku, gdyż wystąpienie tych elementów jest konieczne do zaistnienia dziedziczenia.

$$\{a_1^s, a_2^s, \dots, a_{an}^s | a_i^s \in \mathcal{A}_{\mathcal{M}_s}\} \xrightarrow{\text{SemanticParticlesDefinition}} \{p_1^s, p_2^s, \dots, p_{sn}^s | p_j^s \in \mathcal{P}_{\mathcal{M}_s}\} \quad (5.5)$$

$$p_x^s = \{a_u^s, a_v^s, \dots, a_w^s\}, \quad x \in \{1, 2, \dots, sn\}; \quad u, v, w \in \{1, 2, \dots, an\} \quad (5.6)$$

### 5.1.3. Utworzenie wzorców mapowania

Wzorce mapowania stanowią odwzorowania wyrażające przekształcenie semantyki wyrażonej w ramach metamodelu źródłowego na postać semantyki  $\mathcal{M}^d$ . Wejście wzorca mapowania stanowi pojedyncza cząsteczka semantyczna metamodelu źródłowego. Wyjściem wzorca mapowania jest cząsteczka semantyczna stanowiąca wynik jego zastosowania. Cząsteczka taka ma charakter cząsteczki złożonej, która składa się z jednej lub więcej cząsteczek semantycznych metamodelu docelowego. W efekcie, przekształcenie *SemanticParticlesMapping* pozwala przekształcić semantykę metamodelu źródłowego wyrażoną jako zbiór cząsteczek semantycznych  $\mathcal{P}_{\mathcal{M}^s}$  w zbiór translacyjnych cząsteczek semantycznych  $\mathcal{P}_{\mathcal{M}^s \rightarrow \mathcal{M}^d}$ , który odwzorowuje semantykę metamodelu docelowego pozyskaną w kontekście wyniku przeprowadzonej translacji.

$$\{p_1^s, p_2^s, \dots, p_{sn}^s | s_i \in \mathcal{P}_{\mathcal{M}^s}\} \xrightarrow{\text{SemanticParticlesMapping}} \{p_1^{s \rightarrow d}, p_2^{s \rightarrow d}, \dots, p_{sn}^{s \rightarrow d} | p_i \in \mathcal{P}_{\mathcal{M}^s \rightarrow \mathcal{M}^d}\} \quad (5.7)$$

Sposób, w jaki konkretna cząsteczka semantyczna jest odwzorowywana odzwierciedla wiedzę ekspercką uwzględniającą definicje atomów semantycznych wchodzących w skład cząsteczki oraz ich pojęć. Cząsteczki semantyczne metamodelu docelowego wchodzące w skład translacyjnych cząsteczek semantycznych ekstrahowane są w sposób heurystyczny. Oznacza to, iż ekspert wedle swojej wiedzy dokonuje selekcji kategorii metamodelu docelowego celem uniknięcia potrzeby konceptualizacji całego metamodelu docelowego. W następnym kroku ekstrakcja semantyki odbywa się w sposób analogiczny do metamodelu źródłowego. W dalszej części, dla każdej z cząsteczek semantycznych metamodelu źródłowego tworzona jest jej reprezentacja dla metamodelu docelowego, która jest wynikiem zastosowania wzorca mapowania (*MappingPattern*, wyr. 5.8). Zbiór wszystkich wzorców mapowania cząsteczek metamodelu źródłowego na cząsteczki translacyjne przedstawiono na wyr. 5.9.

$$p_i^s \in \mathcal{P}_{\mathcal{M}^s} \xrightarrow{\text{MappingPattern}_i} p_i^{s \rightarrow d} \in \mathcal{P}_{\mathcal{M}^s \rightarrow \mathcal{M}^d}, \text{ dla } i = 1..sn \quad (5.8)$$

$$\Pi_{\mathcal{M}^s \rightarrow \mathcal{M}^d} = \{\text{MappingPattern}_i | i = 1..sn\} \quad (5.9)$$

### 5.1.4. Przekształcenia aplikacyjne

Kolejne przekształcenia stanowią operacje realizujące konkretne translacje, tj. wykonywane są w dziedzinie modeli.

#### 5.1.4.1. Ekstrakcja semantyki modelu źródłowego

Ekstrakcja semantyki modelu źródłowego polega na konkretyzacji atomów semantycznych metamodelu źródłowego. Polega ona na podstawieniu pojęć reprezentujących konkretne kategorie modelu pod kategorie abstrakcyjne. Tym samym, model przekształcany jest do jego reprezentacji semantycznej, która ma postać zbioru skonkretyzowanych atomów semantycznych.

Ekstrakcja ta składa się z następujących przekształceń:

- zdefiniowanie pojęć modelu (*ModelConceptDefinition*, wyr. 5.10) przekształca zbiór instancji kategorii modelu źródłowego w zbiór pojęć jednostkowych w rozumieniu systemu pojęć CLoM. Pojęcia te reprezentują konkretne elementy modelu i konkretyzują ich abstrakcyjne odpowiedniki w ramach metamodelu. Przykładem może być pojęcie reprezentujące asocjację  $\diamond StaticProperty$  z modelu przedstawionego na diagramie 2.11 stanowiącą konkretyzację pojęcia [aom:Assoc](#) reprezentującego kategorię asocjacji (*Assoc*) w metamodelu asocjacyjnym.

$$\mathcal{K}_{m \triangleleft \mathcal{M}^s} \xrightarrow{\text{ModelConceptDefinition}} \mathcal{C}_{m \triangleleft \mathcal{M}^s} \quad (5.10)$$

- konkretyzacja semantyki modelu (*ModelSemanticsConcretization*, wyr. 5.11) ma na celu wydobycie semantyki z modelu w kontekście metamodelu źródłowego  $\mathcal{M}^s$ . Zadanie to odbywa się poprzez analizę kontekstu występowania poszczególnych pojęć modelu w ramach modelu  $m \triangleleft \mathcal{M}^s$ . Pozwala to uchwycić ich konkretną charakterystykę, ograniczenia oraz związki z innymi pojęciami, a następnie zidentyfikować atomy semantyczne metamodelu, które tę semantykę abstrahują. W celu wyrażenia konkretnej semantyki, tworzone są skonkretyzowane atomy semantyczne modelu  $\mathcal{A}_{m \triangleleft \mathcal{M}^s}$ . Konkretyzacja odbywa się poprzez podstawienie konkretnych pojęć modelu pod ich abstrakcyjne odpowiedniki występujące w atomach semantycznych metamodelu.

$$(\mathcal{C}_{m \triangleleft \mathcal{M}^s}, \mathcal{A}_{\mathcal{M}^s}, m \triangleleft \mathcal{M}^s) \xrightarrow{\text{ModelSemanticsConcretization}} \mathcal{A}_{m \triangleleft \mathcal{M}^s} \quad (5.11)$$

#### 5.1.4.2. Dobór wzorców mapowania

Dobór wzorców mapowania jest etapem, którego wynikiem jest określenie podzbioru wzorców mapowania, który zostanie użyty do wykonania określonej translacji. Istnieje wiele rozwiązań na wykonanie tego przekształcenia, a wybór najlepszego rozwiązania zależy od przyjętych kryteriów, które to z kolei mogą być uzależnione od wielu czynników, w tym również tak nieuchwytnych jak konwencje, style modelowania.

Postulaty, które należy uwzględnić podczas doboru wzorców to:

- *jednoznaczność odwzorowania* – w przypadku, gdy podzbiór skonkretyzowanych atomów semantycznych będzie mógł zasilić wiele konstrukcji semantycznych metamodelu źródłowego, należy wskazać, który wzorec mapowania wybrać,
- *kompletność odwzorowania* – należy zapewnić taki dobór wzorców, który pokrywa cały model.

Spełnienie tych postulatów wiąże się z szeregiem decyzji projektowych, gdyż w ogólnym przypadku jedno zagadnienie może posiadać wiele rozwiązań, w sensie kombinacji użytych wzorców mapowania. Każde z tych rozwiązań będzie posiadało pewne własności, które mogą być istotne w danym kontekście, domenie lub z powodu innych uwarunkowań związanych z modelowanym systemem. Ze względu na złożoność zagadnienia oraz kluczowe znaczenie wiedzy i doświadczenia osoby modelującej, dobór wzorców mapowania powinien następować w wyniku analizy przeprowadzonej przez eksperta. Ewentualna automatyzacja tego procesu byłaby zagadnieniem ciekawym, o dużej złożoności, jednak nie jest ona przedmiotem niniejszego opracowania.

W ramach doboru wzorców mapowania należy dokonać identyfikacji cząsteczek semantycznych modelu, korzystając w tym celu ze zbioru skonkretyzowanych atomów semantycznych oraz abstrakcyjnych cząsteczek semantycznych, zdefiniowanych w fazie definicyjnej translacji. Identyfikację tą opisuje przekształcenie *ModelSemanticParticlesIdentification* (wyr. 5.12).

$$(\mathcal{A}_{m \triangleleft \mathcal{M}^s}, \mathcal{P}_{\mathcal{M}^s}) \xrightarrow{\text{ModelSemanticParticlesIdentification}} \mathcal{P}_{m \triangleleft \mathcal{M}^s} \quad (5.12)$$

Pełen zdefiniowany zbiór wzorców mapowania  $\Pi_{\mathcal{M}^s \mapsto \mathcal{M}^d}$  stanowi przestrzeń możliwości do budowania konwencji. Cząsteczki semantyczne, zarówno źródłowe, jak również docelowe, wchodzące w skład wzorców mapowania mogą uwzględniać te same atomy semantyczne. W celu wybrania adekwatnych w ramach konkretnej translacji wzorców, należy przyporządkować konkretyzowanym cząsteczkom semantycznym wzorce mapowania, które będą je zasilać.

$$(\mathcal{P}_{m \triangleleft \mathcal{M}^s}, \Pi_{\mathcal{M}^s \mapsto \mathcal{M}^d}) \xrightarrow{\text{ModelMappingPatternSelection}} \Pi_{m \triangleleft \mathcal{M}^s \mapsto m \triangleleft \mathcal{M}^d} \quad (5.13)$$

$$\Pi_{m \triangleleft \mathcal{M}^s \mapsto m \triangleleft \mathcal{M}^d} \subset \Pi_{\mathcal{M}^s \mapsto \mathcal{M}^d} \quad (5.14)$$

### 5.1.4.3. Utworzenie reprezentacji semantycznej modelu docelowego

Przekształcenie to ma na celu zastosowanie dobranych wzorców mapowania do utworzenia reprezentacji semantycznej modelu docelowego, na którą składa się zbiór skonkretyzowanych atomów semantycznych metamodelu docelowego. Aby to uczynić, należy dokonać mapowania cząsteczek semantycznych modelu źródłowego na translacyjne cząsteczki semantyczne modelu docelowego. Transformację tą określono jako *ModelMapping* (wyr. 5.15).

$$(\mathcal{P}_{m \triangleleft \mathcal{M}^s}, \Pi_{m \triangleleft \mathcal{M}^s \mapsto m \triangleleft \mathcal{M}^d}) \xrightarrow{\text{ModelMapping}} \mathcal{P}_{m \triangleleft \mathcal{M}^s \mapsto m \triangleleft \mathcal{M}^d} \quad (5.15)$$

Translacyjne cząsteczki semantyczne stanowią nośnik semantyki modelu docelowego ze względu na fakt, iż są złożone jedynie z cząsteczek semantycznych tego modelu. Przekształcenie *ModelTranslationParticlesUnpacking* polega na pozyskaniu zbioru cząsteczek semantycznych modelu docelowego poprzez wyłuskanie wewnętrznych cząsteczek translacyjnych.

$$\mathcal{P}_{m \triangleleft \mathcal{M}^s \mapsto m \triangleleft \mathcal{M}^d} \xrightarrow{\text{ModelTranslationParticlesUnpacking}} \mathcal{P}_{m \triangleleft \mathcal{M}^d} \quad (5.16)$$

W celu uzyskania zbioru atomów semantycznych odpowiadających reprezentacji semantycznej modelu docelowego  $\mathcal{A}_{m \triangleleft \mathcal{M}^d}^{SCD}$ , należy zdekomponować cząsteczki semantyczne. Ponadto, uzyskuje się zbiór pojęć odpowiadający instancjom kategorii modelu docelowego, których semantyka opisana jest w ramach atomów semantycznych. Przekształcenie *ModelSemanticParticlesDecomposition* (wyr. 5.17) dokonuje takiej dekompozycji.

$$\mathcal{P}_{m \triangleleft \mathcal{M}^d} \xrightarrow{\text{ModelSemanticParticlesDecomposition}} (\mathcal{A}_{m \triangleleft \mathcal{M}^d}^{SCD}, \mathcal{C}_{m \triangleleft \mathcal{M}^d}) \quad (5.17)$$

### 5.1.4.4. Utworzenie modelu docelowego

Zbiór pojęć oraz atomów semantycznych modelu stanowi jego reprezentację semantyczną. Na jej podstawie modelujący może w sposób jednoznaczny utworzyć model (*ModelBuilding*, wyr. 5.18). Działanie to musi uwzględniać odwzorowanie kategorii metamodelu w pojęcia metamodelu oraz atomy semantyczne metamodelu, jednakże budowanie modelu odbywa się w sposób odwrotny, tj. konkretne atomy semantyczne modelu oraz konkretne pojęcia modelu odwzorowywane są w kategorii modelu. Przekształcenie to jest zawsze jednoznaczne, tj. odwzorowanie pojęć oraz atomów semantycznych w kategorii ma charakter izomorficzny.

$$(\mathcal{A}_{m \triangleleft \mathcal{M}^d}^{SCD}, \mathcal{C}_{m \triangleleft \mathcal{M}^d}) \xrightarrow{\text{ModelBuilding}} K_{m \triangleleft \mathcal{M}^d} \quad (5.18)$$

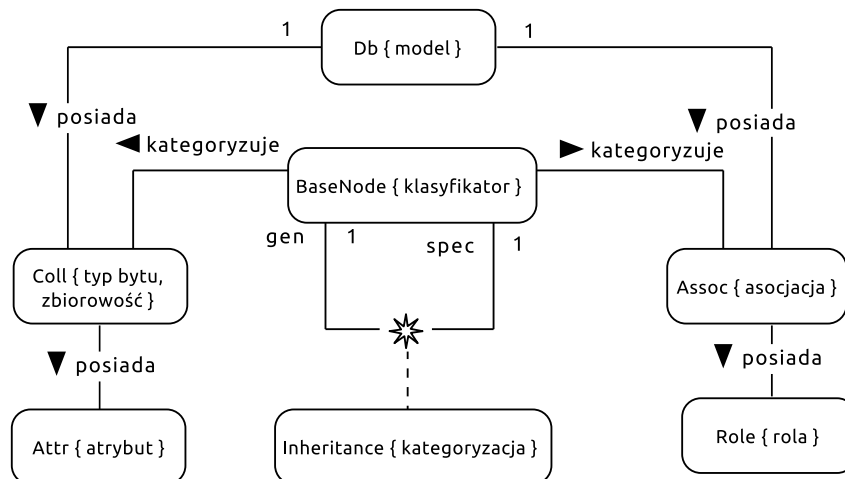


## 5.2. Ekstrakcja semantyki metamodelu asocjacyjnego

W ramach niniejszej sekcji dokonano ekstrakcji semantyki AOM do postaci pojęć opisujących jego kategorie oraz jego atomów semantycznych. Artefakty te posłużą do zdefiniowania dalszych przekształceń służących do odwzorowywania modeli asocjacyjnych do postaci modeli w wybranych w tej pracy metamodelach. W ramach niniejszej konceptualizacji uwzględniono jedynie część intensjonalną metamodelu asocjacyjnego, tj. kategorie uczestniczące w definiowaniu struktury modeli.

### 5.2.1. Pojęcia w ramach metamodelu

Na rysunku 5.2 przedstawiono diagram pojęć reprezentujących kategorie metamodelu asocjacyjnego.



Rysunek 5.2. Diagram pojęć przedstawiający uproszczoną konceptualizację metamodelu asocjacyjnego

#### 5.2.1.1. Pojęcia odpowiadające kategoriom części intensjonalnej metamodelu

##### aom:Db

Pojęcie ogólne: [model](#)

##### aom:BaseNode

Pojęcie ogólne: [klasyfikator](#)

##### aom:Coll

Pojęcie ogólne: [typ bytu](#)

##### aom:Assoc

Pojęcie ogólne: [asocjacja](#)

##### aom:Attr

Pojęcie ogólne: [atrybut](#)

## aom:Role

Pojęcie ogólne: [rola](#)

Pojęcie ogólne: [asocjacja](#)

## aom:Inheritance

Pojęcie ogólne: [kategoryzacja](#)

### 5.2.1.2. Kategorie pomocnicze

## aom:Composition

Typ pojęcia: [typ atomowy](#)

Definicja: [aom:inOwner](#) lub [aom:inDest](#) lub [aom:inBoth](#) lub [aom:none](#)

## aom:DataType

Typ pojęcia: [typ atomowy](#)

Pojęcie ogólne: [domena](#)

Definicja: [aom:int8](#) lub [aom:int16](#) lub [aom:int32](#) lub [aom:int64](#) lub [aom:float](#) lub [aom:double](#)  
lub [aom:bool](#) lub [aom:ascii](#) lub [aom:unicode](#) lub [aom:date](#) lub [aom:time](#) lub [aom:date](#)  
lub [aom:time](#) lub [aom:object](#) lub [aom:reference](#) lub [aom:byte](#) lub [aom:ascii-string](#) lub  
[aom:unicode-string](#) lub [aom:byte-array](#)

## aom:Direction

Typ pojęcia: [typ atomowy](#)

Definicja: [aom:toOwner](#) lub [aom:toDest](#) lub [aom:biDir](#)

## aom:Inheritability

Typ pojęcia: [typ atomowy](#)

Definicja: [aom:none](#) lub [aom:disable](#) lub [aom:nochange](#)

## aom:Navigability

Typ pojęcia: [typ atomowy](#)

Definicja: [aom:uniNav](#) lub [aom:biNav](#)

## aom:Virtuality

Typ pojęcia: [typ atomowy](#)

Definicja: [aom:natural](#) lub [aom:virtual](#) lub [aom:real](#)

### 5.2.2. Atomy semantyczne metamodelu

#### 5.2.2.1. Monadyczne atomy semantyczne

## aom:BaseNode może być abstrakcyjny

◇[1]  
AOM

Definicja: jeżeli [aom:BaseNode](#) jest abstrakcyjny, [aom:BaseNode](#) jest cechowany przez  
[nieinstancjonowalność](#)

## aom:BaseNode może być uninawigowalny

◇[2]  
AOM

Definicja: jeżeli aom:BaseNode jest uninawigowalny, cel każdej roli, którą pełni dany aom:BaseNode w asocjacji, jest nienawigowalny

### aom:Attr może być dziedziczny

◇[3]  
AOM

Definicja: jeżeli aom:Attr jest dziedziczny, aom:Attr jest cechowany przez dziedziczalność

### aom:Attr może być wirtualny

◇[4]  
AOM

Definicja: jeżeli dany aom:Attr jest wirtualny, to każdy aom:Attr należący do pozyskanych cech cechowany przez tę samą nazwę jest tym samym aom:Attr

Uwaga: W ramach metamodelu asocjacyjnego elementem nazwy składowych jest prefiks budowany na bazie informacji o ścieżce kategoryzacji oraz miejscu pozyskania wirtualności. Szczegółowo mechanizm ten został opisany w publikacji [87].

### aom:Role może być wirtualna

◇[5]  
AOM

Definicja: jeżeli dana aom:Role jest wirtualna, to każda aom:Role należąca do pozyskanych cech cechowana przez tę samą nazwę jest tą samą aom:Role

Uwaga: W ramach metamodelu asocjacyjnego elementem nazwy składowych jest prefiks budowany na bazie informacji o ścieżce kategoryzacji oraz miejscu pozyskania wirtualności. Szczegółowo mechanizm ten został opisany w publikacji [87].

### aom:Role może być dziedziczna

◇[6]  
AOM

Definicja: jeżeli aom:Role jest dziedziczna, to aom:Role jest cechowana przez dziedziczalność

### aom:Role może być uninawigowalna

◇[7]  
AOM

Definicja: jeżeli aom:Role jest uninawigowalna, to cel aom:Role jest nienawigowalny

### aom:Role może być skierowana do właściciela

◇[8]  
AOM

Definicja: jeżeli aom:Role jest skierowana do właściciela, to aom:Role jest cechowana przez semantyczną kierunkowość określoną jako cel→źródło

### aom:Role może być skierowana do celu

◇[9]  
AOM

Definicja: jeżeli aom:Role jest skierowana do celu, to aom:Role jest cechowana przez semantyczną kierunkowość określoną jako źródło→cel

### aom:Role może być dwukierunkowa

◇[10]  
AOM

Definicja: jeżeli aom:Role jest dwukierunkowa, to aom:Role jest cechowana przez semantyczną kierunkowość określoną jako dwukierunkowa

### aom:Role może mieć kompozycję po stronie związku

◇[11]  
AOM

Definicja: jeżeli aom:Role ma kompozycję po stronie związku, to źródło aom:role jest cechowane przez zależność czasu życia

### aom:Role może mieć kompozycję po stronie elementu wiązanego

◇[12]  
AOM

Definicja: jeżeli aom:Role ma kompozycję po stronie elementu wiązanego, to cel aom:Role jest cechowany przez zależność czasu życia

### 5.2.2.2. Poliadcyczne atomy semantyczne

**aom:BaseNode<sub>1</sub> może dziedziczyć po aom:BaseNode<sub>2</sub>**

⊙<sup>[13]</sup>  
AOM

Definicja: jeżeli aom:BaseNode<sub>1</sub> dziedziczy po aom:BaseNode<sub>2</sub>, to aom:BaseNode<sub>2</sub> *kategoryzuje* aom:BaseNode<sub>1</sub>

Definicja: aom:Inheritance

**aom:Inheritance jest trybu aom:Inheritability w aspekcie dziedziczenia składowych**

⊙<sup>[14]</sup>  
AOM

Konieczność: jeżeli aom:Inheritability to aom:none to aom:Inheritance nie zawiera dziedziczenia

Konieczność: jeżeli aom:Inheritability to aom:disable to pozyskane cechy są cechowane przez niedziedziczalność

Konieczność: jeżeli aom:Inheritability to aom:nochange to aom:Inheritance zawiera dziedziczenie

**aom:Inheritance jest trybu aom:Inheritability w aspekcie dziedziczenia praw do pełnienia ról**

⊙<sup>[15]</sup>  
AOM

Konieczność: jeżeli aom:Inheritability to aom:none to aom:Inheritance nie zawiera polimorfizmu

Konieczność: jeżeli aom:Inheritability to aom:disable to klasyfikatory kategoryzowane przez specjalizację nie mogą być podstawione za generalizację

Konieczność: jeżeli aom:Inheritability to aom:nochange to aom:Inheritance zawiera polimorfizm

**aom:Inheritance jest trybu aom:Virtuality w zakresie wirtualności**

⊙<sup>[16]</sup>  
AOM

Konieczność: jeżeli aom:Virtuality to aom:virtual to każda pozyskana cecha staje się wirtualna

Konieczność: jeżeli aom:Virtuality to aom:real to każda pozyskana cecha, która jest wirtualna przestaje być wirtualna

**aom:Coll może posiadać aom:Attr**

⊙<sup>[17]</sup>  
AOM

Konieczność: każdy aom:Attr musi być posiadany przez dokładnie jedną aom:Coll

**aom:Attr ma licznosc dodatniej liczby naturalnej**

⊙<sup>[18]</sup>  
AOM

Konieczność: jeżeli dodatnia liczba naturalna jest większa od 1 to aom:Attr jest cechowany przez wielowartościowość

**aom:Attr jest typu aom:DataType**

⊙<sup>[19]</sup>  
AOM

Definicja: aom:Attr jest cechowany przez domenę, którą jest aom:DataType

**aom:Attr ma wartość domyślną aom:Value**

⊙<sup>[20]</sup>  
AOM

Definicja: aom:Attr jest cechowany przez wartość domyślną, która jest określona jako aom:Value

**aom:Assoc może posiadać aom:Role**

⊙<sup>[21]</sup>  
AOM

Konieczność: każda aom:Role musi być posiadana przez dokładnie jedną aom:Assoc

**aom:Assoc może być opisywana przez aom:Coll**

⊙<sup>[22]</sup>  
AOM

Definicja: aom:Coll pełni rolę opisu w aom:Assoc

Konieczność: cel opisu jest cechowany przez krotność, która posiada kres górny równy 1

Konieczność: cel opisu jest nienawigowalny

### aom:BaseNode może uczestniczyć w aom:Role

$\diamond$ [23]  
AOM

Definicja: jeżeli aom:BaseNode uczestniczy w aom:Role, to aom:BaseNode pełni rolę aom:Role w aom:Assoc, która posiada daną aom:Role

### aom:Role może posiadać unikalność równą dodatniej liczbie naturalnej

[24]  
AOM

Definicja: jeżeli aom:Role posiada unikalność równą dodatniej liczbie naturalnej, to aom:Role jest cechowana przez powtarzalność, która jest określona przez dodatnią liczbę naturalną

### krotność aom:Role ma kres górny równy nieograniczonej liczbie naturalnej po stronie związku

[25]  
AOM

Definicja: jeżeli nieograniczona liczba naturalna jest liczbą naturalną to źródło aom:Role jest cechowane przez krotność, która posiada kres górny jej równy

### krotność aom:Role ma kres dolny równy liczbie naturalnej po stronie związku

[26]  
AOM

Definicja: źródło aom:Role jest cechowane przez krotność, która posiada kres dolny równy liczbie naturalnej

### krotność aom:Role ma kres górny równy nieograniczonej liczbie naturalnej po stronie elementu wiążanego

[27]  
AOM

Definicja: jeżeli nieograniczona liczba naturalna jest liczbą naturalną to cel aom:Role jest cechowany przez krotność, która posiada kres górny jej równy

### krotność aom:Role ma kres dolny równy liczbie naturalnej po stronie elementu wiążanego

[28]  
AOM

Definicja: cel aom:Role jest cechowany przez krotność, która posiada kres dolny równy liczbie naturalnej

### aom:Role może być opisywana przez aom:Coll

$\diamond$ [29]  
AOM

Definicja: aom:Coll pełni rolę opisu roli w aom:Role

Konieczność: cel opisu roli jest cechowany przez krotność, która posiada kres górny równy 1

## 5.2.3. Budowa cząsteczek semantycznych metamodelu asocjacyjnego

Poniżej przedstawiono cząsteczki proste AOM, zapewniające pełne pokrycie wszystkich atomów semantycznych metamodelu.

$$*_{AOM}^{[kolekcja]} = \left( \begin{array}{c} *_{AOM}^{\{\diamond 1, \diamond 2\}}, \\ \emptyset, \\ \left\{ \odot_{[aom:Coll]} < \odot_{[aom:BaseNode]} \right\} \end{array} \right) \quad (5.19)$$

$$*_{AOM}^{[asocjacja]} = \left( \begin{array}{c} *_{AOM}^{\{\diamond 1, \diamond 2\}}, \\ \emptyset, \\ \left\{ \odot_{[aom:Assoc]} < \odot_{[aom:BaseNode]} \right\} \end{array} \right) \quad (5.20)$$

$$*_{AOM}^{[atrybut]} = \left( \begin{array}{c} *_{AOM}^{\{\diamond 3, \diamond 4, \diamond 17, 18, 19, 20\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot_{[aom:Attr]}, \\ \odot_{[aom:Coll]}, \\ \odot_{[aom:Value]}, \\ \odot_{[aom:DataType]}, \\ \odot_{[krotność\ atrybutu]} < \left[ *_{AOM}^{[18]} \odot_{[N^+]} \right] \end{array} \right\} \end{array} \right) \quad (5.21)$$

$$*_{AOM}^{[rola]} = \left( \begin{array}{c} *_{AOM}^{\{\diamond 5, \diamond 6, \diamond 7, \diamond 8, \diamond 9, \diamond 10, \diamond 11, \diamond 12, 25, 26, 27, 28, \diamond 21, \diamond 23, \diamond 24\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot_{[aom:Role]}, \odot_{[aom:Assoc]}, \\ \odot_{[końcówka\ roli]} < \left[ *_{AOM}^{[23]} \odot_{[aom:BaseNode]} \right], \\ \odot_{[supremum\ krotności\ po\ str.\ związku]} < \left[ *_{AOM}^{[25]} \odot_{[N^*]} \right], \\ \odot_{[infimum\ krotności\ po\ str.\ związku]} < \left[ *_{AOM}^{[26]} \odot_{[N]} \right], \\ \odot_{[supremum\ krotności\ po\ str.\ el.\ wiązanego]} < \left[ *_{AOM}^{[27]} \odot_{[N^*]} \right], \\ \odot_{[infimum\ krotności\ po\ str.\ el.\ wiązanego]} < \left[ *_{AOM}^{[28]} \odot_{[N]} \right] \end{array} \right\} \end{array} \right) \quad (5.22)$$

$$*_{AOM}^{[opis\ asocjacji]} = \left( \begin{array}{c} *_{AOM}^{\{\diamond 22\}}, \\ \emptyset, \\ \left\{ \odot_{[aom:Assoc]}, \odot_{[aom:Coll]} \right\} \end{array} \right) \quad (5.23)$$

$$*_{AOM}^{[opis\ roli]} = \left( \begin{array}{c} *_{AOM}^{\{\diamond 29\}}, \\ \emptyset, \\ \left\{ \odot_{[aom:Role]}, \odot_{[aom:Coll]} \right\} \end{array} \right) \quad (5.24)$$

$$\begin{aligned}
\begin{matrix} * \\ \text{AOM} \end{matrix}^{[\text{dziedziczenie}]} = & \left( \begin{array}{c} \begin{matrix} * \\ \text{AOM} \end{matrix}^{\{\diamond 13,14,15,16\}} \\ \emptyset, \\ \left\{ \begin{array}{l} \odot_{[\text{aom:BaseNode}_1]}, \\ \odot_{[\text{aom:BaseNode}_2]}, \\ \odot_{[\text{aom:Inheritance}]}, \\ \odot_{[\text{aom:Virtuality}]}, \\ \odot_{[\text{aspekt dziedziczenia składowych}] \triangleleft [\begin{matrix} * \\ \text{AOM} \end{matrix}]^{[14]} \odot_{[\text{aom:Inheritability}]}, \\ \odot_{[\text{aspekt dziedziczenia praw do peł. ról}] \triangleleft [\begin{matrix} * \\ \text{AOM} \end{matrix}]^{[15]} \odot_{[\text{aom:Inheritability}]} \end{array} \right\} \end{array} \right) \quad (5.25)
\end{aligned}$$

### 5.2.3.1. Analiza i budowa części semantycznej opartej o wzorzec modelowania *BACT*

Aby określić budowę cząsteczki semantycznej opisującej wzorzec modelowania *BACT*, dokonano analizy wszystkich koniecznych atomów semantycznych tego wzorca. Atom semantyczny K.1 (wyr. 5.26) determinuje konieczność wyodrębnienia samego wzorca jako koncepcji. Oznacza to, iż atom ten semantycznie determinuje utworzenie nowej cząsteczki dla rozważanego metamodelu. Pozostałe atomy semantyczne determinują wykorzystanie pewnych atomów semantycznych lub konkretyzacji odnoszące się do określenia lub wyspecyfikowania ról uczestników wzorca.

$$\begin{matrix} * \\ \text{[BACT]} \end{matrix}^{K.1} \stackrel{\text{AOM}}{\models} \left\{ \begin{matrix} * \\ \text{AOM} \end{matrix}^{[\text{BACT}]} \right\} \quad (5.26)$$

$$\begin{matrix} * \\ \text{[BACT]} \end{matrix}^{K.2} \stackrel{\text{AOM}}{\models} \left\{ \odot_{[\text{aspekt danych}] \triangleleft \odot_{[\text{aom:Coll}]} \right\} \quad (5.27)$$

$$\begin{matrix} * \\ \text{[BACT]} \end{matrix}^{K.3} \stackrel{\text{AOM}}{\models} \left\{ \odot_{[\text{aspekt związku}] \triangleleft \odot_{[\text{aom:Assoc}]} \right\} \quad (5.28)$$

$$\begin{matrix} * \\ \text{[BACT]} \end{matrix}^{K.4} \stackrel{\text{AOM}}{\models} \left\{ \odot_{[\text{nierozzerwalne połączenie}] \triangleleft \odot_{[\text{aom:Role}]} \right\} \quad (5.29)$$

$$\begin{matrix} * \\ \text{[BACT]} \end{matrix}^{K.4.1} \stackrel{\text{AOM}}{\models} \left( \begin{array}{c} \begin{matrix} * \\ \text{AOM} \end{matrix}^{\{\diamond 21\}} \\ \odot_{[\text{aspekt związku}] \triangleleft \odot_{[\text{aom:Assoc}]}, \\ \odot_{[\text{nierozzerwalne połączenie}] \triangleleft \odot_{[\text{aom:Role}]} \end{array} \right) \quad (5.30)$$

$$\begin{matrix} *_{[BACT]}^{K.4.2} \text{ AOM} \\ \models \end{matrix} \left\{ \begin{array}{l} *_{\text{AOM}}^{\{\diamond 23\}}, \\ \odot [\text{aspekt danych}] \triangleleft \odot [\text{aom:BaseNode}], \\ \odot [\text{nierozzerwalne po\u0142\u0105czenie}] \triangleleft \odot [\text{aom:Role}] \end{array} \right\} \quad (5.31)$$

$$\begin{matrix} *_{[BACT]}^{K.4.2.1} \text{ AOM} \\ \models \end{matrix} \left\{ \begin{array}{l} *_{\text{AOM}}^{\{\diamond 11, \diamond 12\}}, \\ \odot [\text{nierozzerwalne po\u0142\u0105czenie}] \triangleleft \odot [\text{aom:Role}] \end{array} \right\} \quad (5.32)$$

$$\begin{matrix} *_{[BACT]}^{K.4.2.2} \text{ AOM} \\ \models \end{matrix} \left\{ \begin{array}{l} *_{\text{AOM}}^{\{25,26,27,28\}}, \\ \odot [\underline{1}] \triangleleft \odot [\mathbb{N}^*], \\ \odot [\text{nierozzerwalne po\u0142\u0105czenie}] \triangleleft \odot [\text{aom:Role}] \end{array} \right\} \quad (5.33)$$

Analizuj\u0105c powy\u017aszsze konsekwencje semantyczne atom\u00f3w semantycznych wzorca dla metamodelu asocjacyjnego, okre\u015blono cz\u0105steczk\u0119 semantyczn\u0105 wzorca BACT jako:

$$*_{\text{AOM}}^{[BACT]} = \left( \begin{array}{l} *_{\text{AOM}}^{\{\diamond 11, \diamond 12, 25, 26, 27, 28, \diamond 21, \diamond 23\}}, \\ \emptyset, \\ \odot [\text{aspekt danych}] \triangleleft \odot [\text{aom:Coll}], \\ \odot [\text{aspekt zwi\u0105zku}] \triangleleft \odot [\text{aom:Assoc}], \\ \odot [\text{nierozzerwalne po\u0142\u0105czenie}] \triangleleft \odot [\text{aom:Role}], \\ \odot [\text{aspekt danych}] \triangleleft [\text{AOM}^{[23]}] \odot [\text{ko\u0144cówka roli}], \\ \odot [\underline{1}] \triangleleft \odot [\mathbb{N}^*] \end{array} \right) \quad (5.34)$$

### 5.3. Translacja semantyki AOM na EER

W ramach niniejszej sekcji zosta\u0142a opisana translacja semantyki metamodelu asocjacyjnego na semantyk\u0119 przyj\u0119tego rozszerzonego metamodelu zwi\u0105zk\u00f3w-encji. Specyfikacja semantyki zosta\u0142a przeprowadzona poprzez analiz\u0119 tego metamodelu m.in. na bazie nast\u0119puj\u0105cych publikacji: [19, 33, 41, 88, 141, 144].

#### 5.3.1. Specyfikacja fragmentu semantyki metamodelu zwi\u0105zk\u00f3w-encji

##### 5.3.1.1. Poj\u0119cia przyj\u0119tego metamodelu zwi\u0105zk\u00f3w-encji

###### eer:Element

Poj\u0119cie ogólne: [klasyfikator](#)

###### eer:EntitySet

Poj\u0119cie ogólne: [eer:Element](#)



Pojęcie ogólne: [typ bytu](#)

### [eer:RelationshipSet](#)

Pojęcie ogólne: [eer:Element](#)

Pojęcie ogólne: [asocjacja](#)

### [eer:Attribute](#)

Pojęcie ogólne: [atrybut](#)

### [eer:ValueSet](#)

Pojęcie ogólne: [domena](#)

### [eer:Role](#)

Pojęcie ogólne: [rola](#)

### [eer:Generalization/Specialization](#)

Pojęcie ogólne: [kategoryzacja](#)

#### 5.3.1.2. Typy pomocnicze

### [eer:CardinalityKind](#)

Typ pojęcia: [typ atomowy](#)

Definicja: [eer:one](#) lub [eer:many](#)

### [eer:ParticipationKind](#)

Typ pojęcia: [typ atomowy](#)

Definicja: [eer:partial](#) lub [eer:total](#)

### [eer:DisjointnessKind](#)

Typ pojęcia: [typ atomowy](#)

Definicja: [eer:disjoint](#) lub [eer:overlapping](#)

#### 5.3.1.3. Atomy semantyczne przyjętego metamodelu związków-encji

##### 5.3.1.3.1. Monadyczne atomy semantyczne

### [eer:EntitySet](#) może być słaby

 [1]  
EER

Definicja: jeżeli [eer:EntitySet](#) jest słaby, [eer:EntitySet](#) nie jest cechowany przez [identyfikator](#)

Konieczność: każdy [eer:EntitySet](#), który jest słaby musi uczestniczyć w co najmniej jednym [eer:RelationshipSet](#)

### [eer:RelationshipSet](#) może być identyfikujący

 [2]  
EER

Definicja: jeżeli [eer:RelationshipSet](#) jest identyfikujący to [eer:RelationshipSet](#) jest cechowany przez [identyfikację](#)

### [eer:Attribute](#) może być wielowartościowy

 [3]  
EER

Definicja: jeżeli [eer:Attribute](#) jest wielowartościowy to [eer:Attribute](#) jest cechowany przez [wielowartościowość](#)

### eer:Attribute może przyjmować wartości null

◇[4]  
EER

Definicja: jeżeli eer:Attribute przyjmuje wartości null to eer:Attribute jest cechowany przez zerowalność

### eer:Attribute może być unikalny

◇[5]  
EER

Definicja: jeżeli eer:Attribute jest unikalny to eer:Attribute jest cechowany przez unikalność

### eer:Attribute może być złożony

◇[6]  
EER

Definicja: jeżeli eer:Attribute jest złożony to eer:Attribute jest cechowany przez strukturalną złożoność

### eer:Attribute może być pochodny

◇[7]  
EER

Definicja: jeżeli eer:Attribute jest pochodny, eer:Attribute jest cechowany przez pochodność

### eer:Role może być cechowana przez zależność istnienia

◇[8]  
EER

Definicja: jeżeli eer:Role jest cechowana przez zależność istnienia to eer:Role jest cechowana przez zależność istnienia

#### 5.3.1.3.2. Poliadyczne atomy semantyczne

### eer:Element może posiadać eer:Attribute

◇[9]  
EER

Konieczność: każdy eer:Attribute musi być posiadany przez dokładnie jeden eer:Element

### eer:Attribute ma określony eer:ValueSet

[10]  
EER

Definicja: eer:Attribute jest cechowany przez domenę, którą jest eer:ValueSet

### eer:Attribute może wchodzić w skład klucza eer:EntitySet

◇[11]  
EER

Definicja: jeżeli eer:Attribute wchodzi w skład klucza eer:EntitySet to eer:Attribute jest cechowany przez identyfikację

### eer:Attribute może posiadać kres górny równy dodatniej liczbie naturalnej

◇[12]  
EER

Definicja: jeżeli eer:Attribute posiada kres górny równy dodatniej liczbie naturalnej to eer:Attribute jest cechowany przez kres górny określony jako ta dodatnia liczba naturalna

### eer:Attribute może posiadać kres dolny równy nieujemnej liczbie naturalnej

◇[13]  
EER

Definicja: jeżeli eer:Attribute posiada kres dolny równy nieujemnej liczbie naturalnej to eer:Attribute jest cechowany przez kres dolny określony jako ta nieujemna liczba naturalna

### eer:Attribute<sub>1</sub> może składać się z eer:Attribute<sub>2</sub>

◇[14]  
EER

Definicja: jeżeli eer:Attribute<sub>1</sub> składa się z eer:Attribute<sub>2</sub> to eer:Attribute<sub>1</sub> agreguje eer:Attribute<sub>2</sub>

### eer:RelationshipSet *posiada* eer:Role

 [15]  
EER

Konieczność: eer:RelationshipSet musi *posiadać* co najmniej 2 instancje eer:Role

### eer:Role *posiada* eer:ParticipationKind

 [16]  
EER

Konieczność: jeżeli eer:ParticipationKind to eer:total to eer:Role jest *cechowana przez* krotność, która *posiada* kres dolny równy 1

### eer:Role *posiada* eer:CardinalityKind

 [17]  
EER

Konieczność: jeżeli eer:CardinalityKind to eer:one to eer:Role jest *cechowana przez* krotność, która *posiada* kres górny równy 1

### eer:EntitySet *może pełnić* eer:Role

 [18]  
EER

Definicja: jeżeli eer:EntitySet *pełni* eer:Role to eer:EntitySet *pełni* eer:Role w eer:RelationshipSet, który *posiada* daną eer:Role

### eer:EntitySet *może pełnić* rolę *generalizacji* w

### eer:Generalization/Specialization

 [19]  
EER

Definicja: eer:Generalization/Specialization *posiada* generalizację, którą *jest* eer:EntitySet

Konieczność: co najwyżej jeden eer:EntitySet *pełni* rolę *generalizacji* w danym eer:Generalization/Specialization

### eer:EntitySet *może pełnić* rolę *specjalizacji* w eer:Generalization/Specialization

 [20]  
EER

Definicja: eer:Generalization/Specialization *posiada* specjalizację, którą *jest* eer:EntitySet

### eer:Generalization/Specialization *ma określony* eer:DisjointnessKind

 [21]  
EER


Konieczność: jeżeli eer:DisjointnessKind to eer:disjoint to eer:Generalization/Specialization jest *cechowana przez* rozłączność

### eer:Generalization/Specialization *ma określony* eer:ParticipationKind

 [22]  
EER

Konieczność: jeżeli eer:ParticipationKind to eer:total to eer:Generalization/Specialization jest *cechowana przez* pokrycie

### eer:EntitySet *może agregować* eer:Element

 [23]  
EER

Definicja: jeżeli eer:EntitySet *agreguje* eer:Element to eer:EntitySet *posiada* eer:Element

## 5.3.2. Częsteczki semantyczne przyjętego metamodelu związków-encji

$$*_{\text{EER}}^{\text{[zbiór encji]}} = \left( \begin{array}{c} *_{\text{EER}}^{\{ \diamond 1, \}} \\ \emptyset, \\ \left\{ \odot_{\text{[eer:EntitySet]}} \right\} \end{array} \right) \quad (5.35)$$

$$*_{\text{EER}}[\text{zbiór związków}] = \left( \begin{array}{c} *_{\text{EER}}\{\diamond 2, \} \\ \emptyset, \\ \left\{ \odot[\underline{\text{eer:RelationshipSet}}] \right\} \end{array} \right) \quad (5.36)$$

$$*_{\text{EER}}[\text{atrybut zbioru encji}] = \left( \begin{array}{c} *_{\text{EER}}\{\diamond 3, \diamond 4, \diamond 5, \diamond 7, \diamond 9, 10, \diamond 11, \diamond 12, \diamond 13 \}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot[\underline{\text{eer:Attribute}}] \\ \odot[\underline{\text{eer:EntitySet}}] < \odot[\underline{\text{eer:Element}}] \\ \odot[\underline{\text{eer:ValueSet}}] \\ \diamond \odot[\underline{\text{kres g\u00f3rny atrybutu}}] \triangleleft \odot[\underline{\text{dodatnia liczba naturalna}}], \\ \diamond \odot[\underline{\text{kres dolny atrybutu}}] \triangleleft \odot[\underline{\text{liczba naturalna}}] \end{array} \right\} \end{array} \right) \quad (5.37)$$

$$*_{\text{EER}}[\text{atrybut zbioru zwi\u0105zku}] = \left( \begin{array}{c} *_{\text{EER}}\{\diamond 3, \diamond 4, \diamond 5, \diamond 7, 9, 10, \diamond 11, \diamond 12, \diamond 13 \}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot[\underline{\text{eer:Attribute}}], \\ \odot[\underline{\text{eer:RelationshipSet}}] < \odot[\underline{\text{eer:Element}}], \\ \odot[\underline{\text{eer:ValueSet}}], \\ \diamond \odot[\underline{\text{kres g\u00f3rny atrybutu}}] \triangleleft \odot[\underline{\text{dodatnia liczba naturalna}}], \\ \diamond \odot[\underline{\text{kres dolny atrybutu}}] \triangleleft \odot[\underline{\text{liczba naturalna}}] \end{array} \right\} \end{array} \right) \quad (5.38)$$

$$*_{\text{EER}}[\text{atrybut z\u0142o\u017cy}] = \left( \begin{array}{c} *_{\text{EER}}\{\diamond 6, 14 \}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot[\underline{\text{atrybut nadrz\u0119dny}}] \triangleleft [ *_{\text{EER}}^{[6]} ] \odot[\underline{\text{eer:Attribute}}] \\ \odot[\underline{\text{atrybut nadrz\u0119dny}}] \triangleleft [ *_{\text{EER}}^{[14]} ] \odot[\underline{\text{eer:Attribute}}_1] \\ \odot[\underline{\text{atrybut podrz\u0119dny}}] \triangleleft [ *_{\text{EER}}^{[14]} ] \odot[\underline{\text{eer:Attribute}}_2] \end{array} \right\} \end{array} \right) \quad (5.39)$$

$$*_{\text{EER}}^{\text{[rola związku]}} = \left( \begin{array}{c} *_{\text{EER}}^{\{\diamond 8,15,16,17,\diamond 18\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot \underline{\text{[eer:Role]}}, \\ \odot \underline{\text{[eer:RelationshipSet]}}, \\ \odot \underline{\text{[eer:EntitySet]}}, \\ \odot \underline{\text{[eer:CardinalityKind]}}, \\ \odot \underline{\text{[eer:ParticipationKind]}} \end{array} \right\} \end{array} \right) \quad (5.40)$$




$$*_{\text{EER}}^{\text{[generalizacja-specjalizacja]}} = \left( \begin{array}{c} *_{\text{EER}}^{\{\diamond 19,\diamond 20,21,22\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot \underline{\text{[eer:Generalization/Specialization]}} \\ \odot \underline{\text{[generalizacja]}} \triangleleft \left[ *_{\text{EER}}^{[19]} \right] \odot \underline{\text{[eer:EntitySet]}}, \\ \odot \underline{\text{[specjalizacja]}} \triangleleft \left[ *_{\text{EER}}^{[20]} \right] \odot \underline{\text{[eer:EntitySet]}}, \\ \odot \underline{\text{[eer:DisjointnessKind]}}, \\ \odot \underline{\text{[eer:ParticipationKind]}} \end{array} \right\} \end{array} \right) \quad (5.41)$$




$$*_{\text{EER}}^{\text{[agregacja]}} = \left( \begin{array}{c} *_{\text{EER}}^{\{\diamond 23\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot \underline{\text{[eer:EntitySet]}} \\ \odot \underline{\text{[eer:Element]}} \end{array} \right\} \end{array} \right) \quad (5.42)$$

### 5.3.3. Wzorce mapowania

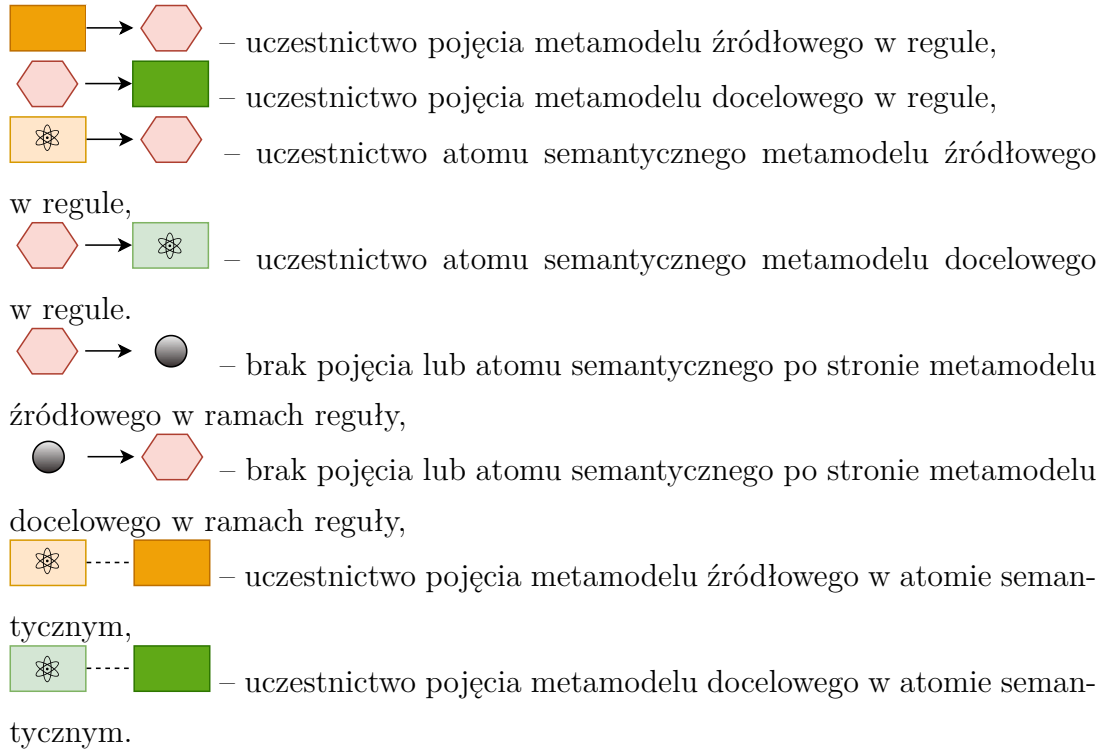
W ramach niniejszej sekcji przedstawiono wzorce mapowania dla wszystkich zidentyfikowanych częściłek semantycznych AOM. Wzorce mapowania składają się z szeregu reguł, zdefiniowanych z wykorzystaniem systemu pojęć CLoM oraz wyekstrahowanych specyfikacji semantyki metamodeli. Pomocniczym elementem prezentującym te reguły są diagramy obrazujące odwzorowania.

Węzły na diagramach należy czytać w następujący sposób:

- symbol  obrazuje konkretną regułę w ramach wzorca mapowania,
- symbol  oznacza pojęcie metamodelu źródłowego,
- symbol  oznacza pojęcie metamodelu docelowego,

- symbol  oznacza atom semantyczny metamodelu źródłowego,
- symbol  oznacza atom semantyczny metamodelu docelowego,
- symbol  oznacza brak pojęcia lub atomu semantycznego.

Możliwe związki pomiędzy węzłami:



### 5.3.3.1. Wzorzec mapowania: kolekcja

Wzorzec mapowania ma postać odwzorowania:

$$*_{\text{AOM}}^{[\text{kolekcja}]} \mapsto *_{\text{EER}_{\text{AOM}}}^{[\text{kolekcja}]} \quad (5.43)$$

gdzie:

$$*_{\text{EER}_{\text{AOM}}}^{[\text{kolekcja}]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{\text{EER}}^{[\text{zbiór encji}]} \left[ \begin{array}{c} \diamond 1 \\ *_{\text{EER}} \end{array} \right] \right\}, \\ \left\{ \ominus_{\text{[eer:EntitySet]}} \right\} \end{array} \right) \quad (5.44)$$

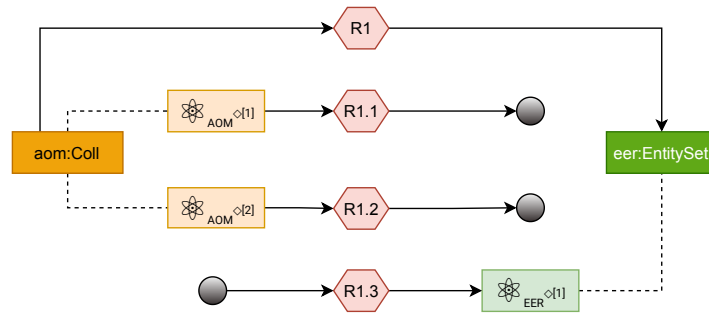
Reguły wzorca mapowania (rys. 5.3):

R1 jest konieczne aby aom:Coll odpowiadało eer:EntitySet

R1.1 jest niemożliwe, aby to, że aom:Coll jest abstrakcyjna, posiadało odpowiednik w EER

R1.2 jest niemożliwe, aby to, że aom:Coll jest uninawigowalna, posiadało odpowiednik w EER

R1.3 jest obowiązkowe, aby eer:EntitySet, który odpowiada aom:Coll nie był słaby



Rysunek 5.3. Diagram obrazujący odwzorowanie  $*_{AOM}^{[kolekcja]} \mapsto *_{EER_{AOM}}^{[kolekcja]}$

### 5.3.3.2. Wzorzec mapowania: atrybut

Wzorzec mapowania ma postać odwzorowania:

$$*_{AOM}^{[atrybut]} \mapsto *_{EER_{AOM}}^{[atrybut]} \quad (5.45)$$

gdzie:

$$*_{EER_{AOM}}^{[atrybut]} = \left( \begin{array}{l} \emptyset, \\ \left\{ *_{EER}^{[atrybut\ zbioru\ encji]} \left[ *_{EER}^{\diamond 4, \diamond 7, \diamond 5, \diamond 11} \right], \right. \\ \left. \diamond *_{EER}^{[atrybut\ złożony]} \right\}, \\ \left. \begin{array}{l} \odot [eer:Attribute], \\ \odot [eer:EntitySet], \\ \odot [eer:ValueSet], \\ \diamond \odot [dodatnia\ liczba\ naturalna], \\ \diamond \odot [liczba\ naturalna], \\ \diamond \odot [eer:Attribute] \triangleleft \odot [atrybut\ nadrzędny], \\ \diamond \odot [atrybut\ podrzędny] \end{array} \right\} \end{array} \right) \quad (5.46)$$

Reguły wzorca mapowania (rys. 5.4):

R1 jest konieczne aby aom:Attr odpowiadało eer:Attribute

R1.1 jest niemożliwe, aby to, że aom:Attr jest dziedziczny, posiadało odpowiednik w EER

R1.2 jest niemożliwe, aby to, że aom:Attr jest wirtualny, posiadało odpowiednik w EER

- R1.3 jest obowiązkowe, aby aom:Attr, który odpowiada aom:Attr nie przyjmował wartości null
- R1.4 jest obowiązkowe, aby eer:Attribute, który odpowiada aom:Attr nie był unikalny
- R1.5 jest obowiązkowe, aby eer:Attribute, który odpowiada aom:Attr nie był pochodny
- R1.6 jest obowiązkowe, aby aom:Attribute, który odpowiada aom:Attr nie wchodził w skład klucza eer:EntitySet
- R2 jest konieczne aom:Coll odpowiadało eer:Element
- R2.1 jeżeli aom:Coll posiada aom:Attr to odpowiadający eer:Element posiada odpowiadający eer:Attribute
- R3 jest konieczne aby dodatnia liczba naturalna będąca licznikiem aom:Attr odpowiadała dodatniej liczbie naturalnej będącej kresem górnym eer:Attribute oraz liczbie naturalnej będącej kresem dolnym eer:Attribute
- R3.1 jeżeli aom:Attr ma licznik, która jest większa od 1 to odpowiadający eer:Attribute jest wielowartościowy
- R4 jest konieczne aby aom:DataType odpowiadało eer:ValueSet
- R4.1 eer:Attribute musi posiadać eer:ValueSet odpowiadający aom:DataType będącego typem dla aom:Attr
- R4.2 jeżeli aom:Attr jest typu aom:object to eer:Attribute jest złożony oraz składa się z eer:Attributes odpowiadających aom:Attr posiadany przez aom:Coll posiadającej aom:Value
- R5 jest niemożliwe, aby to, że aom:Attr ma wartość domyślną aom:Value, posiadało odpowiednik w EER

### 5.3.3.3. Wzorzec mapowania: asocjacja

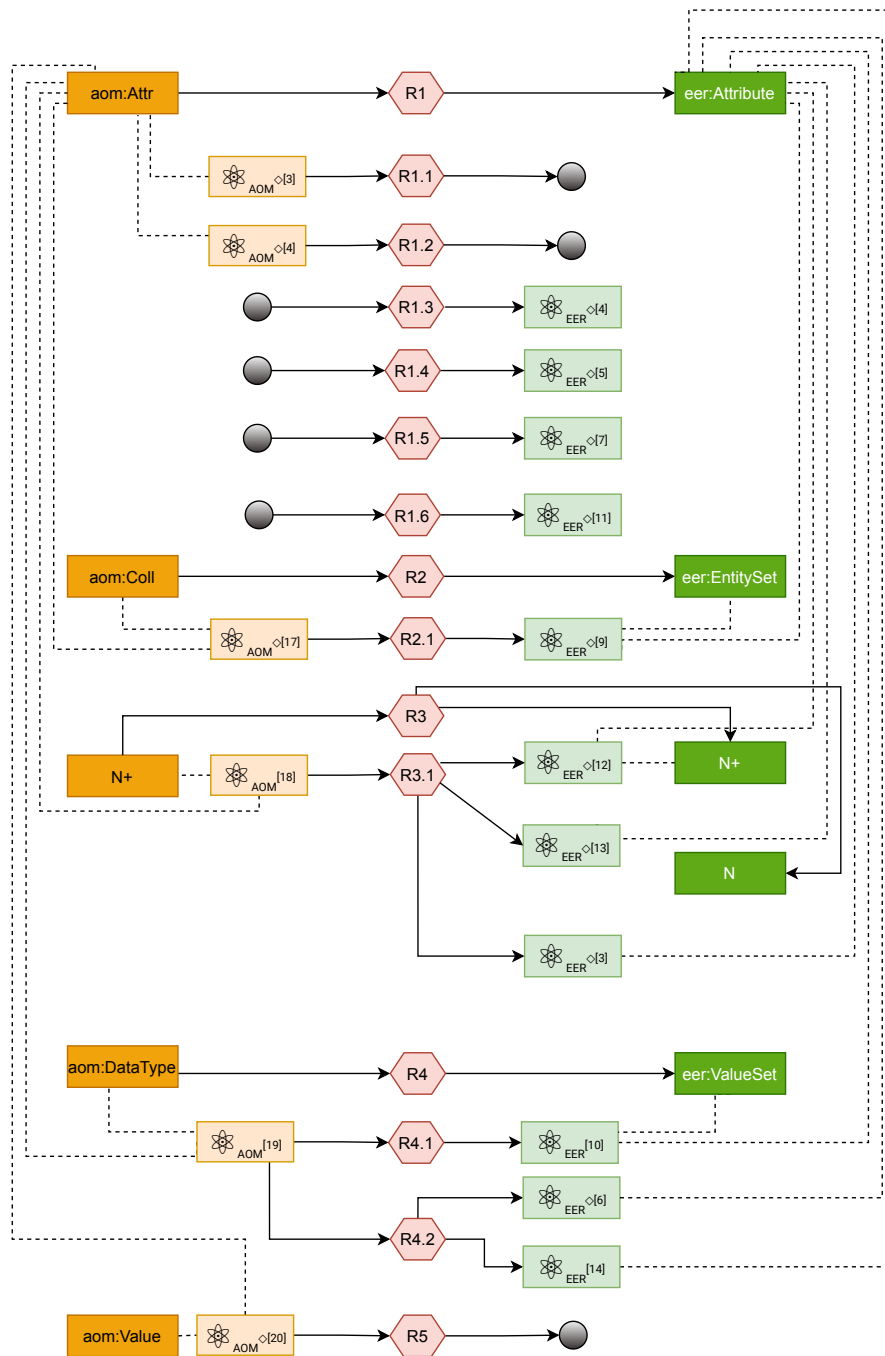
Wzorzec mapowania ma postać odwzorowania:

$$*_{AOM}^{[asocjacja]} \mapsto *_{EER_{AOM}}^{[asocjacja]} \quad (5.47)$$

gdzie:

$$*_{EER_{AOM}}^{[asocjacja]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{EER}^{[zbiór encji | zbiór związków]} \left[ *_{EER}^{\diamond 1 | \diamond 2} \right] \right\}, \\ \left\{ \begin{array}{c} \odot_{[eer:EntitySet]} \\ \odot_{[eer:RelationshipSet]} \end{array} \right\} \end{array} \right) \quad (5.48)$$





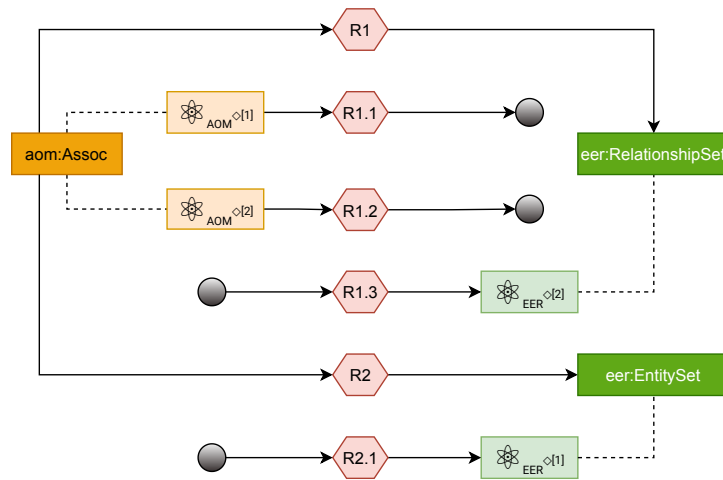
Rysunek 5.4. Diagram obrazujący odwzorowanie  $*_{AOM}^{[atrybut]} \mapsto *_{EER_{AOM}}^{[atrybut]}$

Reguły wzorca mapowania (rys. 5.5):

R1 jeżeli aom:Assoc posiada co najmniej 2 aom:Role lub jest opisywana przez aom:Coll oraz jej generalizacja nie odpowiada aom:Coll to odpowiada jej eer:RelationshipSet

R1.1 jest niemożliwe, aby to, że aom:Assoc jest abstrakcyjna, posiadało odpowiednik w EER

- R1.2 jest niemożliwe, aby to, że aom:Assoc jest uninawigowalna, posiadało odpowiednik w EER
- R1.3 jest obowiązkowe, aby eer:RelationshipSet, który odpowiada aom:Assoc nie był identyfikujący
- R2 jeżeli aom:Assoc posiada co najwyżej 2 aom:Role lub nie jest opisywana przez aom:Coll oraz jej generalizacja odpowiada aom:Coll to odpowiada jej eer:EntitySet
- R2.1 jest obowiązkowe, aby eer:EntitySet, który odpowiada aom:Assoc nie był słaby



Rysunek 5.5. Diagram obrazujący odwzorowanie  $*_{AOM}^{[asocjacja]} \mapsto *_{EER_{AOM}}^{[asocjacja]}$

#### 5.3.3.4. Wzorzec mapowania: rola

Wzorzec mapowania ma postać odwzorowania:

$$*_{AOM}^{[rola]} \mapsto *_{EER_{AOM}}^{[rola]} \quad (5.49)$$

gdzie:

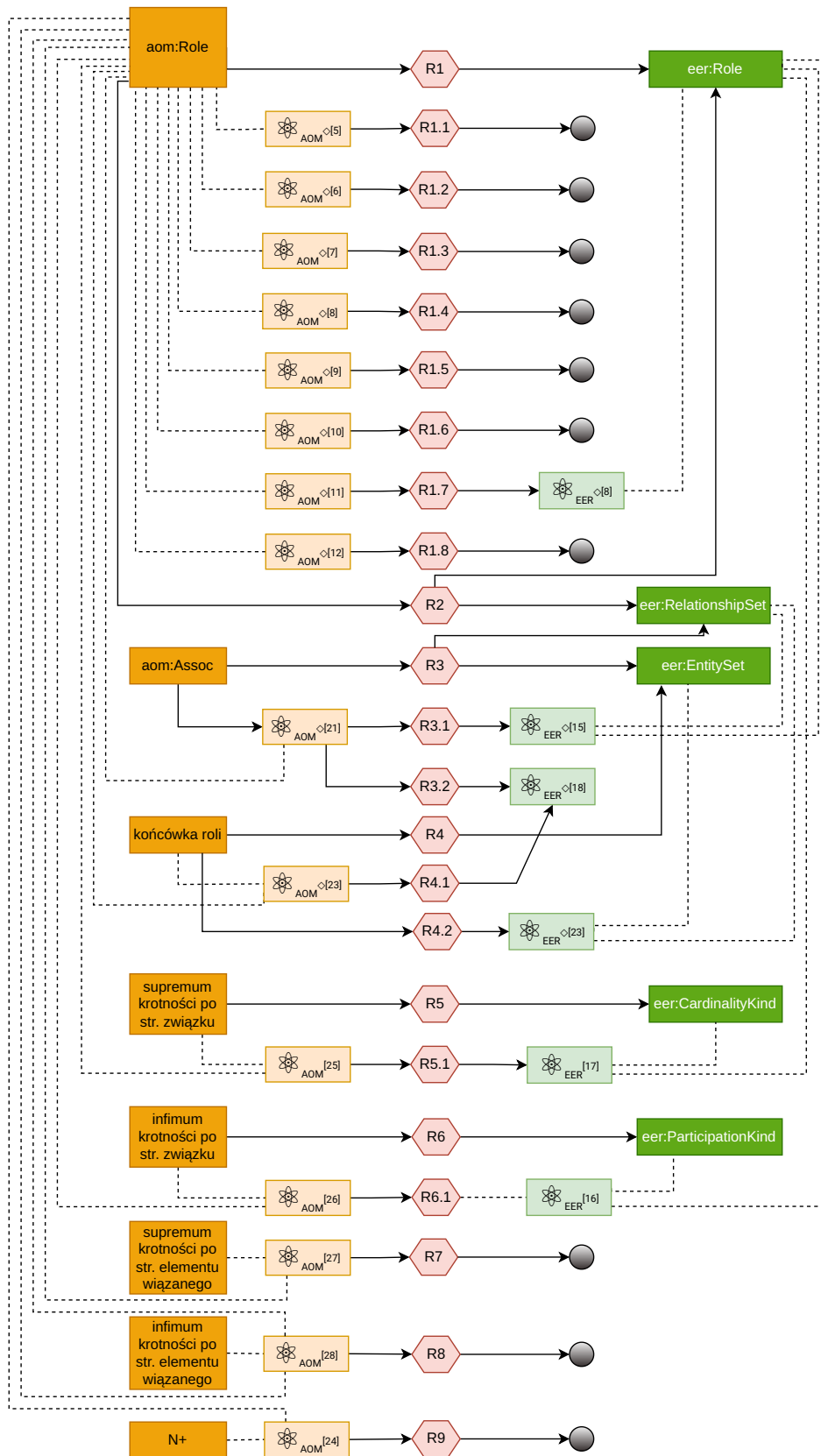
$$*_{EER_{AOM}}^{[rola]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{EER}^{[rola \text{ związku} \mid \text{zbiór związków}]}, \right. \\ \left. \left\{ \begin{array}{l} \odot_{[eer:Role]} \\ \odot_{[eer:EntitySet]} \\ \odot_{[eer:RelationshipSet]} \end{array} \right\} \right\} \right) \quad (5.50)$$

Reguły wzorca mapowania (rys. 5.6):

- R1 jeżeli aom:Assoc posiadająca aom:Role posiada więcej niż 2 aom:Role to aom:Role odpowiada eer:Role

- R1.1 jest niemożliwe, aby to, że aom:Role jest wirtualna, posiadało odpowiednik w EER
- R1.2 jest niemożliwe, aby to, że aom:Role jest dziedziczalna, posiadało odpowiednik w EER
- R1.3 jest niemożliwe, aby to, że aom:Role jest uninawigowalna, posiadało odpowiednik w EER
- R1.4 jest niemożliwe, aby to, że aom:Role jest skierowana do właściciela, posiadało odpowiednik w EER
- R1.5 jest niemożliwe, aby to, że aom:Role jest skierowana do celu, posiadało odpowiednik w EER
- R1.6 jest niemożliwe, aby to, że aom:Role jest dwukierunkowa, posiadało odpowiednik w EER
- R1.7 jeżeli aom:Role ma kompozycję po stronie związku to odpowiadająca eer:Role jest cechowana przez zależność istnienia
- R1.8 jest niemożliwe, aby to, że aom:Role ma kompozycje po stronie elementu wiążanego, posiadało odpowiednik w EER
- R2 jeżeli aom:Assoc posiadająca aom:Role posiada co najwyżej 2 aom:Role to aom:Role odpowiada eer:RelationshipSet a końcówki roli odpowiadają eer:Role
- R3 jeżeli aom:Assoc posiadająca aom:Role posiada więcej niż 2 aom:Role to aom:Assoc odpowiada eer:RelationshipSet a w przeciwnym wypadku odpowiada eer:EntitySet
- R3.1 jeżeli aom:Assoc posiada aom:Role to odpowiadający eer:RelationshipSet posiada eer:Role a odpowiadający eer:EntitySet pełni eer:Role
- R4 jest konieczne, aby końcówka roli odpowiadała eer:EntitySet
- R4.1 jeżeli końcówka roli pełni aom:Role to eer:EntitySet pełni eer:Role
- R4.2 jeżeli końcówka roli jest kategoryzowana przez aom:Assoc to eer:EntitySet agreguje eer:RelationshipSet
- R5 supremum krotności po str. związku odpowiada eer:CardinalityKind
- R5.1 jeżeli krotność aom:Role posiada kres dolny równy supremum krotności po str. związku równy 1 to odpowiadająca eer:Role posiada eer:CardinalityKind równy eer:one a w przeciwnym wypadku odpowiadająca eer:Role posiada eer:CardinalityKind równy eer:many
- R6 infimum krotności po str. związku odpowiada eer:ParticipationKind
- R7 supremum krotności po str. elementu wiążanego nie posiada odpowiednika w EER
- R8 infimum krotności po str. elementu wiążanego nie posiada odpowiednika w EER

R9  $N^+$  nie posiada odpowiednika w EER



Rysunek 5.6. Diagram obrazujący odwzorowanie  $\ast_{AOM}^{[rola]} \mapsto \ast_{EER_{AOM}}^{[rola]}$

### 5.3.3.5. Wzorzec mapowania: opis asocjacji

Wzorzec mapowania ma postać odwzorowania:

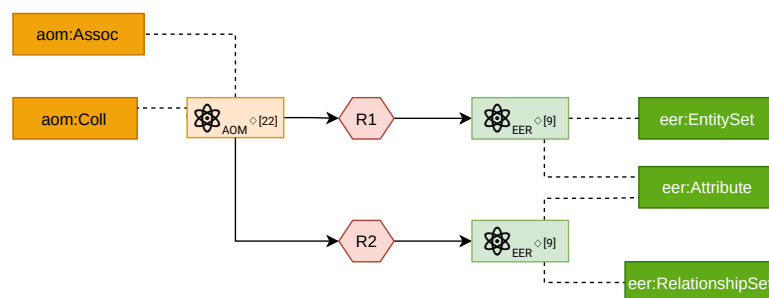
$$\ast_{AOM}^{[opis\ asocjacji]} \mapsto \ast_{EER_{AOM}}^{[opis\ asocjacji]} \quad (5.51)$$

gdzie:

$$\ast_{EER_{AOM}}^{[opis\ asocjacji]} = \left( \begin{array}{c} \emptyset, \\ \left\{ \ast_{EER}^{[atrybut\ zbioru\ zwi\u0105zku_k]} \mid \ast_{EER}^{[atrybut\ zbioru\ encji_k]} \right\}, \\ \left. \begin{array}{l} \odot \underline{[eer:EntitySet]} \\ \odot \underline{[eer:RelationshipSet]} \\ \odot \underline{[eer:Attribute_k]} \\ \odot \underline{[eer:ValueSet_k]} \end{array} \right\} \end{array} \right) \quad (5.52)$$

Reguły mapowania (rys. 5.7):

- R1 jeżeli aom:Assoc opisywany przez aom:Coll odpowiada eer:EntitySet to każdemu aom:Attr posiadany przez aom:Coll odpowiada eer:Attribute, który jest posiadany przez eer:EntitySet
- R2 jeżeli aom:Assoc opisywany przez aom:Coll odpowiada eer:RelationshipSet to każdemu aom:Attr posiadany przez aom:Coll odpowiada eer:Attribute, który jest posiadany przez eer:RelationshipSet



Rysunek 5.7. Diagram obrazujący odwzorowanie  $\ast_{AOM}^{[opis\ asocjacji]} \mapsto \ast_{EER_{AOM}}^{[opis\ asocjacji]}$

### 5.3.3.6. Wzorzec mapowania: opis roli

Opis roli w AOM z technicznego punktu widzenia pozwala określić kolekcję, z której jeden, wybrany obiekt będzie przechowywał dodatkowe dane reprezentujące konkretną rolę. Należy zauważyć, iż jest to opcjonalne przyporządkowanie o krotnościach zero do jeden. Dlatego odwzorowanie opisu roli w EER należy traktować jak

dodatkową rolę pełnioną przez zbiór encji odwzorowujący kolekcję będącą opisem roli. Z tego powodu wzorzec mapowania ma postać odwzorowania:

$$*_{\text{AOM}}^{[\text{opis roli}]} \mapsto *_{\text{EER}_{\text{AOM}}}^{[\text{opis roli}]} \quad (5.53)$$

gdzie:

$$*_{\text{EER}_{\text{AOM}}}^{[\text{opis roli}]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{\text{EER}}^{[\text{rola związku} \mid \text{zbiór związków}]}, \right. \\ \left. \begin{array}{l} \odot \underline{\text{eer:Role}} \\ \odot \underline{\text{eer:RelationshipSet}} \\ \odot \underline{\text{eer:EntitySet}} \\ \odot \underline{\text{eer:one}} \triangleleft \odot \underline{\text{eer:CardinalityKind}} \\ \odot \underline{\text{eer:partial}} \triangleleft \odot \underline{\text{eer:ParticipationKind}} \end{array} \right\} \end{array} \right) \quad (5.54)$$

Reguły mapowania są analogiczne jak dla aom:Role.

### 5.3.3.7. Wzorzec mapowania: dziedziczenie

Wzorzec mapowania ma postać odwzorowania:

$$*_{\text{AOM}}^{[\text{dziedziczenie}]} \mapsto *_{\text{EER}_{\text{AOM}}}^{[\text{dziedziczenie}]} \quad (5.55)$$

gdzie:

$$*_{\text{EER}_{\text{AOM}}}^{[\text{dziedziczenie}]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{\text{EER}}^{[\text{generalizacja-specjalizacja} \mid \text{zbiór związku}_{1,2}]}, \right. \\ \left. \begin{array}{l} \odot \underline{\text{eer:Generalization/Specialization}}, \\ \odot \underline{\text{generalizacja}} \triangleleft \left[ *_{\text{EER}}^{[19]} \right] \odot \underline{\text{eer:EntitySet}}, \\ \odot \underline{\text{specjalizacja}} \triangleleft \left[ *_{\text{EER}}^{[20]} \right] \odot \underline{\text{eer:EntitySet}}, \\ \odot \underline{\text{eer:DisjointnessKind}}, \\ \odot \underline{\text{eer:ParticipationKind}} \odot \underline{\text{eer:RelationshipSet}_1} \\ \odot \underline{\text{eer:RelationshipSet}_2} \end{array} \right\} \end{array} \right) \quad (5.56)$$

Reguły mapowania

R1 jeżeli aom:BaseNode<sub>1</sub> i aom:BaseNode<sub>2</sub> są specjalizowane jako aom:Coll to aom:Inheritance odpowiada eer:Generalization/Specialization

- R1.1 jeżeli aom:BaseNode<sub>1</sub> dziedziczy po aom:BaseNode<sub>2</sub> to aom:BaseNode<sub>1</sub> odpowiada eer:EntitySet pełniącemu rolę specjalizacji w eer:Generalization/Specialization oraz aom:BaseNode<sub>2</sub> odpowiada eer:EntitySet pełniącemu rolę generalizacji w eer:Generalization/Specialization
- R1.2 jest niemożliwe, aby to, że aom:Inheritance jest trybu aom:Inheritability w aspekcie dziedziczenia składowych, posiadało odpowiednik w EER
- R1.3 jest niemożliwe, aby to, że aom:Inheritance jest trybu aom:Inheritability w aspekcie dziedziczenia praw do pełnienia ról, posiadało odpowiednik w EER
- R1.4 jest niemożliwe, aby to, że aom:Inheritance jest trybu aom:Virtuality w zakresie wirtualności, posiadało odpowiednik w EER
- R1.5 jest obowiązkowe, aby eer:Generalization/Specialization miał określony eer:DisjointnessKind jako eer:disjoint
- R1.6 jest obowiązkowe, aby eer:Generalization/Specialization miał określony eer:ParticipationKind jako eer:total
- R2 jeżeli aom:BaseNode<sub>1</sub> i aom:BaseNode<sub>2</sub> są specjalizowane jako aom:Assoc to nie posiada to odpowiednika w EER
- R2.1 aom:Assoc specjalizującej aom:BaseNode<sub>1</sub> odpowiada eer:RelationshipSet<sub>1</sub>
- R2.2 aom:Assoc specjalizującej aom:BaseNode<sub>2</sub> odpowiada eer:RelationshipSet<sub>2</sub>
- R2.3 jeżeli aom:Inheritance nie jest trybu aom:none w aspekcie dziedziczenia składowych to każdej dziedziczalnej aom:Role posiadanej przez aom:Assoc specjalizującej aom:BaseNode<sub>2</sub> odpowiada eer:Role, która jest posiadana przez eer:RelationshipSet<sub>1</sub>

### 5.3.3.8. Wzorzec mapowania: BACT

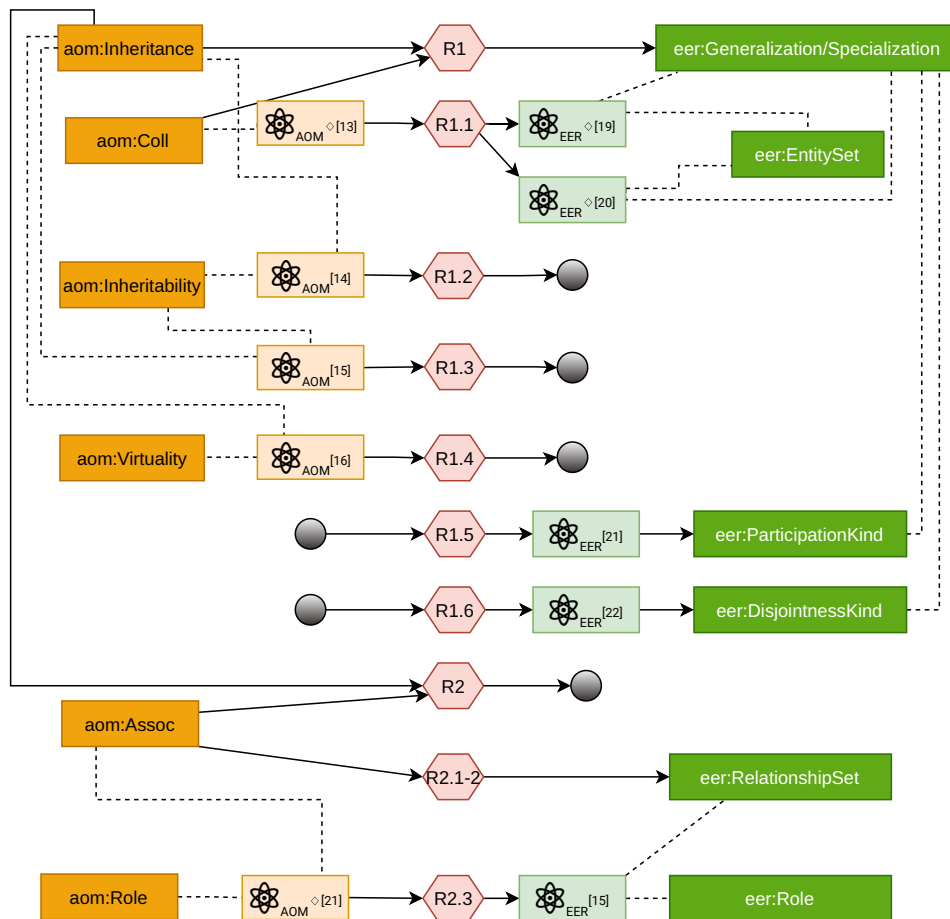
Wzorzec mapowania ma postać odwzorowania:

$$\ast_{\text{AOM}}^{[\text{BACT}]} \mapsto \ast_{\text{EER}_{\text{AOM}}}^{[\text{BACT}]} \quad (5.57)$$

gdzie:

$$\ast_{\text{EER}_{\text{AOM}}}^{[\text{BACT}]} = \left( \begin{array}{c} \emptyset, \\ \left\{ \ast_{\text{EER}}^{[\text{zbiór encji}]} \left[ \ast_{\text{EER}}^{\diamond 1} \right] \right\}, \\ \left\{ \odot_{\text{eer:EntitySet}} \right\} \end{array} \right) \quad (5.58)$$

Reguły wzorca mapowania:



Rysunek 5.8. Diagram obrazujący odwzorowanie  $\ast_{AOM}^{[dziedziczenie]} \mapsto \ast_{EER_{AOM}}^{[dziedziczenie]}$

- R1 jest konieczne aby aspekt danych oraz aspekt związku oraz 1 oraz nierozzerwale połączenie odpowiadały eer:EntitySet
- R1.1 jest obowiązkowe, aby eer:EntitySet, który odpowiada aspektowi danych nie był słaby
- R1.2 jeżeli aspekt związku posiada aom:Role, które nie są nierozzerwalnym połączeniem to każdej takiej aom:Role odpowiada eer:RelationshipSet
- R1.3 to, że nierozzerwale połączenie ma kompozycje po stronie związku odpowiada eer:EntitySet
- R1.4 to, że nierozzerwale połączenie ma kompozycje po stronie elementu związanego odpowiada eer:EntitySet
- R1.5 to, że nierozzerwale połączenie ma kres górny równy 1 po stronie elementu związanego odpowiada eer:EntitySet
- R1.6 to, że nierozzerwale połączenie ma kres dolny równy 1 po stronie elementu związanego odpowiada eer:EntitySet



R1.7 to, że nierozzerwalne połączenie ma kres górny równy 1 po stronie związku odpowiada eer:EntitySet

R1.8 to, że nierozzerwalne połączenie ma kres dolny równy 1 po stronie związku odpowiada eer:EntitySet

## 5.4. Translacja semantyki AOM na ORM

Niniejszy podrozdział zawiera opis translacji semantyki AOM na ORM. Ekstrakcja semantyki modelowania obiekt-rola została przeprowadzona według poniższych opisów specyfikacji tego podejścia: [25, 58, 60, 118].

### 5.4.1. Specyfikacja fragmentu semantyki metamodelu modelowania obiekt-rola

#### 5.4.1.1. Główne pojęcia przyjętego metamodelu modelowania obiekt-rola

##### orm:Object Type

Pojęcie ogólne: typ bytu

##### orm:EntityType

Pojęcie ogólne: orm:Object Type

##### orm:ValueType

Pojęcie ogólne: orm:Object Type

Pojęcie ogólne: typ atomowy

##### orm:Role

Pojęcie ogólne: rola

Pojęcie ogólne: atrybut

##### orm:Fact Type

Pojęcie ogólne: asocjacja

##### orm:Subtyping

Pojęcie ogólne: kategoryzacja

#### 5.4.1.2. Pojęcia pomocnicze

##### orm:Value

Pojęcie ogólne: byt atomowy

##### orm:ValueRange

Definicja: zbiór bytów atomowych nie większych niż koniec zakresu oraz nie mniejszych niż początek zakresu

## orm:ValueList

Pojęcie ogólne: domena

Definicja: zbiór orm:Value lub orm:ValueRange

## orm:RingKind

Typ pojęcia: typ atomowy

Definicja: orm:irreflexive lub orm:asymmetric lub orm:intransitive lub orm:antisymmetric lub orm:acyclic lub orm:asymmetric+intransitive lub orm:acyclic+intransitive lub orm:symmetric lub orm:symmetric+irreflexive lub orm:symmetric+intransitive lub orm:purly reflexive

### 5.4.1.3. Atomy semantyczne przyjętego metamodelu modelowania obiekt-rola

#### 5.4.1.3.1. Monadyczne atomy semantyczne

##### orm:FactType może być pochodny

✿◇[1]  
ORM

Definicja: jeżeli orm:FactType jest pochodny to orm:FactType jest cechowany przez pochodność

##### orm:ObjectType może być niezależny

✿◇[2]  
ORM

Definicja: jeżeli orm:ObjectType jest niezależny to nie musi pełnić żadnej roli orm:Role

Konieczność: każdy orm:ObjectType, który jest niezależny musi być pierwotny

##### orm:ObjectType może być pierwotny

✿◇[3]  
ORM

Definicja: jeżeli orm:ObjectType jest pierwotny, to nie może być podtypem w żadnym orm:Subtyping

##### zbiór orm:Role może być ograniczony przez obowiązkowość

✿◇[4]  
ORM

Definicja: jeżeli zbiór orm:Role jest ograniczony przez obowiązkowość to każda instancja klasyfikowana przez orm:ObjectType pełniący rolę każdej orm:Role będącej elementem zbioru musi brać udział w instancji roli tej orm:Role

Konieczność: liczność zbioru orm:Role musi wynosić co najmniej 1

Uwaga: Pojęciem atomu semantycznego może być również orm:Role. W przypadku takiej werbalizacji, należy rozumieć, iż chodzi o jednoelementowy zbiór zawierający konkretną rolę.

##### zbiór orm:Role może być ograniczony przez unikalność

✿◇[5]  
ORM

Definicja: jeżeli zbiór orm:Role jest ograniczony przez unikalność to zbiór orm:Role traktuje się jak cechę, która jest cechowana przez unikalność

Konieczność: liczność zbioru orm:Role musi wynosić co najmniej 1

Konieczność: jeżeli pewna orm:Role<sub>a</sub> należy do jednoelementowego zbioru ograniczonego przez unikalność oraz jest w orm:FactType, który posiada 2 orm:Role to cel orm:Role<sub>a</sub> jest cechowany przez krotność posiadający kres górny równy 1

Uwaga: Pojęciem atomu semantycznego może być również orm:Role. W przypadku takiej werbalizacji, należy rozumieć, iż chodzi o jednoelementowy zbiór zawierający konkretną rolę.

##### orm:Subtyping może być ograniczony przez całkowitość

✿◇[6]  
ORM

Definicja: jeżeli orm:Subtyping jest ograniczony przez całkowitość to orm:Subtyping jest cechowany przez pokrycie

### orm:Subtyping może być ograniczony przez rozłączność

◇[7]  
ORM

Definicja: jeżeli orm:Subtyping jest ograniczony przez całkowitość to orm:Subtyping jest cechowany przez rozłączność

#### 5.4.1.3.2. Poliadyczne atomy semantyczne

### orm:ObjectType może pełnić rolę orm:Role

◇[8]  
ORM

Definicja: jeżeli orm:ObjectType pełni rolę orm:Role to orm:ObjectType pełni orm:Role w orm:FactType, który posiada daną orm:Role

### orm:Role jest w orm:FactType

[9]  
ORM

Forma synonimiczna: orm:FactType posiada orm:Role

### orm:EntityType może reifikować orm:FactType

◇[10]  
ORM

Definicja: jeżeli orm:EntityType reifikuje orm:FactType to orm:EntityType posiada referencję do orm:FactType umożliwiającą pośrednie uczestnictwo w związku asocjacji

### orm:Subtyping posiada nadtyp orm:ObjectType

[11]  
ORM

Definicja: orm:Subtyping posiada generalizację, którą jest orm:ObjectType

Konieczność: orm:Subtyping posiada co najmniej jeden nadtyp orm:ObjectType

Forma synonimiczna: orm:ObjectType jest nadtypem w orm:Subtyping

### orm:Subtyping posiada podtyp orm:ObjectType

[12]  
ORM

Definicja: orm:Subtyping posiada specjalizację, którą jest orm:ObjectType

Konieczność: orm:Subtyping posiada co najmniej jeden podtyp orm:ObjectType

Forma synonimiczna: orm:ObjectType jest podtypem w orm:Subtyping

### orm:ValueType może być ograniczony przez orm:ValueList

◇[13]  
ORM

Definicja: jeżeli orm:ValueType jest ograniczony przez orm:ValueList to orm:ValueType jest cechowany przez domenę, którą jest orm:ValueList

### orm:Role może być ograniczona przez orm:ValueList

◇[14]  
ORM

Definicja: jeżeli orm:RoleType jest ograniczony przez orm:ValueList to instancje orm:ObjectType pełniące rolę orm:Role muszą przechowywać wartości z orm:ValueList realizując orm:Role

### orm:Role<sub>1</sub> może być podzbiorem orm:Role<sub>2</sub>

◇[15]  
ORM

Definicja: jeżeli orm:Role<sub>1</sub> jest podzbiorem orm:Role<sub>2</sub> to każda instancja klasyfikowana przez orm:ObjectType pełniący rolę orm:Role<sub>1</sub> bierze udział w instancji roli klasyfikowanej przez orm:Role<sub>2</sub>

### orm:Role<sub>1</sub> może być równa orm:Role<sub>2</sub>

◇[16]  
ORM

Definicja: jeżeli orm:Role<sub>1</sub> jest równa orm:Role<sub>2</sub> to każda instancja klasyfikowana przez orm:ObjectType pełniący rolę orm:Role<sub>1</sub> bierze udział w instancji roli klasyfikowanej przez orm:Role<sub>2</sub> oraz każda instancja klasyfikowana przez orm:ObjectType pełniący rolę orm:Role<sub>2</sub> bierze udział w instancji roli klasyfikowanej przez orm:Role<sub>1</sub>

orm:Role<sub>1</sub> oraz orm:Role<sub>2</sub> mogą wzajemnie się wykluczać

◇[17]  
ORM

Definicja: jeżeli orm:Role<sub>1</sub> oraz orm:Role<sub>2</sub> wzajemnie się wykluczają to żadna instancja klasyfikowana przez orm:ObjectType pełniący rolę orm:Role<sub>1</sub> nie bierze udziału w instancji roli klasyfikowanej przez orm:Role<sub>2</sub> oraz żadna instancja klasyfikowana przez orm:ObjectType pełniący rolę orm:Role<sub>2</sub> nie bierze udziału w instancji roli klasyfikowanej przez orm:Role<sub>1</sub>

orm:Role może być ograniczona przez częstotliwość równą liczbie całkowitej

◇[18]  
ORM

Definicja: jeżeli orm:Role jest ograniczona przez częstotliwość równą liczbie całkowitej to źródło orm:Role jest cechowane przez krotność, która posiada kres dolny oraz kres górny równy liczbie całkowitej

orm:Role może być ograniczona przez częstotliwość wynoszącą co najmniej liczbę całkowitą

◇[19]  
ORM

Definicja: jeżeli orm:Role jest ograniczona przez częstotliwość wynoszącą co najmniej liczbę całkowitą to źródło orm:Role jest cechowane przez krotność, która posiada kres dolny równy liczbie całkowitej

orm:Role może być ograniczona przez częstotliwość wynoszącą co najwyżej liczbę całkowitej

◇[20]  
ORM

Definicja: jeżeli orm:Role jest ograniczona przez częstotliwość wynoszącą co najwyżej liczbę całkowitą to źródło orm:Role jest cechowane przez krotność, która posiada kres górny równy liczbie całkowitej

orm:Role może być ograniczona przez częstotliwość z przedziału od liczby całkowitej<sub>1</sub> do liczby całkowitej<sub>2</sub>

◇[21]  
ORM

Definicja: jeżeli jeżeli orm:Role jest ograniczona przez częstotliwość z przedziału od liczby całkowitej<sub>1</sub> do liczby całkowitej<sub>2</sub> to źródło orm:Role jest cechowane przez krotność, która posiada kres dolny równy liczbie całkowitej<sub>1</sub> oraz kres górny równy liczbie całkowitej<sub>2</sub>

orm:Role<sub>1</sub> oraz orm:Role<sub>2</sub> mogą być w pierścieniowym ograniczeniu orm:RingKind

◇[22]  
ORM

Definicja: jeżeli orm:Role<sub>1</sub> oraz orm:Role<sub>2</sub> są w pierścieniowym ograniczeniu orm:RingKind to orm:Role<sub>1</sub> oraz orm:Role<sub>2</sub> są cechowane przez dwuargumentowe ograniczenie roli określone przez orm:RingKind

#### 5.4.2. Części semantyczne przyjętego metamodelu modelowania obiekt-rola

$$*_{ORM}^{[obiekt-byt]} = \left( \begin{array}{c} *_{ORM}^{\{\diamond 2, \diamond 3\}}, \\ \emptyset, \\ \left\{ \odot_{[orm:EntityType]} < \odot_{[orm:ObjectType]} \right\} \end{array} \right) \quad (5.59)$$

$$*_{ORM}^{[obiekt-wartość]} = \left( \begin{array}{c} *_{ORM}^{\{\diamond 2, \diamond 13\}}, \\ \emptyset, \\ \left\{ \odot_{[orm:ValueType]} < \odot_{[orm:ObjectType]} \right\} \end{array} \right) \quad (5.60)$$

$$*_{ORM}^{[fakt]} = \left( \begin{array}{c} *_{ORM}^{\{\diamond 1\}}, \\ \emptyset, \\ \left\{ \odot_{[orm:FactType]} \right\} \end{array} \right) \quad (5.61)$$

$$*_{ORM}^{[rola]} = \left( \begin{array}{c} *_{ORM}^{\{\diamond 4, \diamond 5, \diamond 8, 9, \diamond 14, \diamond 15, \diamond 16, \diamond 17, \diamond 18, \diamond 19, \diamond 20, \diamond 21, \diamond 22\}}, \\ \emptyset, \\ \left\{ \begin{array}{c} \odot_{[orm:Role]} \\ \odot_{[orm:FactType]} \\ \odot_{[orm:ObjectType]} \\ \diamond \odot_{[orm:RingKind]} \end{array} \right\} \end{array} \right) \quad (5.62)$$

$$*_{ORM}^{[podtyp]} = \left( \begin{array}{c} *_{ORM}^{\{\diamond 11, \diamond 12\}}, \\ \emptyset, \\ \left\{ \begin{array}{c} \odot_{[orm:Subtyping]} \\ \odot_{[nadtyp]} \triangleleft [ *_{ORM}^{[11]} ] \odot_{[orm:ObjectType]} \\ \odot_{[podtyp]} \triangleleft [ *_{ORM}^{[12]} ] \odot_{[orm:ObjectType]} \end{array} \right\} \end{array} \right) \quad (5.63)$$

$$*_{ORM}^{[zbiór podtypów]} = \left( \begin{array}{c} *_{ORM}^{\{\diamond 6, \diamond 7\}}, \\ \emptyset, \\ \left\{ \odot_{[orm:Subtyping_n]} \right\} \end{array} \right) \quad (5.64)$$

$$*_{ORM}^{[reifikacja]} = \left( \begin{array}{c} *_{ORM}^{\{\diamond 10\}} \\ \emptyset, \\ \left\{ \begin{array}{c} \odot \underline{orm:EntityType} \\ \odot \underline{orm:FactType} \end{array} \right\} \end{array} \right) \quad (5.65)$$

### 5.4.3. Wzorce mapowania

#### 5.4.3.1. Wzorzec mapowania: kolekcja

Wzorzec mapowania ma postać odwzorowania:

$$*_{AOM}^{[kolekcja]} \mapsto *_{ORM_{AOM}}^{[kolekcja]} \quad (5.66)$$

gdzie:

$$*_{ORM_{AOM}}^{[kolekcja]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{ORM}^{[obiekt-byt]} \left[ *_{ORM}^{\{\diamond 2\}} \right] \right\}, \\ \left\{ \odot \underline{orm:EntityType} \right\} \end{array} \right) \quad (5.67)$$

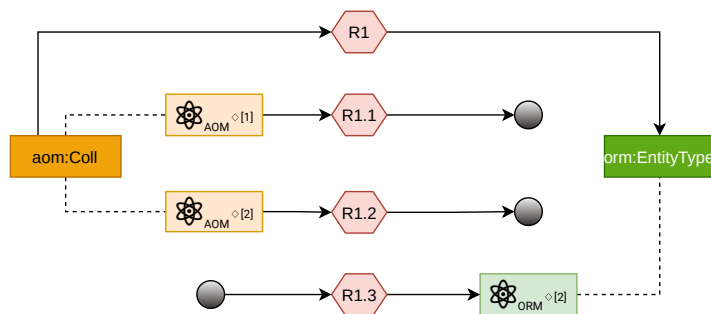
Reguły wzorca mapowania (rys. 5.9):

R1 **jest konieczne aby** aom:Coll **odpowiadało** orm:EntityType

R1.1 **jest niemożliwe, aby to, że** aom:Coll **jest abstrakcyjna, posiadało odpowiednik w** ORM

R1.2 **jest niemożliwe, aby to, że** aom:Coll **jest uninawigowalna, posiadało odpowiednik w** ORM

R1.3 **jest obowiązkowe, aby** orm:EntityType, **który odpowiada** aom:Coll **był niezależny**



Rysunek 5.9. Diagram obrazujący odwzorowanie  $*_{AOM}^{[kolekcja]} \mapsto *_{ORM_{AOM}}^{[kolekcja]}$

### 5.4.3.2. Wzorzec mapowania: atrybut

Wzorzec mapowania ma postać odwzorowania:

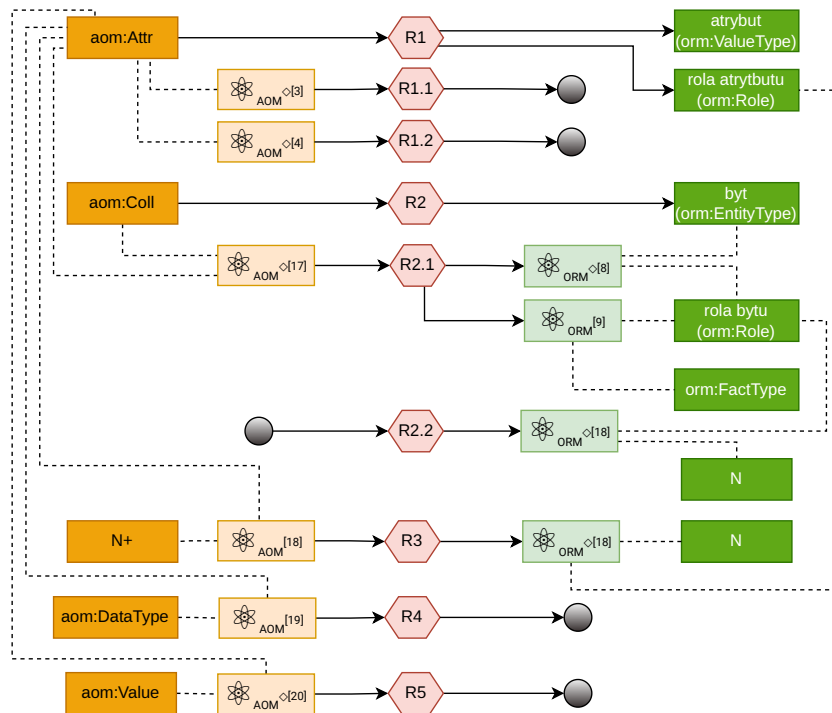
$$*_{AOM}^{[atrybut]} \mapsto *_{ORM_{AOM}}^{[atrybut]} \quad (5.68)$$

gdzie:

$$*_{ORM_{AOM}}^{[atrybut]} = \left( \begin{array}{c} \emptyset, \\ \left\{ \begin{array}{l} *_{ORM}^{[rola\ bytu]} \triangleleft *_{ORM}^{[rola]}, \\ *_{ORM}^{[rola\ atrybutu]} \triangleleft *_{ORM}^{[rola]}, \\ *_{ORM}^{[obiekt-byt]}, \\ *_{ORM}^{[obiekt-wartosc]}, \\ *_{ORM}^{[typ\ faktu]} \end{array} \right\}, \\ \left\{ \begin{array}{l} \odot_{[byt]} \triangleleft [ *_{ORM}^{[rola\ bytu]}, *_{ORM}^{[obiekt-byt]} ] \odot_{[orm:EntityType]}, \\ \odot_{[atrybut]} \triangleleft [ *_{ORM}^{[rola\ atrybutu]}, *_{ORM}^{[obiekt-wartosc]} ] \odot_{[orm:ValueType]}, \\ \odot_{[rola\ bytu]} \triangleleft [ *_{ORM}^{[rola\ bytu]} ] \odot_{[orm:Role]}, \\ \odot_{[rola\ atrybutu]} \triangleleft [ *_{ORM}^{[rola\ atrybutu]} ] \odot_{[orm:Role]}, \\ \odot_{[orm:FactType]}, \end{array} \right\} \end{array} \right) \quad (5.69)$$

Reguły wzorca mapowania (rys. 5.10):

- R1 jest konieczne aby aom:Attr odpowiadało orm:Role oraz orm:ValueType
- R1.1 jest niemożliwe, aby to, że aom:Attr jest dziedziczny, posiadało odpowiednik w ORM
- R1.2 jest niemożliwe, aby to, że aom:Attr jest wirtualny, posiadało odpowiednik w ORM
- R2 jest konieczne aom:Coll odpowiadało orm:EntityType
- R2.1 jeżeli aom:Coll posiada aom:Attr to odpowiadający orm:EntityType pełni rolę rola bytu, która jest w orm:FactType posiadającym rolę atrybutu pełnioną przez orm:ValueType, który odpowiada aom:Attr
- R2.2 jest konieczne aby rola bytu była ograniczona przez częstotliwość równą 1
- R3 jest konieczne aby  dodatnia liczba naturalna będąca licznikiem aom:Attr odpowiadała liczbie całkowitej będącej częstotliwością ograniczającą rolę atrybutu
- R4 jest niemożliwe, aby to, że aom:Attr jest typu aom:DataType, posiadało odpowiednik w ORM
- R5 jest niemożliwe, aby to, że aom:Attr ma wartość domyślną aom:Value, posiadało odpowiednik w ORM



Rysunek 5.10. Diagram obrazujący odwzorowanie  $\ast_{AOM}^{[atrybut]} \mapsto \ast_{ORM_{AOM}}^{[atrybut]}$

### 5.4.3.3. Wzorzec mapowania: asocjacja

Wzorzec mapowania ma postać odwzorowania:

$$\ast_{AOM}^{[asocjacja]} \mapsto \ast_{ORM_{AOM}}^{[asocjacja]} \quad (5.70)$$

gdzie:

$$\ast_{ORM_{AOM}}^{[asocjacja]} = \left( \begin{array}{c} \emptyset, \\ \left\{ \ast_{ORM}^{[fakt]}, \right. \\ \left. \diamond \ast_{ORM}^{[reifikacja]} \right\}, \\ \left\{ \odot [orm:FactType] \right. \\ \left. \diamond \odot [orm:EntityType] \right\} \end{array} \right) \quad (5.71)$$

Reguły wzorca mapowania (rys. 5.11):

R1 jest konieczne aby aom:Assoc odpowiadało orm:FactType

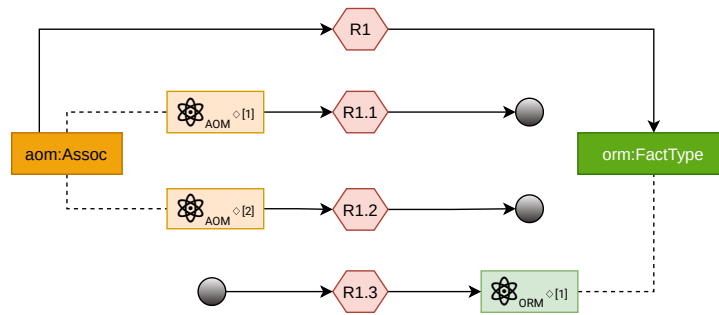
R1.1 jest niemożliwe, aby to, że aom:Assoc jest abstrakcyjna, posiadało odpowiednik w ORM

R1.2 jest niemożliwe, aby to, że aom:Assoc jest uninawigowalna, posiadało odpowiednik w ORM

R1.3 jest zabronione, aby orm:FactType, który odpowiada aom:Assoc był pochodny



R2 jeżeli aom:Assoc uczestniczy w aom:Role, to aom:Assoc odpowiada orm:EntityType, który reifikuje orm:FactType



Rysunek 5.11. Diagram obrazujący odwzorowanie  $*_{AOM}^{[asocjacja]} \mapsto *_{ORM_{AOM}}^{[asocjacja]}$

#### 5.4.3.4. Wzorzec mapowania: rola

Wzorzec mapowania ma postać odwzorowania:

$$*_{AOM}^{[rola]} \mapsto *_{ORM_{AOM}}^{[rola]} \quad (5.72)$$

gdzie:

$$*_{ORM_{AOM}}^{[rola]} = \left( \begin{array}{c} \emptyset, \\ \{ *_{ORM}^{[rola]} \}, \\ \left\{ \begin{array}{l} \odot_{[orm:Role]}, \\ \odot_{[orm:ObjectType]}, \\ \odot_{[związek]} \triangleleft \odot_{[orm:FactType]}, \end{array} \right\} \end{array} \right) \quad (5.73)$$

Reguły wzorca mapowania (rys. 5.12):

R1 jest konieczne aby aom:Role odpowiadało orm:Role

R1.1 jest niemożliwe, aby to, że aom:Role jest wirtualna, posiadało odpowiednik w ORM

R1.2 jest niemożliwe, aby to, że aom:Role jest dziedziczna, posiadało odpowiednik w ORM

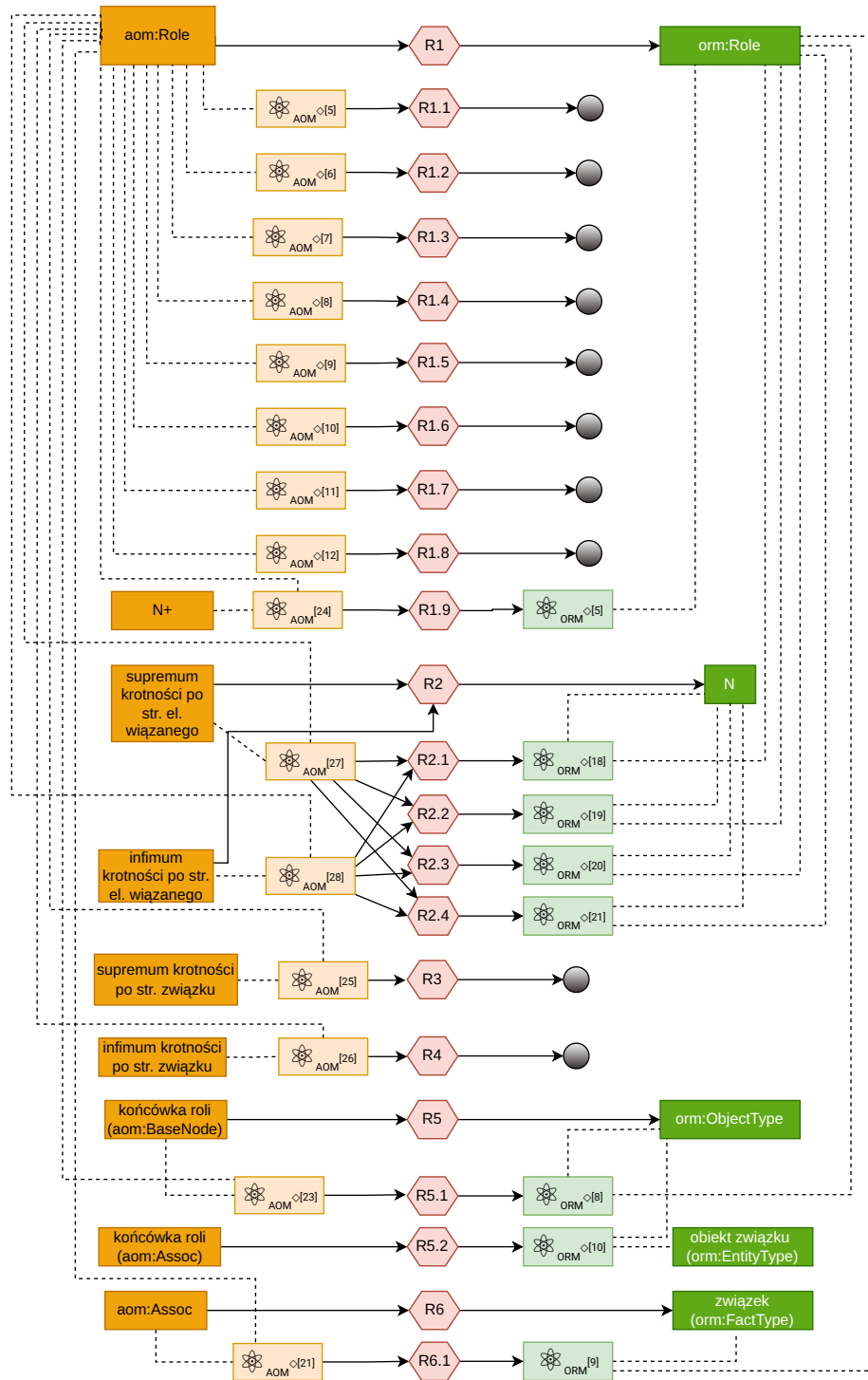
R1.3 jest niemożliwe, aby to, że aom:Role jest uninawigowalna, posiadało odpowiednik w ORM

R1.4 jest niemożliwe, aby to, że aom:Role jest skierowana do właściciela, posiadało odpowiednik w ORM

R1.5 jest niemożliwe, aby to, że aom:Role jest skierowana do celu, posiadało odpowiednik w ORM

- R1.6 jest niemożliwe, aby to, że aom:Role jest dwukierunkowa, posiadało odpowiednik w ORM
- R1.7 jest niemożliwe, aby to, że aom:Role ma kompozycję po stronie związku posiadało odpowiednik w ORM
- R1.8 jest niemożliwe, aby to, że aom:Role ma kompozycje po stronie elementu związanego, posiadało odpowiednik w ORM
- R1.9 jeżeli aom:Role posiada unikalność równą 1 to odpowiadająca orm:Role jest ograniczona przez unikalność
- R2 jest konieczne aby supremum krotności po str. związku odpowiadało liczbie całkowitej oraz infimum krotności po str. związku odpowiadało liczbie całkowitej
- R2.1 jeżeli supremum krotności po str. związku jest równe infimum krotności po str. związku to orm:Role jest ograniczona przez częstotliwość równą odpowiadającej liczbie całkowita
- R2.2 jeżeli infimum krotności po str. związku jest liczbą naturalną oraz supremum krotności po str. związku nie jest liczbą naturalną to orm:Role jest ograniczona przez częstotliwość wynoszącą co najmniej odpowiadającej liczbie całkowitej równej infimum krotności po str. związku
- R2.3 jeżeli infimum krotności po str. związku nie jest liczbą naturalną oraz supremum krotności po str. związku jest liczbą naturalną to orm:Role jest ograniczona przez częstotliwość wynoszącą co najwyżej odpowiadającej liczbie całkowitej równej supremum krotności po str. związku
- R2.4 jeżeli infimum krotności po str. elementu związanego jest liczbą naturalną oraz supremum krotności po str. związku jest liczbą naturalną oraz infimum krotności po str. związku nie jest równa supremum krotności po str. związku to orm:Role jest ograniczona przez częstotliwość z przedziału od liczby całkowitej równej infimum krotności po str. związku do liczby całkowitej równej supremum krotności po str. związku
- R3 infimum krotności po str. el. związanego nie posiada odpowiednika w ORM
- R3.1 jeżeli infimum krotności po str. związku jest większy od 1 to orm:Role jest ograniczona przez obowiązkowość
- R4 supremum krotności po str. el. związanego nie posiada odpowiednika w ORM
- R4.1 jeżeli supremum krotności po str. związku jest równy 1 to orm:Role jest ograniczona przez unikalność
- R5 jest konieczne aby końcówka roli odpowiadała orm:ObjectType

- R5.1 jeżeli końcówka roli uczestniczy w aom:Role to odpowiadająca orm:ObjectType pełni rolę orm:Role
- R6 jest konieczne aby aom:Assoc odpowiadało związkowi
- R6.1 jeżeli aom:Assoc posiada aom:Role to odpowiadający związek posiada orm:Role



Rysunek 5.12. Diagram obrazujący odwzorowanie  $\ast_{AOM}^{[rola]} \mapsto \ast_{ORM_{AOM}}^{[rola]}$

### 5.4.3.5. Wzorzec mapowania: opis asocjacji

Wzorzec mapowania ma postać odwzorowania:

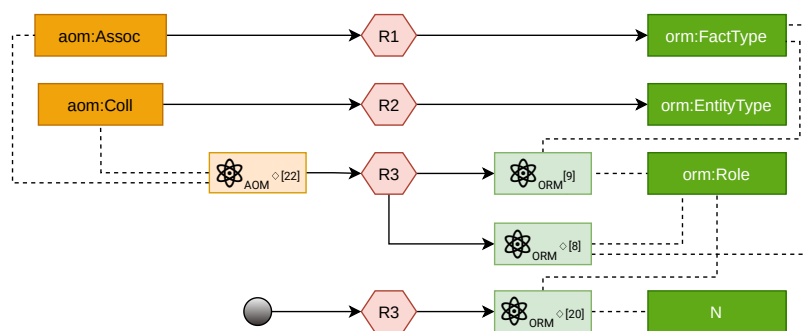
$$\ast_{\text{AOM}}^{[\text{opis asocjacji}]} \mapsto \ast_{\text{ORM}_{\text{AOM}}}^{[\text{opis asocjacji}]} \quad (5.74)$$

gdzie:

$$\ast_{\text{ORM}_{\text{AOM}}}^{[\text{opis asocjacji}]} = \left( \begin{array}{c} \emptyset, \\ \left\{ \ast_{\text{ORM}}^{[\text{rola}]} \left[ \ast_{\text{ORM}}^{\diamond 5} \right] \right\}, \\ \left\{ \begin{array}{l} \odot_{\text{opis}} \triangleleft \odot_{\text{orm:Role}}, \\ \odot_{\text{orm:EntityType}} \triangleleft \odot_{\text{orm:ObjectType}}, \\ \odot_{\text{orm:FactType}}, \\ \odot_{\underline{1}} \triangleleft \left[ \ast_{\text{ORM}}^{\diamond 20} \right] \odot_{\text{liczba całkowita}} \end{array} \right\} \end{array} \right) \quad (5.75)$$

Reguły wzorca mapowania (rys. 5.13):

- R1 jest konieczne, aby aom:Assoc odpowiadało orm:FactType
- R2 jest konieczne, aby aom:Coll odpowiadało orm:EntityType
- R3 jeżeli aom:Assoc jest opisywana przez aom:Coll to odpowiadający aom:Assoc orm:FactType posiada opis, która jest pełniona przez orm:EntityType odpowiadający aom:Coll
- R3.1 jest obowiązkowe, aby opis był ograniczony przez częstotliwość wynoszącą co najwyżej 1



Rysunek 5.13. Diagram obrazujący odwzorowanie  $\ast_{\text{AOM}}^{[\text{opis asocjacji}]} \mapsto \ast_{\text{ORM}_{\text{AOM}}}^{[\text{opis asocjacji}]}$

### 5.4.3.6. Wzorzec mapowania: opis roli

Opis roli w AOM z technicznego punktu widzenia pozwala określić kolekcję, z której jeden, wybrany obiekt będzie przechowywał dodatkowe dane reprezentujące konkretną rolę. Należy zauważyć, iż jest to opcjonalne przyporządkowanie zero do jeden.

Dlatego, analogicznie jak w przypadku EER, dla ORM odwzorowanie opisu roli należy traktować jak dodatkową rolę pełnioną przez opisujący obiekt, odwzorowujący kolekcję będącą opisem roli. Wzorzec mapowania ma postać odwzorowania:

$$*_{\text{AOM}}^{[\text{opis roli}]} \mapsto *_{\text{ORM}_{\text{AOM}}}^{[\text{opis roli}]} \quad (5.76)$$

gdzie:

$$*_{\text{ORM}_{\text{AOM}}}^{[\text{opis roli}]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{\text{UML}}^{[\text{rola}]} \right\}, \\ \left\{ \begin{array}{c} \odot \underline{\text{orm:Role}} \\ \odot \underline{\text{orm:ObjectType}} \\ \odot \underline{\text{orm:FactType}} \end{array} \right\} \end{array} \right) \quad (5.77)$$

Reguły mapowania są analogiczne jak dla roli.

#### 5.4.3.7. Wzorzec mapowania: dziedziczenie

Wzorzec mapowania ma postać odwzorowania:

$$*_{\text{AOM}}^{[\text{dziedziczenie}]} \mapsto *_{\text{ORM}_{\text{AOM}}}^{[\text{dziedziczenie}]} \quad (5.78)$$

gdzie:

$$*_{\text{ORM}_{\text{AOM}}}^{[\text{dziedziczenie}]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{\text{ORM}}^{[\text{podtyp}]} \right. \\ \left. \diamond *_{\text{ORM}}^{[\text{reifikacja}_{1,2}]} \right\}, \\ \left\{ \begin{array}{c} \odot \underline{\text{orm:Subtyping}} \\ \odot \underline{\text{nadtyp}} \\ \odot \underline{\text{podtyp}} \\ \diamond \odot \underline{\text{orm:FactType}_{1,2}} \end{array} \right\} \end{array} \right) \quad (5.79)$$

Reguły wzorca mapowania (rys. 5.14):

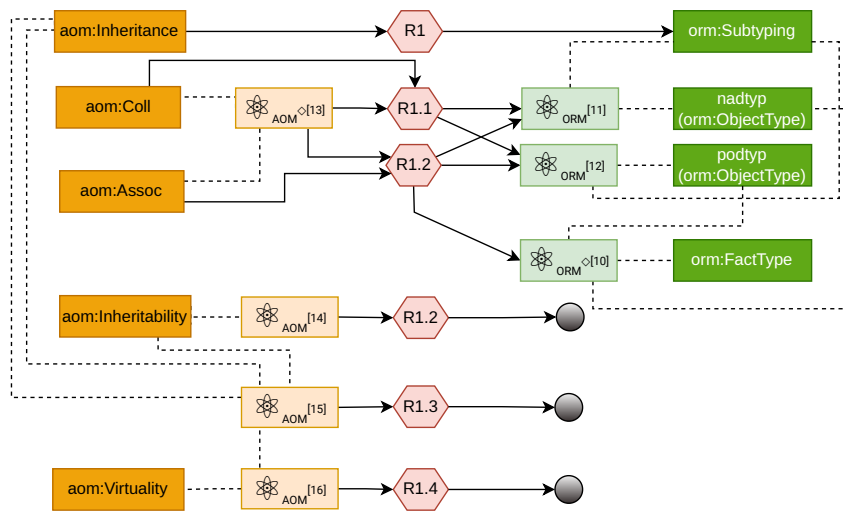
R1 jest konieczne, aby aom:Inheritance odpowiadało orm:Subtyping

R1.1 jeżeli aom:BaseNode<sub>1</sub> dziedziczy po aom:BaseNode<sub>2</sub> oraz aom:BaseNode<sub>1</sub> i aom:BaseNode<sub>2</sub> są specjalizowane jako aom:Coll to aom:BaseNode<sub>1</sub> odpowiada podtyp oraz aom:BaseNode<sub>2</sub> odpowiada nadtyp

R1.2 jeżeli aom:BaseNode<sub>1</sub> dziedziczy po aom:BaseNode<sub>2</sub> oraz aom:BaseNode<sub>1</sub> i aom:BaseNode<sub>2</sub> są specjalizowane jako aom:Assoc to aom:BaseNode<sub>1</sub> odpowiada orm:FactType<sub>1</sub> reifikowanemu przez podtyp oraz aom:BaseNode<sub>2</sub>

odpowiada orm:FactType<sub>2</sub> reifikowanemu przez nadtyp oraz orm:Subtyping posiada nadtyp nadtyp i posiada podtyp podtyp

- R2 jest niemożliwe, aby to, że aom:Inheritance jest trybu aom:Inheritability w aspekcie dziedziczenia składowych, posiadało odpowiednik w ORM
- R3 jest niemożliwe, aby to, że aom:Inheritance jest trybu aom:Inheritability w aspekcie dziedziczenia praw do pełnienia ról, posiadało odpowiednik w ORM
- R4 jest niemożliwe, aby to, że aom:Inheritance jest trybu aom:Virtuality w zakresie wirtualności, posiadało odpowiednik w ORM



Rysunek 5.14. Diagram obrazujący odwzorowanie  $\ast_{AOM}^{[dziedziczenie]} \mapsto \ast_{ORM_{AOM}}^{[dziedziczenie]}$

#### 5.4.3.8. Wzorzec mapowania: BACT

Wzorzec mapowania ma postać odwzorowania:

$$\ast_{AOM}^{[BACT]} \mapsto \ast_{ORM_{AOM}}^{[BACT]} \quad (5.80)$$

gdzie:

$$\ast_{ORM_{AOM}}^{[BACT]} = \left( \begin{array}{c} \emptyset, \\ \left\{ \begin{array}{l} \ast_{ORM}^{[obiekt-byt]} \\ \ast_{ORM}^{[rola]} \left[ \ast_{ORM}^{\{4,5,18\}} \right] \\ \ast_{ORM}^{[reifikacja]} \end{array} \right\}, \\ \left\{ \begin{array}{l} \odot \underline{orm:EntityType} \\ \odot \underline{orm:Role} \\ \odot \underline{orm:FactType} \end{array} \right\} \end{array} \right) \quad (5.81)$$

Reguły wzorca mapowania:

- R1 jest konieczne aby aspekt danych odpowiadał orm:EntityType
- R2 jest konieczne aby aspekt związku odpowiadał orm:FactType
- R3 jest konieczne aby nierozzerwalne połączenie oraz 1 odpowiadały orm:Role
- R3.1 to, że nierozzerwalne połączenie ma kompozycje po stronie związku odpowiada eer:Role
- R3.2 to, że nierozzerwalne połączenie ma kompozycje po stronie elementu związanego odpowiada eer:Role
- R3.3 to, że nierozzerwalne połączenie ma kres górny równy 1 po stronie elementu związanego odpowiada orm:Role ograniczonej przez unikalność
- R3.4 to, że nierozzerwalne połączenie ma kres dolny równy 1 po stronie elementu związanego odpowiada orm:Role ograniczonej przez obowiązkowość
- R3.5 to, że nierozzerwalne połączenie ma kres górny oraz kres dolny równy 1 po stronie związku odpowiada orm:Role ograniczonej przez częstotliwość równą 1

## 5.5. Translacja semantyki AOM na UML

W niniejszym podrozdziale zawarto opis translacji semantyki AOM na UML. Ekstrakcja semantyki tego metamodelu obejmuje pojęcia oraz atomy semantyczne występujące na diagramie klas. Należy zauważyć, iż UML jest metamodelem bogatym gramatycznie i semantycznie, obejmującym wiele aspektów modelowania systemów. W ramach niniejszej pracy wzięto pod uwagę jedynie podzbiór semantyki tego standardu istotny z punktu widzenia metamodelu asocjacyjnego. Ekstrakcja semantyki została przeprowadzona przy użyciu opisów metamodelu w ramach następujących publikacji: [63, 114].

### 5.5.1. Specyfikacja fragmentu semantyki metamodelu zunifikowanego języka modelowania

#### 5.5.1.1. Główne pojęcia przyjętego metamodelu zunifikowanego języka modelowania

##### uml:Class

Pojęcie ogólne: typ bytu

Pojęcie ogólne: uml:Classifier

##### uml:Association

Pojęcie ogólne: asocjacja

Pojęcie ogólne: uml:Classifier

## uml:AssociationClass

Pojęcie ogólne: [uml:Class](#)

Pojęcie ogólne: [uml:Association](#)

## uml:Generalization

Pojęcie ogólne: [kategoryzacja](#)

## uml:Property

Pojęcie ogólne: [atrybut](#)

Pojęcie ogólne: [rola](#)

## uml:DataType

Pojęcie ogólne: [typ bytu](#)

Pojęcie ogólne: [uml:Classifier](#)

## uml:ValueSpecification

Pojęcie ogólne: [wartość](#)

### 5.5.1.2. Pojęcia pomocnicze przyjętego metamodelu zunifikowanego języka modelowania

#### uml:NamedElement

Pojęcie ogólne: [pojęcie modelowania](#)

#### uml:Type

Pojęcie ogólne: [uml:NamedElement](#)

#### uml:Classifier

Pojęcie ogólne: [klasyfikator](#)

Pojęcie ogólne: [uml:Type](#)

#### uml:Enumeration

Pojęcie ogólne: [typ atomowy](#)

#### uml:VisibilityKind

Typ pojęcia: [typ atomowy](#)

Definicja: [uml:package](#) lub [uml:private](#) lub [uml:protected](#) lub [uml:public](#)

### 5.5.1.3. Atomy semantyczne przyjętego metamodelu zunifikowanego języka modelowania

#### 5.5.1.3.1. Monadyczne atomy semantyczne

##### uml:Classifier może być abstrakcyjny



Definicja: jeżeli [uml:Classifier](#) jest abstrakcyjny to [uml:Classifier](#) jest cechowany przez nieinstancjonowalność

##### uml:Property może mieć własność współdzielonej agregacji





Definicja: jeżeli uml:Property ma własność współdzielonej agregacji to uml:Property jest cechowane przez semantyczną kierunkowość określającą związek część-całość

Konieczność: uml:Property, które ma własność współdzielonej agregacji musi być końcówką uml:asocjacji binarnej

### uml:Property może mieć własność kompozycyjnej agregacji

◇[3]  
UML

Definicja: jeżeli uml:Property ma własność kompozycyjnej agregacji to uml:Property jest cechowane przez zależność czasu życia

Konieczność: uml:Property, które ma własność kompozycyjnej agregacji musi być końcówką uml:asocjacji binarnej

Konieczność: jeżeli uml:Property ma własność współdzielonej agregacji to źródło i cel uml:Property są cechowane przez krotność mniejszą lub równą 1

### uml:Property może być identyfikatorem

◇[4]  
UML

Definicja: jeżeli uml:Property jest identyfikatorem to uml:Property jest cechowane przez identyfikację

### uml:Property może być tylko do odczytu

◇[5]  
UML

Definicja: jeżeli uml:Property jest tylko do odczytu to uml:Property nie jest cechowane przez mutowalność

### uml:Property może być uporządkowane

◇[6]  
UML

Definicja: jeżeli uml:Property jest uporządkowane to uml:Property jest cechowane przez porządek

### uml:Property może być unikalne

◇[7]  
UML

Definicja: jeżeli uml:Property jest unikalne to uml:Property jest cechowane przez unikalność

### uml:Property może być statyczne

◇[8]  
UML

Definicja: jeżeli uml:Property jest statyczne to uml:Property jest cechowane przez statyczność

### zbiór uml:Generalization może być pokrywający

◇[9]  
UML

Definicja: jeżeli zbiór uml:Generalization jest pokrywający to konkretyzuje on kategoryzację cechowaną przez pokrycie

### zbiór uml:Generalization może być rozłączny

◇[10]  
UML

Definicja: jeżeli zbiór uml:Generalization jest rozłączny to konkretyzuje on kategoryzację cechowaną przez rozłączność

### uml:Generalization może być zastępowalna

◇[11]  
UML

Definicja: jeżeli uml:Generalization jest zastępowalna to uml:Generalization jest cechowana przez polimorfizm

### 5.5.1.3.2. Poliadyczne atomy semantyczne

#### uml:Class może posiadać atrybut uml:Property

◇[12]  
UML

Definicja: jeżeli uml:Class posiada atrybut uml:Property to uml:Class posiada uml:Property

#### uml:Property może być końcówką uml:Association

◇[13]  
UML

Definicja: jeżeli uml:Property jest końcówką uml:Association to pewien uml:Classifier pełni rolę uml:Property w uml:Association

#### uml:Association może posiadać końcówkę uml:Property

◇[14]  
UML

Definicja: jeżeli uml:Association posiada końcówką uml:Property to uml:Association posiada uml:Property

#### uml:Property może być nawigowalna z uml:Association

◇[15]  
UML

Definicja: jeżeli uml:Property jest nawigowalna z uml:Association to źródło uml:Property jest nawigowalne

#### uml:Property<sub>1</sub> może być podzbiorem uml:Property<sub>2</sub>

◇[16]  
UML

Definicja: jeżeli uml:Property<sub>1</sub> jest podzbiorem uml:Property<sub>2</sub> to każda instancja klasyfikowana przez uml:Type będącego typem dla uml:Property<sub>1</sub> bierze udział w instancji roli klasyfikowanej przez uml:Property<sub>2</sub>

#### uml:Property może mieć zdefiniowany kres dolny jako liczbę całkowitą

◇[17]  
UML

Definicja: jeżeli uml:Property ma zdefiniowany kres dolny jako liczbę całkowitą to cel uml:Property jest cechowany przez krotność posiadający kres dolny równy liczbie całkowitej

Forma synonimiczna: liczba całkowita może być kresem dolnym uml:Property

#### uml:Property może mieć zdefiniowany kres górny jako nieograniczoną liczbę całkowitą

◇[18]  
UML

Definicja: jeżeli uml:Property ma zdefiniowany kres górny jako liczbę całkowitą to cel uml:Property jest cechowany przez krotność posiadający kres górny równy liczbie całkowitej

Forma synonimiczna: nieograniczona liczba całkowita może być kresem górnym uml:Property

#### uml:Type może być typem dla uml:Property

◇[19]  
UML

Definicja: jeżeli uml:Type jest typem dla uml:Property to uml:Type pełni uml:Property w pewnej uml:Association

Forma synonimiczna: uml:Property może być typu uml:Type

#### uml:Property może posiadać wartość domyślną uml:ValueSpecification

◇[20]  
UML

Definicja: jeżeli uml:Property posiada wartość domyślną uml:ValueSpecification to uml:Property jest cechowana przez wartość domyślną, która jest określona jako uml:ValueSpecification

#### uml:Classifier może być generalizacją w uml:Generalization

◇[21]  
UML

Definicja: jeżeli uml:Classifier jest generalizacją uml:Generalization to uml:Classifier jest generalizacją w ramach uml:Generalization

uml:Classifier może być specjalizacją uml:Generalization \* $\diamond$ [22]  
UML

Definicja: jeżeli uml:Classifier jest specjalizacją uml:Generalization to uml:Classifier jest specjalizacją w ramach uml:Generalization

uml:NamedElement może mieć określony uml:VisibilityKind \* $\diamond$ [23]  
UML

Definicja: jeżeli uml:NamedElement ma określony uml:VisibilityKind to uml:NamedElement jest cechowany przez widoczność określoną jako uml:VisibilityKind

### 5.5.2. Częsteczki semantyczne przyjętego metamodelu zunifikowanego języka modelowania

$$*_{\text{UML}}^{\text{klasa}} = \left( \begin{array}{c} *_{\text{UML}}^{\{\diamond 1, \diamond 23\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot \text{[uml:Class]} < \odot \text{[uml:Classifier]}, \\ \odot \text{[uml:Class]} < \odot \text{[uml:NamedElement]}, \\ \diamond \odot \text{[uml:VisibilityKind]} \end{array} \right\} \end{array} \right) \quad (5.82)$$

$$*_{\text{UML}}^{\text{atrybut}} = \left( \begin{array}{c} *_{\text{UML}}^{\{\diamond 12, \diamond 3, \diamond 4, \diamond 5, \diamond 6, \diamond 7, \diamond 8, \diamond 19, \diamond 17, \diamond 18, \diamond 20, \diamond 23\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot \text{[uml:Class]}, \\ \odot \text{[uml:Property]}, \\ \odot \text{[uml:Type]}, \\ \odot \text{[uml:Property]} < \left[ *_{\text{UML}}^{\{\diamond 23\}} \right] \odot \text{[uml:NamedElement]}, \\ \diamond \odot \text{[nieograniczona liczba całkowita]}, \\ \diamond \odot \text{[liczba całkowita]}, \\ \diamond \odot \text{[uml:ValueSpecification]}, \\ \diamond \odot \text{[uml:VisibilityKind]} \end{array} \right\} \end{array} \right) \quad (5.83)$$

$$*_{\text{UML}}^{\text{asocjacja}} = \left( \begin{array}{c} *_{\text{UML}}^{\{\diamond 1, \diamond 23\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot \text{[uml:Association]} < \odot \text{[uml:Classifier]}, \\ \odot \text{[uml:Association]} < \odot \text{[uml:NamedElement]}, \\ \diamond \odot \text{[uml:VisibilityKind]} \end{array} \right\} \end{array} \right) \quad (5.84)$$

$$*_{UML}^{[rola]} = \left( \begin{array}{c} *_{UML}^{\{\diamond 13, \diamond 14, \diamond 15, \diamond 2, \diamond 3, \diamond 4, \diamond 5, \diamond 6, \diamond 7, \diamond 8, \diamond 16, \diamond 17, \diamond 18, \diamond 19, \diamond 23\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \diamond \odot [\underline{uml:Class}], \\ \diamond \odot [\underline{uml:Association}], \\ \odot [\underline{uml:Property}], \\ \odot [\underline{uml:Type}], \\ \diamond \odot [\underline{uml:Property}] < \left[ *_{UML}^{\{\diamond 16\}} \right] \odot [\underline{uml:Property}_1], \\ \diamond \odot [\underline{nieograniczona\ liczba\ ca\krowita}], \\ \diamond \odot [\underline{liczba\ ca\krowita}], \\ \diamond \odot [\underline{uml:Property}_2], \\ \diamond \odot [\underline{uml:VisibilityKind}] \end{array} \right\} \end{array} \right) \quad (5.85)$$

$$*_{UML}^{[klasa\ asocjacyjna]} = \left( \begin{array}{c} \emptyset, \\ \left\{ \begin{array}{l} *_{UML}^{[klasa]}, \\ *_{UML}^{[asocjacja]} \end{array} \right\}, \\ \left\{ \begin{array}{l} \odot [\underline{uml:AssociationClass}] < \odot [\underline{uml:Association}], \\ \odot [\underline{uml:AssociationClass}] < \odot [\underline{uml:Class}], \\ \diamond \odot [\underline{uml:VisibilityKind}] \end{array} \right\} \end{array} \right) \quad (5.86)$$

$$*_{UML}^{[generalizacja]} = \left( \begin{array}{c} *_{UML}^{\{\diamond 11, \diamond 21, \diamond 22\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot [\underline{uml:Generalization}], \\ \odot [\underline{generalizacja}] \triangleleft \left[ *_{AOM}^{[21]} \right] \odot [\underline{uml:Classifier}], \\ \odot [\underline{specjalizacja}] \triangleleft \left[ *_{AOM}^{[22]} \right] \odot [\underline{uml:Classifier}] \end{array} \right\} \end{array} \right) \quad (5.87)$$

$$*_{UML}^{[zbi\o{r}\ generalizacji]} = \left( \begin{array}{c} *_{UML}^{\{\diamond 9, \diamond 10\}}, \\ \left\{ \begin{array}{l} *_{UML}^{[generalizacja_n]} \end{array} \right\}, \\ \left\{ \begin{array}{l} \odot [\underline{uml:Generalization}_n], \\ \odot [\underline{generalizacja}_n], \\ \odot [\underline{specjalizacja}_n] \end{array} \right\} \end{array} \right) \quad (5.88)$$

### 5.5.3. Wzorce mapowania

#### 5.5.3.1. Wzorzec mapowania: kolekcja

Wzorzec mapowania ma postać odwzorowania:

$$*_{AOM}^{[kolekcja]} \mapsto *_{UML_{AOM}}^{[kolekcja]} \quad (5.89)$$

gdzie:

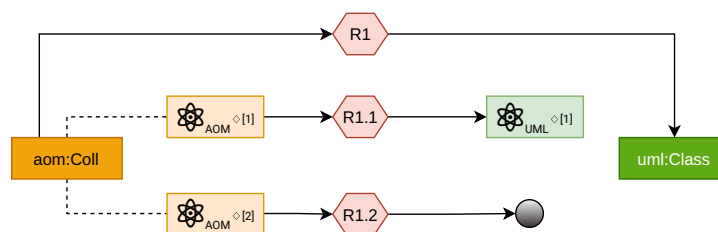
$$*_{UML_{AOM}}^{[kolekcja]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{UML}^{[klasa]} \left[ \begin{array}{c} \diamond 23 \\ *_{UML} \end{array} \right] \right\}, \\ \left\{ \odot \underline{uml:Class} \right\} \end{array} \right) \quad (5.90)$$

Reguły wzorca mapowania (rys. 5.15):

R1 **jest konieczne aby aom:Coll odpowiadało uml:Class**

R1.1 **jeżeli aom:Coll jest abstrakcyjna, to odpowiadająca uml:Class jest abstrakcyjna**

R1.2 **jest niemożliwe, aby to, że aom:Coll jest uninawigowalna, posiadało odpowiednik w UML**



Rysunek 5.15. Diagram obrazujący odwzorowanie  $*_{AOM}^{[kolekcja]} \mapsto *_{UML_{AOM}}^{[kolekcja]}$

#### 5.5.3.2. Wzorzec mapowania: atrybut

Wzorzec mapowania ma postać odwzorowania:

$$*_{AOM}^{[atrybut]} \mapsto *_{UML_{AOM}}^{[atrybut]} \quad (5.91)$$

gdzie:

$$*_{UML_{AOM}}^{[atrybut]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{UML}^{[atrybut]} \right\}, \\ \left\{ \begin{array}{c} \odot \underline{uml:Class}, \\ \odot \underline{uml:Property}, \\ \odot \underline{uml:Type} \end{array} \right\} \end{array} \right) \quad (5.92)$$

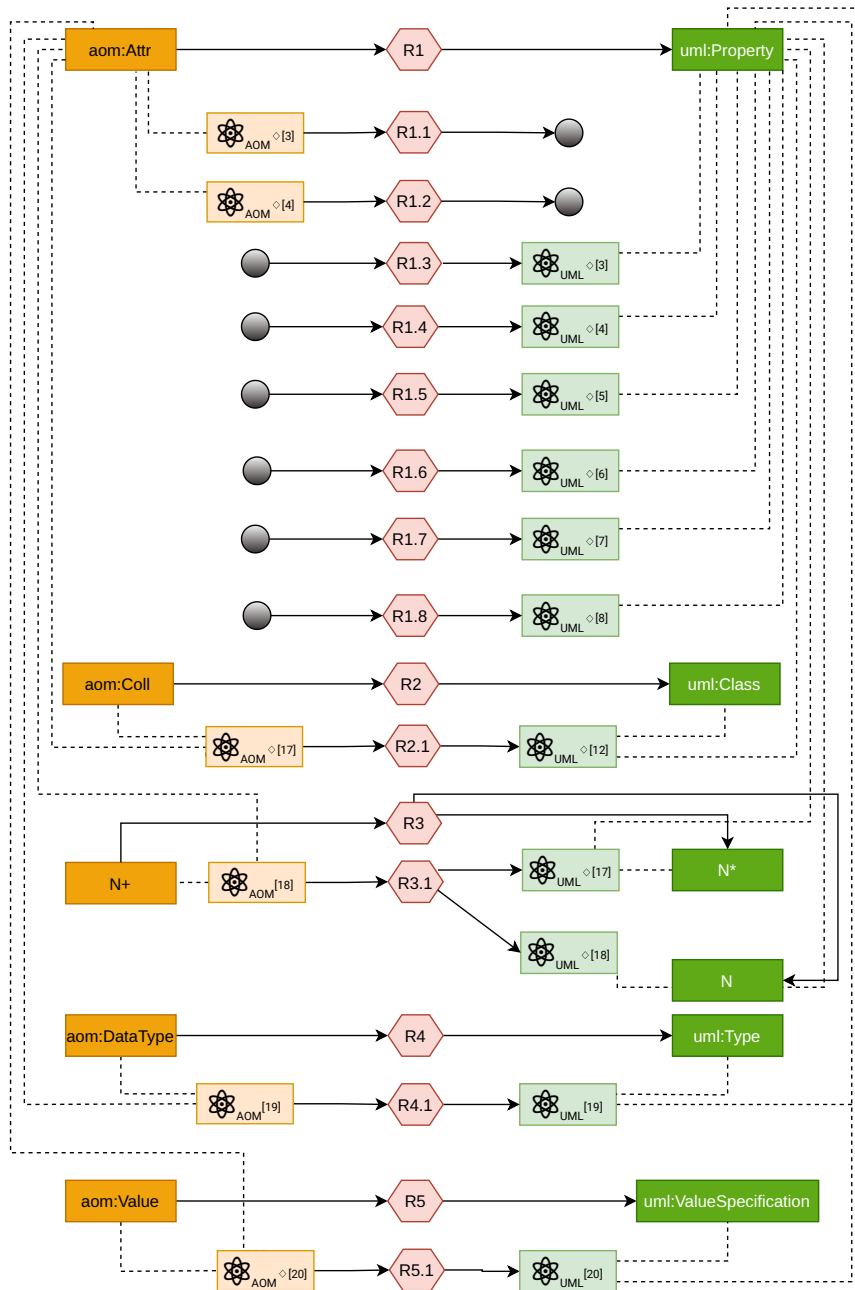
Reguły wzorca mapowania (rys. 5.16):

- R1 jest konieczne aby aom:Attr odpowiadało uml:Property
- R1.1 jest niemożliwe, aby to, że aom:Attr jest dziedziczny, posiadało odpowiednik w UML
- R1.2 jest niemożliwe, aby to, że aom:Attr jest wirtualny, posiadało odpowiednik w UML
- R1.3 jest obowiązkowe, aby uml:Property, która odpowiada aom:Attr miała własność kompozycyjnej agregacji
- R1.4 jest obowiązkowe, aby uml:Property, która odpowiada aom:Attr nie była identyfikatorem
- R1.5 jest obowiązkowe, aby uml:Property, która odpowiada aom:Attr nie była tylko do odczytu
- R1.6 jest obowiązkowe, aby uml:Property, która odpowiada aom:Attr nie była uporządkowana
- R1.7 jest obowiązkowe, aby uml:Property, która odpowiada aom:Attr nie była unikalna
- R1.8 jest obowiązkowe, aby uml:Property, która odpowiada aom:Attr nie była statyczna
- R2 jest konieczne aby aom:Coll odpowiadało uml:Class
- R2.1 jeżeli aom:Coll posiada aom:Attr to odpowiadająca uml:Class posiada atrybut odpowiadający aom:Property
- R3 jest konieczne aby dodatnia liczba naturalna będąca licznikiem aom:Attr odpowiadała liczbie całkowitej będącej kresem dolnym odpowiadającego uml:Property oraz nieograniczonej liczbie całkowitej będącej kresem górnym odpowiadającego uml:Property
- R4 jest konieczne aby aom:DataType odpowiadało uml:Type
- R4.1 uml:Type, które jest typem dla uml:Property odpowiada aom:DataType będącemu typem dla aom:Attr
- R5 jest konieczne aby aom:Value odpowiadało uml:ValueSpecification
- R5.1 aom:Value będąca wartością domyślną dla aom:Attr odpowiada uml:ValueSpecification będąca wartością domyślną dla odpowiadającego uml:Property

### 5.5.3.3. Wzorzec mapowania: asocjacja

Wzorzec mapowania ma postać odwzorowania:

$$\begin{matrix} * & \text{[asocjacja]} \\ \text{AOM} \end{matrix} \mapsto \begin{matrix} * & \text{[kolekcja]} \\ \text{UML}_{\text{AOM}} \end{matrix} \quad (5.93)$$



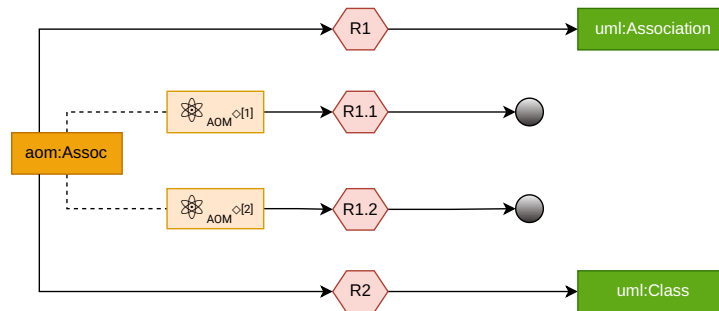
Rysunek 5.16. Diagram obrazujący odwzorowanie  $*_{AOM}^{[atrybut]} \mapsto *_{UML_{AOM}}^{[atrybut]}$

gdzie:

$$*_{UML_{AOM}}^{[asocjacja]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{UML}^{[asocjacja | klasa]} \right\}, \\ \left\{ \odot [uml:Association], \right. \\ \left. \odot [uml:Class] \right\} \end{array} \right) \quad (5.94)$$

Reguły wzorca mapowania (rys. 5.17):

- R1 jeżeli aom:Assoc jest binarna lub jest opisywana przez aom:Coll to odpowiada jej uml:Association
- R1.1 jest niemożliwe, aby to, że aom:Assoc jest abstrakcyjna, posiadało odpowiednik w UML
- R1.2 jest niemożliwe, aby to, że aom:Assoc jest uninawigowalna, posiadało odpowiednik w UML
- R2 jeżeli aom:Assoc jest binarna lub nie jest opisywana przez aom:Coll to odpowiada jej uml:Class



Rysunek 5.17. Diagram obrazujący odwzorowanie  $*_{AOM}^{[asocjacja]} \mapsto *_{UML_{AOM}}^{[asocjacja]}$

#### 5.5.3.4. Wzorzec mapowania: rola

Wzorzec mapowania ma postać odwzorowania:

$$*_{AOM}^{[rola]} \mapsto *_{UML_{AOM}}^{[rola]} \quad (5.95)$$

gdzie:

$$*_{UML_{AOM}}^{[rola]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{UML}^{[rola | asocjacja]} \right\}, \\ \left( \begin{array}{c} \odot [\underline{uml:Association}], \\ \odot [\underline{uml:Property}], \\ \odot [\underline{uml:Class}] < \odot [\underline{uml:Type}], \\ \odot [\underline{nieograniczona\ liczba\ ca\u0142kowita}], \\ \odot [\underline{liczba\ ca\u0142kowita}] \end{array} \right) \end{array} \right) \quad (5.96)$$

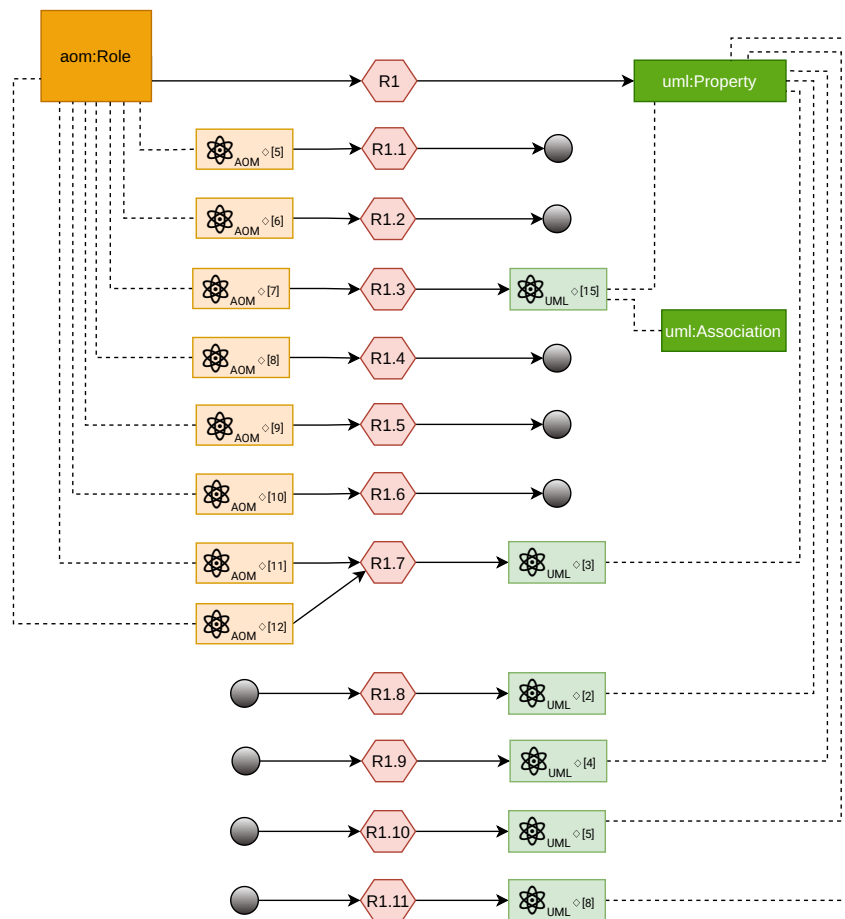
Regu\u0142y wzorca mapowania (rys. 5.18 oraz 5.19):

- R1 jeżeli aom:Assoc posiadaj\u0105ca aom:Role posiada wi\u0119cej ni\u017c 2 aom:Role to aom:Role odpowiada uml:Property
- R1.1 jest niemo\u017cliwe, aby to, \u017ce aom:Role jest wirtualna, posiada\u0142o odpowiednik w UML



- R1.2 jest niemożliwe, aby to, że aom:Role jest dziedziczna, posiadało odpowiednik w UML
- R1.3 jeżeli aom:Role jest uninawigowalna, to odpowiadające uml:Property jest nawigowalna z uml:Assoc
- R1.4 jest niemożliwe, aby to, że aom:Role jest skierowana do właściciela, posiadało odpowiednik w UML
- R1.5 jest niemożliwe, aby to, że aom:Role jest skierowana do celu, posiadało odpowiednik w UML
- R1.6 jest niemożliwe, aby to, że aom:Role jest dwukierunkowa, posiadało odpowiednik w UML
- R1.7 jeżeli aom:Assoc posiadająca aom:Role jest binarna oraz aom:Role<sub>1</sub> ma kompozycję po stronie elementu wiążanego oraz aom:Role<sub>2</sub> ma kompozycję po stronie związku to uml:Property odpowiadająca aom:Role<sub>1</sub> ma własność kompozycyjnej agregacji
- R1.8 jest obowiązkowe, aby uml:Property, która odpowiada aom:Role nie miała własność współdzielonej agregacji
- R1.9 jest obowiązkowe, aby uml:Property, która odpowiada aom:Role nie była identyfikatorem
- R1.10 jest obowiązkowe, aby uml:Property, która odpowiada aom:Role nie była tylko do odczytu
- R1.11 jest obowiązkowe, aby uml:Property, która odpowiada aom:Role nie była statyczna
- R2 jeżeli aom:Assoc posiadająca aom:Role posiada co najwyżej 2 aom:Role to aom:Role odpowiada uml:Association a końcówki roli odpowiadają uml:Property
- R3 jeżeli aom:Assoc posiadająca aom:Role posiada więcej niż 2 aom:Role to aom:Assoc odpowiada uml:Association a w przeciwnym wypadku odpowiada uml:Class
- R3.1 jeżeli aom:Assoc posiada aom:Role to odpowiadające uml:Property jest końcówką uml:Association lub uml:Class jest typem dla odpowiadającej uml:Property
- R4 jest konieczne, aby końcówka roli odpowiadała uml:Class
- R4.1 jeżeli końcówka roli pełni aom:Role to uml:Class jest typem dla uml:Property
- R4.2 jeżeli końcówka roli jest kategoryzowana przez aom:Assoc to aom:Assoc odpowiada uml:Class
- R5 supremum krotności po str. związku nie posiada odpowiednika w UML
- R6 infimum krotności po str. związku nie posiada odpowiednika w UML

- R7 jest konieczne aby supremum krotności po str. elementu wiążanego odpowiadało nieograniczonej liczbie naturalnej
- R7.1 krotność aom:Role mająca kres górny równy nieograniczonej liczbie naturalnej odpowiada nieograniczonej liczbie całkowitej definiującej kres górny uml:Property
- R8 jest konieczne aby infimum krotności po str. elementu wiążanego odpowiadało liczbie naturalnej
- R8.1 krotność aom:Role mająca kres dolny równy liczbie naturalnej odpowiada nieograniczonej liczbie całkowitej definiującej kres dolny uml:Property
- R9 jeżeli aom:Role posiada unikalność równą 1 to odpowiadająca uml:Property jest unikalna

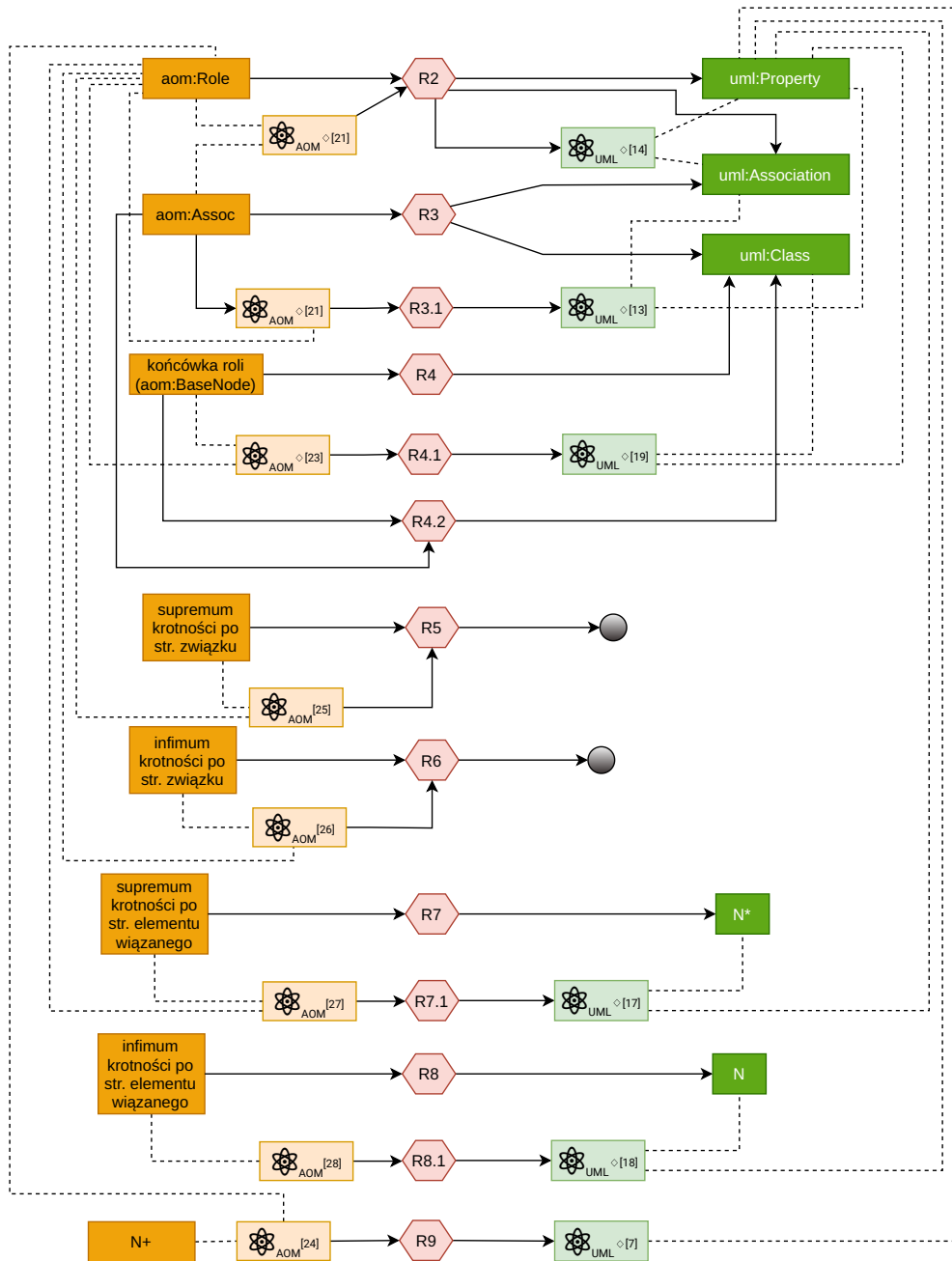


Rysunek 5.18. Diagram obrazujący odwzorowanie  $*_{AOM}^{[rola]} \mapsto *_{UML_{AOM}}^{[rola]}$  (1)

### 5.5.3.5. Wzorzec mapowania: opis asocjacji

Wzorzec mapowania ma postać odwzorowania:

$$*_{AOM}^{[opis\ asocjacji]} \mapsto *_{UML_{AOM}}^{[opis\ asocjacji]} \quad (5.97)$$



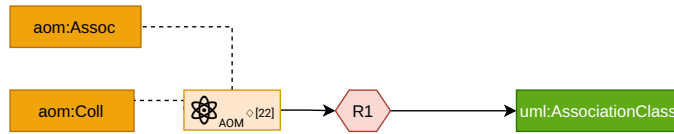
Rysunek 5.19. Diagram obrazujący odwzorowanie  $*_{AOM}^{[rola]} \mapsto *_{UML_{AOM}}^{[rola]}$  (2)

gdzie:

$$*_{UML_{AOM}}^{[opis\ asocjacji]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{UML}^{[klasa\ asocjacyjna]} \right\}, \\ \left\{ \odot_{[uml:AssociationClass]} \right\} \end{array} \right) \quad (5.98)$$

Reguły wzorca mapowania:

R1 jeżeli aom:Assoc jest opisywana przez aom:Coll to jest konieczne, aby aom:Assoc oraz aom:Coll odpowiadało uml:AssociationClass



Rysunek 5.20. Diagram obrazujący odwzorowanie  $*_{AOM}^{[opis\ asocjacji]} \mapsto *_{UML_{AOM}}^{[opis\ asocjacji]}$

### 5.5.3.6. Wzorzec mapowania: opis roli

Opis roli w AOM z technicznego punktu widzenia pozwala określić kolekcję, z której jeden, wybrany obiekt będzie przechowywał dodatkowe dane reprezentujące konkretną rolę. Należy zauważyć, iż jest to opcjonalne przyporządkowanie o krotnościach zero do jeden. Dlatego, analogicznie jak w przypadku EER oraz ORM, dla UML odwzorowanie opisu roli należy traktować jak dodatkowe uml:Property będące typu uml:Class, które odpowiada opisującej aom:Coll.

Wzorzec mapowania ma postać odwzorowania:

$$*_{AOM}^{[opis\ roli]} \mapsto *_{UML_{AOM}}^{[opis\ roli]} \quad (5.99)$$

gdzie:

$$*_{UML_{AOM}}^{[opis\ roli]} = \left( \begin{array}{c} \emptyset, \\ \{ *_{UML}^{[rola]} \}, \\ \left\{ \begin{array}{l} \odot [uml:Class], \\ \odot [uml:Property], \\ \odot [uml:Association], \\ \odot [1] \triangleleft \odot [nieograniczona\ liczba\ całkowita], \\ \odot [0] \triangleleft \odot [liczba\ całkowita] \end{array} \right\} \end{array} \right) \quad (5.100)$$

Reguły mapowania są analogiczne jak dla  $*_{UML_{AOM}}^{[rola]}$ .

### 5.5.3.7. Wzorzec mapowania: dziedziczenie

Wzorzec mapowania ma postać odwzorowania:

$$*_{AOM}^{[dziedziczenie]} \mapsto *_{UML_{AOM}}^{[dziedziczenie]} \quad (5.101)$$

gdzie:

$$*_{\text{UML}_{\text{AOM}}}^{\text{[dziedziczenie]}} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{\text{UML}}^{\text{[generalizacja | asocjacja}_{1,2}]}} \right\}, \\ \left\{ \begin{array}{l} \odot_{\text{uml:Generalization}}, \\ \odot_{\text{[generalizacja]}}, \\ \odot_{\text{[specjalizacja]}}, \\ \odot_{\text{uml:Association}_1}, \\ \odot_{\text{uml:Association}_2} \end{array} \right\} \end{array} \right) \quad (5.102)$$

Reguły wzorca mapowania (rys. 5.21):

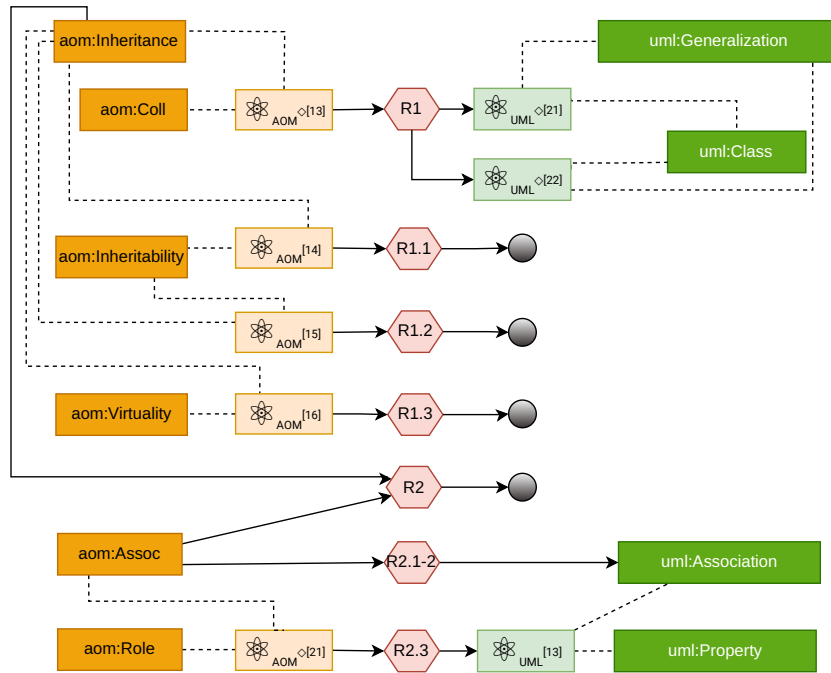
- R1 jeżeli aom:BaseNode<sub>1</sub> dziedziczy po aom:BaseNode<sub>2</sub> oraz aom:BaseNode<sub>1</sub> i aom:BaseNode<sub>2</sub> są specjalizowane jako aom:Coll to jest konieczne, aby aom:Inheritance odpowiadało uml:Generalization oraz aom:BaseNode<sub>1</sub> odpowiada specjalizacji oraz aom:BaseNode<sub>2</sub> odpowiada generalizacji
- R1.1 jest niemożliwe, aby to, że aom:Inheritance jest trybu aom:Inheritability w aspekcie dziedziczenia składowych, posiadało odpowiednik w UML
- R1.2 jest niemożliwe, aby to, że aom:Inheritance jest trybu aom:Inheritability w aspekcie dziedziczenia praw do pełnienia ról, posiadało odpowiednik w UML
- R1.3 jest niemożliwe, aby to, że aom:Inheritance jest trybu aom:Virtuality w zakresie wirtualności, posiadało odpowiednik w UML
- R2 jeżeli aom:BaseNode<sub>1</sub> dziedziczy po aom:BaseNode<sub>2</sub> oraz aom:BaseNode<sub>1</sub> i aom:BaseNode<sub>2</sub> są specjalizowane jako aom:Assoc to nie posiada to odpowiednika w AOM
- R2.1 aom:Assoc specjalizującej aom:BaseNode<sub>1</sub> odpowiada uml:Association<sub>1</sub>
- R2.2 aom:Assoc specjalizującej aom:BaseNode<sub>2</sub> odpowiada uml:Association<sub>2</sub>
- R2.3 jeżeli aom:Inheritance nie jest trybu aom:none w aspekcie dziedziczenia składowych to każdej dziedziczalnej aom:Role posiadanej przez aom:Assoc specjalizującą aom:BaseNode<sub>2</sub> odpowiada uml:Property, które jest końcówką uml:Assoc<sub>1</sub>

### 5.5.3.8. Wzorzec mapowania: BACT

Wzorzec mapowania ma postać odwzorowania:

$$*_{\text{AOM}}^{\text{[BACT]}} \mapsto *_{\text{UML}_{\text{AOM}}}^{\text{[BACT]}} \quad (5.103)$$

gdzie:



Rysunek 5.21. Diagram obrazujący odwzorowanie  $*_{AOM}^{[dziedziczenie]} \mapsto *_{UML_{AOM}}^{[dziedziczenie]}$

$$*_{UML_{AOM}}^{[BACT]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{UML}^{[klasa]} \left[ \begin{array}{c} \diamond 1 \\ UML \end{array} \right] \right\}, \\ \left\{ \odot [uml:Class] \right\} \end{array} \right) \quad (5.104)$$

Reguły wzorca mapowania:

- R1 jest konieczne aby aspekt danych oraz aspekt związku oraz 1 oraz nierozzerwalne połączenie odpowiadały uml:Class
- R1.1 jest konieczne aby odpowiadająca uml:Class nie była abstrakcyjna
- R1.2 jeżeli aspekt związku posiada aom:Role, które nie są nierozzerwalnym połączeniem to każdej takiej aom:Role odpowiada uml:Association
- R1.3 to, że nierozzerwalne połączenie ma kompozycje po stronie związku odpowiada uml:Class
- R1.4 to, że nierozzerwalne połączenie ma kompozycje po stronie elementu związanego odpowiada uml:Class
- R1.5 to, że nierozzerwalne połączenie ma kres górny równy 1 po stronie elementu związanego odpowiada uml:Class
- R1.6 to, że nierozzerwalne połączenie ma kres dolny równy 1 po stronie elementu związanego odpowiada uml:Class

R1.7 to, że nierozzerwalne połączenie ma kres górny równy 1 po stronie związku odpowiada uml:Class

R1.8 to, że nierozzerwalne połączenie ma kres dolny równy 1 po stronie związku odpowiada uml:Class

## 5.6. Podsumowanie

W niniejszym rozdziale zaprezentowano szereg zagadnień dotyczących metody translacji semantyki. Przedstawiono koncepcję metody translacji poprzez zdefiniowanie artefaktów translacyjnych istotnych z punktu widzenia translacji semantyki oraz określenie kolejnych przekształceń pomiędzy artefaktami translacyjnymi. Koncepcję tą zrealizowano dla metamodelu asocjacyjnego poprzez ekstrakcję jego semantyki oraz jej translację do metamodeli EER, ORM oraz UML.

## 6. Metoda ewaluacji translacji semantyki

Niniejszy rozdział przedstawia opracowaną metodę pozwalającą na ocenę jakości translacji semantyki. Ocena ta dotyczy zarówno warstwy definicyjnej translacji, tj. odwzorowań cząsteczek semantycznych metamodeli, jak również warstwy aplikacyjnej i pozwala na ocenę każdej konkretnej translacji modeli danych. W wyniku translacji może dojść do pewnych negatywnych zjawisk, tj. utraty, zmiany lub manifestacji nowej semantyki. Pomiar jakości translacji pozwala na ocenę stopnia podobieństwa semantycznego modeli i metamodeli, co już samo w sobie jest zagadnieniem złożonym i ważnym zarówno z teoretycznego, jak i praktycznego punktu widzenia. Rozwiązanie to zapewnia możliwość ocenę skuteczności przyjętych metod translacji, a co za tym idzie doboru najlepszego sposobu odwzorowywania semantyki. W obszarze teoretycznym zaproponowana metoda pozwala na ocenę metamodeli pod względem ich zdolności do przenoszenia określonej semantyki. W kontekście przedstawionej w poprzednim rozdziale metody translacji, metoda ewaluacji translacji semantyki stanowi bardzo istotne jej uzupełnienie. Obie metody są wzajemnie komplementarne i z metodologicznego punktu widzenia, stanowią spójne rozwiązanie. W tym rozdziale zostaną przedstawione miary jakości translacji dla wzorców mapowania, semantyki metamodeli oraz semantyki modeli.

### 6.1. Miary translacji wzorców mapowania

Poniżej określono miary służące do ewaluacji wzorców mapowania (wyr. 5.8), odwzorowujących cząsteczki semantyczne metamodelu źródłowego w translacyjne cząsteczki semantyczne metamodelu docelowego.

Wzorzec mapowania  $\pi$  obrazuje przekształcenie cząsteczki  $p^s$  w cząsteczkę translacyjną  $p^{s \rightarrow d}$ , która to z kolei definiuje postać odwzorowanej cząsteczki metamodelu docelowego  $p^d$ . Można zatem strukturalnie wyrazić wzorzec mapowania jako parę:

$$\pi = (p^s, p^{s \rightarrow d}) \quad (6.1)$$

W celu uproszczenia zapisu, w definicji metryk stosowane będą następujące konwencje oznaczeń:



- $\otimes_{p^s}$  – multizbiór atomów semantycznych odwzorowywanej cząsteczki wzorca mapowania  $\pi$  (metamodelu źródłowego),
- $\otimes_{p^d}$  – multizbiór atomów semantycznych odwzorowanej cząsteczki wzorca mapowania  $\pi$  (metamodelu docelowego),
- $\odot_{p^s}$  – multizbiór pojęć CLoM, które opisują kategorie metamodelu źródłowego wchodzące w skład odwzorowywanej cząsteczki wzorca mapowania  $\pi$ ,
- $\odot_{p^d}$  – multizbiór pojęć CLoM, które opisują kategorie metamodelu docelowego wchodzące w skład odwzorowanej cząsteczki wzorca mapowania  $\pi$ .

W ramach analizy cząsteczek semantycznych, każdy atom semantyczny należący do  $\otimes_{p^s}$  został przyporządkowany do jednej z trzech grup:

- $\otimes_{p^s}^T$  – przełożone atomy semantyczne wzorca mapowania  $\pi$  (atomy, których semantykę udało się zrealizować we wzorcu mapowania *bez zmiany semantyki*),
- $\otimes_{p^s}^A$  – zmienione atomy semantyczne wzorca mapowania  $\pi$  (atomy, których semantykę udało się zrealizować we wzorcu mapowania *ze zmianą semantyki*),
- $\otimes_{p^s}^L$  – utracone atomy semantyczne wzorca mapowania  $\pi$  (atomy, których semantyki nie udało się zrealizować we wzorcu mapowania, a więc *semantyka została utracona*).

Niezależnie, atomy semantyczne należące do  $\otimes_{p^d}$  zostały przyporządkowane do jednej z dwóch grup:

- $\otimes_{p^d}^R$  – wynikowe atomy semantyczne wzorca mapowania  $\pi$  (atomy, których manifestacja spowodowała była atomami semantycznymi metamodelu źródłowego bądź charakterystyką pojęć metamodelu źródłowego),
- $\otimes_{p^d}^F$  – wymuszone atomy semantyczne wzorca mapowania  $\pi$  (atomy, które nie odwzorowywały żadnej semantyki metamodelu źródłowego, lecz ze względu na budowę metamodelu spowodowały *manifestację nowej semantyki*).

Dodatkowo, analizowano odpowiadające sobie pary pojęć należące do  $\odot_{p^s}$  oraz  $\odot_{p^d}$ . Zbiór odpowiadających sobie par pojęć wzorca mapowania  $\pi$  dla metamodelu źródłowego  $\mathcal{M}^s$  i docelowego  $\mathcal{M}^d$  oznaczony będzie jako  $\odot_{p(s,d)}$ . Dla pojęć, które nie mają odpowiednika po stronie metamodelu docelowego lub źródłowego utworzono pary  $(\odot, \emptyset)$  oraz odpowiednio  $(\emptyset, \odot)$ . Każda para odpowiadających sobie pojęć została odpowiednio przyporządkowana do jednej z czterech grup:

- $\odot_{p(s,d)}^T$  – pary pojęć wzorca mapowania  $\pi \in \Pi_{\mathcal{M}^s \rightarrow \mathcal{M}^d}$ , w których pojęcia są zbieżne, tzn. udało się je odwzorować *bez utraty semantyki*,
- $\odot_{p(s,d)}^A$  – pary pojęć wzorca mapowania  $\pi \in \Pi_{\mathcal{M}^s \rightarrow \mathcal{M}^d}$ , w których pojęcia są rozbieżne, tzn. udało się je odwzorować *ze zmianą semantyki*,

- $\odot_{p(s,d)}^F$  – pary pojęć wzorca mapowania  $\pi \in \Pi_{\mathcal{M}^s \mapsto \mathcal{M}^d}$ , w których pojęcie metamodelu docelowego zostało wymuszone, gdyż nie posiada odpowiednika w metamodelu źródłowym, tzn. pary pojęć powodujące *manifestację nowej semantyki*,
- $\odot_{p(s,d)}^L$  – pary pojęć wzorca mapowania  $\pi \in \Pi_{\mathcal{M}^s \mapsto \mathcal{M}^d}$ , w których pojęcie metamodelu źródłowego zostało utracone, ponieważ nie posiada odpowiednika w metamodelu docelowym, tzn. pary pojęć powodujące *utrata semantyki*.

### 6.1.1. Przełożone atomy semantyczne

Miara przełożonych atomów semantycznych wzorca mapowania (ang. *translated semantic atoms – TSA*) określa stosunek liczby atomów semantycznych metamodelu źródłowego, których semantykę udało się zrealizować we wzorcu mapowania w metamodelu docelowym, do liczby wszystkich atomów semantycznych metamodelu źródłowego:

$$TSA_p = \frac{|\otimes_{p^s}^T|}{|\otimes_{p^s}|} \quad (6.2)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część semantyki metamodelu źródłowego została przeniesiona w sposób precyzyjny, tzn. bez utraty semantyki, bądź jej zmiany.

### 6.1.2. Zmienione atomy semantyczne

Miara zmienionych atomów semantycznych wzorca mapowania (ang. *altered semantic atoms – ASA*) określa stosunek liczby atomów semantycznych metamodelu źródłowego, których semantyka została odwzorowana w metamodelu docelowym ze zmianami, do liczby wszystkich atomów semantycznych metamodelu źródłowego.

$$ASA_p = \frac{|\otimes_{p^s}^A|}{|\otimes_{p^s}|} \quad (6.3)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część semantyki metamodelu źródłowego została przeniesiona, jednakże wprowadzając pewne modyfikacje semantyki wynikowej.

### 6.1.3. Wymuszone atomy semantyczne

Miara wymuszonych atomów semantycznych (ang. *forced semantic atoms – FSA*) określa stosunek liczby atomów semantycznych metamodelu docelowego, których

semantyka została wymuszona przez metamodel docelowy, do liczby wszystkich atomów semantycznych metamodelu docelowego:

$$FSA_p = \frac{|\otimes_p^F|}{|\otimes_p^d|} \quad (6.4)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . W odróżnieniu od miar  $TSA_p$ ,  $ASA_p$ ,  $LSA_p$ , miara ta normalizowana jest ze względu na liczbę wynikowych atomów semantycznych. Im wyższa wartość tej miary, tym większa część semantyki metamodelu docelowego została zmanifestowana bez odpowiadającej semantyki metamodelu źródłowego.

#### 6.1.4. Utracone atomy semantyczne

Miara utraconych atomów semantycznych wzorca mapowania (ang. *lost semantic atoms* –  $LSA$ ) określa stosunek liczby atomów semantycznych metamodelu źródłowego, których semantyka nie została przeniesiona, do liczby wszystkich atomów semantycznych metamodelu źródłowego:

$$LSA_p = \frac{|\otimes_p^L|}{|\otimes_p^s|} \quad (6.5)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część semantyki metamodelu źródłowego została utracona.

Miary  $TSA_p$ ,  $ASA_p$  oraz  $LSA_p$  są komplementarne, co wyrazić można zależnością:

$$TSA_p + ASA_p + LSA_p = 1 \quad (6.6)$$

#### 6.1.5. Przełożone pojęcia

Miara przełożonych pojęć wzorca mapowania (ang. *translated concepts* –  $TC$ ) określa stosunek liczby par pojęć, które są zbieżne, do liczby wszystkich par pojęć:

$$TC_p = \frac{|\odot_p^T|}{|\odot_p^{(s,d)}|} \quad (6.7)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część odpowiadających sobie par pojęć została przeniesiona bez utraty, zmiany bądź dodania nowej semantyki.

### 6.1.6. Zmienione pojęcia

Miara zmienionych pojęć wzorca mapowania (ang. *altered concepts* – *AC*) określa stosunek liczby par pojęć, które są rozbieżne, do liczby wszystkich par pojęć:

$$AC_p = \frac{|\odot_{p^{(s,d)}}^A|}{|\odot_{p^{(s,d)}}|} \quad (6.8)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część odpowiadających sobie par pojęć została przeniesiona, lecz w sposób wprowadzający pewne modyfikacje w ramach odwzorowywania pojęć.

### 6.1.7. Wymuszone pojęcia

Miara wymuszonych pojęć wzorca mapowania (ang. *forced concepts* – *FC*) określa stosunek liczby par pojęć, które zostały wymuszone, do liczby wszystkich par pojęć:

$$FC_p = \frac{|\odot_{p^{(s,d)}}^F|}{|\odot_{p^{(s,d)}}|} \quad (6.9)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym więcej pojęć metamodelu docelowego nie miało odpowiednika w metamodelu źródłowym.

### 6.1.8. Utracone pojęcia

Miara utraconych pojęć wzorca mapowania (ang. *lost concepts* – *LC*) określa stosunek liczby par pojęć, które nie zostały przeniesione do metamodelu docelowego, do liczby wszystkich par pojęć:

$$LC_p = \frac{|\odot_{p^{(s,d)}}^L|}{|\odot_{p^{(s,d)}}|} \quad (6.10)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym więcej pojęć metamodelu źródłowego nie miało odpowiednika w metamodelu docelowym.

Miary  $TC_p$ ,  $AC_p$ ,  $FC_p$  oraz  $LC_p$  są komplementarne, tzn. cechuje je poniższa zależność:

$$TC_p + AC_p + FC_p + LC_p = 1 \quad (6.11)$$

## 6.2. Miary translacji semantyki metamodeli

Miary translacji semantyki metamodeli zostały oparte o zbiory wzorców mapowania  $\Pi_{\mathcal{M}^s \mapsto \mathcal{M}^d}$  (wyr. 5.9). Dla uproszczenia zapisu, zbiór wzorców mapowania  $\Pi_{\mathcal{M}^s \mapsto \mathcal{M}^d}$  oznaczono jako  $\Pi_{s,d}$ .

### 6.2.1. Miary pierwszego poziomu

Miary pierwszego poziomu stanowią agregację miar poszczególnych wzorców mapowania wchodzących w skład zbioru  $\Pi_{s,d}$ . W celu wzmocnienia znaczenia wzorców mapowania o bardziej złożonej semantyce, zastosowano normalizację opartą o średnią ważoną, gdzie wagą jest odpowiednio liczba atomów semantycznych lub par odpowiadających sobie pojęć danego wzorca mapowania  $\pi$ .

#### 6.2.1.1. Przełożone atomy semantyczne

Miara przełożonych atomów semantycznych  $TSA_{s,d}$  agreguje miarę  $TSA_p$  ze względu na zbiór wzorców mapowania  $\Pi_{s,d}$ .

$$TSA_{s,d} = \frac{\sum_{\pi \in \Pi_{s,d}} |\otimes_{p^s}| \cdot TSA_p}{\sum_{\pi \in \Pi_{s,d}} |\otimes_{p^s}|} \quad (6.12)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część semantyki metamodelu źródłowego została przeniesiona w sposób precyzyjny, tzn. bez utraty semantyki, bądź jej zmiany.

#### 6.2.1.2. Zmienione atomy semantyczne

Miara zmienionych atomów semantycznych  $ASA_{s,d}$  agreguje miarę  $ASA_p$  ze względu na zbiór wzorców mapowania  $\Pi_{s,d}$ .

$$ASA_{s,d} = \frac{\sum_{\pi \in \Pi_{s,d}} |\otimes_{p^s}| \cdot ASA_p}{\sum_{\pi \in \Pi_{s,d}} |\otimes_{p^s}|} \quad (6.13)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część semantyki metamodelu źródłowego została przeniesiona, jednakże wprowadzając pewne modyfikacje semantyki wynikowej.

### 6.2.1.3. Wymuszone atomy semantyczne

Miara wymuszone atomów semantycznych  $FSA_{s,d}$  agreguje miarę  $FSA_p$  ze względu na zbiór wzorców mapowania  $\Pi_{s,d}$ .

$$FSA_{s,d} = \frac{\sum_{\pi \in \Pi_{s,d}} |\otimes_{p^d}| \cdot FSA_p}{\sum_{\pi \in \Pi_{s,d}} |\otimes_{p^d}|} \quad (6.14)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym więcej atomów semantycznych metamodelu docelowego nie miało odpowiednika w metamodelu źródłowym.

### 6.2.1.4. Utracone atomy semantyczne

Miara utraconych atomów semantycznych  $LSA_{s,d}$  agreguje miarę  $LSA_p$  ze względu na zbiór wzorców mapowania  $\Pi_{s,d}$ .

$$LSA_{s,d} = \frac{\sum_{\pi \in \Pi_{s,d}} |\otimes_{p^s}| \cdot LSA_p}{\sum_{\pi \in \Pi_{s,d}} |\otimes_{p^s}|} \quad (6.15)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część semantyki metamodelu źródłowego została utracona.

Miary  $TSA_{s,d}$ ,  $ASA_{s,d}$  oraz  $LSA_{s,d}$  są komplementarne, co wyrazić można zależnością:

$$TSA_{s,d} + ASA_{s,d} + LSA_{s,d} = 1 \quad (6.16)$$

### 6.2.1.5. Przełożone pojęcia

Miara przełożonych pojęć  $TC_{s,d}$  agreguje miarę  $TC_p$  ze względu na zbiór wzorców mapowania  $\Pi_{s,d}$ .

$$TC_{s,d} = \frac{\sum_{\pi \in \Pi_{s,d}} |\odot_{p^{(s,d)}}| \cdot TC_p}{\sum_{\pi \in \Pi_{s,d}} |\odot_{p^{(s,d)}}|} \quad (6.17)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część odpowiadających sobie par pojęć została przeniesiona bez utraty, zmiany bądź dodania nowej semantyki.

### 6.2.1.6. Zmienione pojęcia

Miara zmienionych pojęć  $AC_{s,d}$  agreguje miarę  $AC_p$  ze względu na zbiór wzorców mapowania  $\Pi_{s,d}$ .

$$AC_{s,d} = \frac{\sum_{\pi \in \Pi_{s,d}} |\odot_{p(s,d)}| \cdot AC_p}{\sum_{\pi \in \Pi_{s,d}} |\odot_{p(s,d)}|} \quad (6.18)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część par pojęć została przeniesiona w sposób przyporządkowujący sobie pojęcia o odmiennej semantyce.

### 6.2.1.7. Wymuszone pojęcia

Miara wymuszonych pojęć  $FC_{s,d}$  agreguje miarę  $FC_p$  ze względu na zbiór wzorców mapowania  $\Pi_{s,d}$ .

$$FC_{s,d} = \frac{\sum_{\pi \in \Pi_{s,d}} |\odot_{p(s,d)}| \cdot FC_p}{\sum_{\pi \in \Pi_{s,d}} |\odot_{p(s,d)}|} \quad (6.19)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym więcej pojęć metamodelu docelowego nie miało odpowiednika w metamodelu źródłowym.

### 6.2.1.8. Utracone pojęcia

Miara utraconych pojęć  $LC_{s,d}$  agreguje miarę  $LC_p$  ze względu na zbiór wzorców mapowania  $\Pi_{s,d}$ .

$$LC_{s,d} = \frac{\sum_{\pi \in \Pi_{(s,d)}} |\odot_{p(s,d)}| \cdot LC_p}{\sum_{\pi \in \Pi_{s,d}} |\odot_{p(s,d)}|} \quad (6.20)$$

Miara ta jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym więcej pojęć metamodelu źródłowego nie miało odpowiednika w metamodelu docelowym.

Miary  $TC_{s,d}$ ,  $AC_{s,d}$ ,  $FC_{s,d}$  oraz  $LC_{s,d}$  są komplementarne, tzn. cechuje je poniższa zależność:

$$TC_{s,d} + AC_{s,d} + FC_{s,d} + LC_{s,d} = 1 \quad (6.21)$$

## 6.2.2. Miary semantyki wynikowej

W poprzedniej sekcji przedstawiono osiem miar podstawowych charakteryzujących translację. Na ich bazie można utworzyć miary pochodne, mające za zadanie odkryć zależności występujące pomiędzy poszczególnymi miarami. W celu ukazania stopnia,

w jakim semantyka została przeniesiona z metamodelu źródłowego do docelowego utworzono miary semantyki wynikowej (ang. *resulting semantics* – *RS*).

Miarę semantyki wynikowej atomów semantycznych  $RSSA_{s,d}$  określono jako:

$$RSSA_{s,d} = \frac{TSA_{s,d}}{TSA_{s,d} + ASA_{s,d} + FSA_{s,d} + LSA_{s,d}} \quad (6.22)$$

Miara ta przyjmuje wartości z przedziału  $[0, 1]$  oraz odzwierciedla stosunek wartości miary przełożonych atomów semantycznych do sumy wartości wszystkich miar atomów semantycznych (przełożonych, zmienionych, wymuszonych, utraconych). Im wyższa wartość tej miary, tym wyższa względna jakość atomów semantycznych w metamodelu docelowym (jakość rozumiana jako stopień poprawnie przełożonych atomów semantycznych).

Z kolei miarę semantyki wynikowej odpowiadających sobie par pojęć  $RSC_{s,d}$  określono jako:

$$RSC_{s,d} = \frac{TC_{s,d}}{TC_{s,d} + AC_{s,d} + FC_{s,d} + LC_{s,d}} \quad (6.23)$$

Miara ta przyjmuje wartości z przedziału  $[0, 1]$  oraz odzwierciedla stosunek wartości miary przełożonych pojęć do sumy wartości wszystkich miar dla par pojęć (przełożonych, zmienionych, wymuszonych, utraconych). Im wyższa wartość tej miary, tym wyższa względna jakość pojęć w metamodelu docelowym (jakość rozumiana jako stopień poprawnie przełożonych pojęć).

Finalna miara semantyki wynikowej  $RS_{s,d}$  stanowi średnią arytmetyczną  $RSSA_{s,d}$  oraz  $RSC_{s,d}$

$$RS_{s,d} = \frac{RSSA_{s,d} + RSC_{s,d}}{2} \quad (6.24)$$

### 6.3. Miary ewaluacji semantyki modeli

Miary ewaluacji semantyki modeli służą do ewaluacji jakości translacji modelu wyrażonego w metamodelu źródłowym zwanego dalej modelem źródłowym do modelu wyrażonego w metamodelu docelowym – zwanego konsekwentnie modelem docelowym.

W celu oznaczenia tych modeli, w metrykach użyto następującego zapisu:

$m^s$  – model źródłowy,

$m^d$  – model docelowy.

Podczas konkretyzacji cząsteczek semantycznych metamodelu źródłowego, atomy możliwe muszą zostać albo aktywowane, albo dezaktywowane. Z tego po-



wodu miary w dziedzinie modeli są odmienne od tych w dziedzinie metamodeli, gdyż odnoszą się tylko do atomów i pojęć, które są konieczne lub aktywowane.

Fakt, iż cząsteczka semantyczna  $cp$  jest konkretyzacją cząsteczki semantycznej  $p$  istniejącą w ramach modelu  $m$  wyrażonym w metamodelu  $\mathcal{M}$  można wyrazić jako:

$$\ast_{m \triangleleft \mathcal{M}}^{[cp]} \triangleleft \ast_{\mathcal{M}}^{[p]}. \quad (6.25)$$

Zbiór skonkretyzowanych cząsteczek semantycznych modelu  $m$  wyrażonego w metamodelu  $\mathcal{M}$  oznaczono symbolem  $\mathcal{P}_{m \triangleleft \mathcal{M}}$ , zatem:

$$\ast_{m \triangleleft \mathcal{M}}^{[cp]} \in \mathcal{P}_{m \triangleleft \mathcal{M}} \quad (6.26)$$

lub równoznaczne dla uproszczenia

$$cp \in \mathcal{P}_{m \triangleleft \mathcal{M}}. \quad (6.27)$$

Konkretyzowane cząsteczki podlegają odwzorowaniu według dobranych wzorców mapowania  $\pi_m \in \Pi_{m^s \rightarrow m^d}$ . Strukturalnie, dobrany wzorzec mapowania można reprezentować jako parę skonkretyzowanej cząsteczki semantycznej modelu źródłowego  $cp^s$  oraz skonkretyzowanej cząsteczki translacyjnej  $cp^{s \rightarrow d}$  określającej postać skonkretyzowanej cząsteczki modelu docelowego  $cp^d$ :

$$\pi_m = (cp^s, cp^{s \rightarrow d}) \quad (6.28)$$

Dla każdego dobranego wzorca mapowania  $\pi_m$  można zatem wyznaczyć:

- $\ast_{cp^s}$  – multizbiór koniecznych lub aktywowanych atomów semantycznych odwzorowywanej, skonkretyzowanej cząsteczki (modelu źródłowego),
- $\ast_{cp^d}$  – multizbiór koniecznych lub aktywowanych atomów semantycznych odwzorowywanej, skonkretyzowanej cząsteczki (modelu docelowego),
- $\odot_{cp^s}$  – multizbiór koniecznych lub aktywowanych pojęć CLoM, które opisują pojęcia modelu źródłowego wchodzące w skład odwzorowywanej, skonkretyzowanej cząsteczki  $cp^s$  dobranego wzorca mapowania  $\pi_m$ ,
- $\odot_{cp^d}$  – multizbiór koniecznych lub aktywowanych pojęć CLoM, które opisują pojęcia modelu docelowego, wchodzące w skład odwzorowanej, skonkretyzowanej cząsteczki  $cp^d$  określonej ze względu na skonkretyzowaną cząsteczkę translacyjną dobranego wzorca mapowania  $\pi_m$ .

Analogicznie do abstrakcyjnych cząsteczek, analiza każdego atomu skonkretyzowanej cząsteczki semantycznej modelu źródłowego skutkowałą przypisaniem go do

jednej z trzech grup:  $\otimes_{cp^s}^T$ ,  $\otimes_{cp^s}^A$ ,  $\otimes_{cp^s}^L$  a w przypadku cząsteczek modelu docelowego do dwóch grup:  $\otimes_{cp^d}^R$ ,  $\otimes_{cp^d}^F$ . Metoda przyporządkowania do owych grup jest analogiczna jak w przypadku cząsteczek abstrakcyjnych, tzn. opiera się o analizę pełnego przełożenia semantyki, przełożenia ze zmianą, wymuszenia, bądź utraty semantyki.

Podobnie jak w przypadku metamodeli, grupowane są również odpowiadające sobie pary pojęć modelu w ramach każdej z instancji wzorca mapowania. Zbiór odpowiadających sobie par pojęć wzorca mapowania  $p$  dla modelu źródłowego  $m^s$  i docelowego  $m^d$  oznaczono jako  $\odot_{cp(s,d)}$ . Dla pojęć, które nie mają odpowiednika po stronie któregośkolwiek z modeli utworzono pary  $(\odot, \emptyset)$  oraz odpowiednio  $(\emptyset, \odot)$ . Każda para odpowiadających pojęć została odpowiednio przyporządkowana do jednej z czterech grup o znaczeniu analogicznym jak w przypadku metamodeli:  $\odot_{cp(s,d)}^T$ ,  $\odot_{cp(s,d)}^A$ ,  $\odot_{cp(s,d)}^F$ ,  $\odot_{cp(s,d)}^L$ , grupując pary pojęć odwzorowane kolejno: bez utraty semantyki, ze zmianą semantyki, manifestując nową semantykę, z utratą semantyki.

### 6.3.1. Miary pierwszego poziomu

Miary pierwszego poziomu oceny jakości translacji semantyki modelu określone zostały jako stosunek sumy licznosci multizbiorów grupujących pary pojęć / atomy semantyczne wybranego typu do sumy licznosci pełnych multizbiorów par pojęć / atomów semantycznych dla wszystkich dobranych wzorców mapowania.

#### 6.3.1.1. Przełożone atomy semantyczne modelu

$$TSA_{m^s, m^d} = \frac{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\otimes_{cp^s}^T|}{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\otimes_{cp^s}|} \quad (6.29)$$

Miara przełożonych atomów semantycznych modelu  $TSA_{m^s, m^d}$  jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część semantyki modelu źródłowego została przeniesiona w sposób precyzyjny, tzn. bez utraty semantyki, bądź jej zmiany.

#### 6.3.1.2. Zmienione atomy semantyczne modelu

$$ASA_{m^s, m^d} = \frac{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\otimes_{cp^s}^A|}{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\otimes_{cp^s}|} \quad (6.30)$$

Miara zmienionych atomów semantycznych modelu  $ASA_{m^s, m^d}$  jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część semantyki modelu źródłowego została przeniesiona do modelu docelowego, jednakże wprowadzając pewne modyfikacje semantyki.

### 6.3.1.3. Wymuszone atomy semantyczne modelu

$$FSA_{m^s, m^d} = \frac{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\otimes_{cp^d}^F|}{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\otimes_{cp^d}^F|} \quad (6.31)$$

Miara wymuszone atomów semantycznych  $FSA_{m^s, m^d}$  jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym więcej atomów semantycznych modelu docelowego nie powstało poprzez odwzorowanie modelu źródłowego.

### 6.3.1.4. Utracone atomy semantyczne modelu

$$LSA_{m^s, m^d} = \frac{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\otimes_{cp^s}^L|}{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\otimes_{cp^s}^L|} \quad (6.32)$$

Miara utraconych atomów semantycznych  $LSA_{m^s, m^d}$  jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część semantyki modelu źródłowego została utracona.

### 6.3.1.5. Przełożone pojęcia modelu

$$TC_{m^s, m^d} = \frac{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\odot_{cp^{(s,d)}}^T|}{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\odot_{cp^{(s,d)}}^T|} \quad (6.33)$$

Miara przełożonych pojęć  $TC_{m^s, m^d}$  jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część odpowiadających sobie par pojęć została przeniesiona bez utraty, zmiany bądź dodania nowej semantyki.

### 6.3.1.6. Zmienione pojęcia modelu

$$AC_{m^s, m^d} = \frac{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\odot_{cp^{(s,d)}}^A|}{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\odot_{cp^{(s,d)}}^A|} \quad (6.34)$$

Miara zmienionych pojęć modelu  $AC_{m^s, m^d}$  jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym większa część pojęć modelu źródłowego została odwzorowana, jednakże wprowadzając pewne modyfikacje ich semantyki w ramach modelu docelowego.

### 6.3.1.7. Wymuszone pojęcia modelu

$$FC_{m^s, m^d} = \frac{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\odot_{cp^{(s,d)}}^F|}{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\odot_{cp^{(s,d)}}^F|} \quad (6.35)$$

Miara wymuszonych pojęć modelu  $FC_{m^s, m^d}$  jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym więcej pojęć modelu docelowego nie ma odpowiednika w modelu źródłowym.

### 6.3.1.8. Utracone pojęcia modelu

$$LC_{m^s, m^d} = \frac{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\ominus_{cp(s,d)}^L|}{\sum_{\pi_m \in \Pi_{m^s \rightarrow m^d}} |\ominus_{cp(s,d)}|} \quad (6.36)$$

Miara przełożonych pojęć modelu  $LC_{m^s, m^d}$  jest znormalizowana, ma charakter liniowy i przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość miary, tym więcej pojęć modelu źródłowego nie miało odpowiednika w modelu docelowym.

### 6.3.2. Miary semantyki wynikowej modelu

Definicja miar pochodnych dla modeli, agregujących miary podstawowe, jest analogiczna jak dla metamodeli. Aby przedstawić stopień przeniesienia semantyki z modelu źródłowego do docelowego utworzono miary semantyki wynikowej modelu.

Miarę semantyki wynikowej atomów semantycznych  $RSSA_{m^s, m^d}$  określono jako stosunek wartości miary przełożonych atomów semantycznych modelu do sumy wartości wszystkich czterech miar dotyczących atomów semantycznych modelu:

$$RSSA_{m^s, m^d} = \frac{TSA_{m^s, m^d}}{TSA_{m^s, m^d} + ASA_{m^s, m^d} + FSA_{m^s, m^d} + LSA_{m^s, m^d}} \quad (6.37)$$

Miara ta przyjmuje wartości z przedziału  $[0, 1]$ . Im wyższa wartość  $RSSA_{m^s, m^d}$ , tym wyższa względna jakość atomów semantycznych w modelu docelowym  $m^d$  (jakość rozumiana jako stopień poprawnie przełożonych atomów semantycznych).

Analogicznie, dla par pojęć modelu określono miarę semantyki wynikowej  $RSC_{m^s, m^d}$ :

$$RSC_{m^s, m^d} = \frac{TC_{m^s, m^d}}{TC_{m^s, m^d} + AC_{m^s, m^d} + FC_{m^s, m^d} + LC_{m^s, m^d}} \quad (6.38)$$

Miara ta przyjmuje wartości z przedziału  $[0, 1]$  oraz odzwierciedla stosunek wartości miary poprawnie przełożonych par pojęć modelu do sumy wartości wszystkich miar dotyczących par pojęć modelu (przełożonych, zmienionych, wymuszonych, utraconych). Im wyższa wartość tej miary, tym wyższa względna jakość pojęć w modelu docelowym (jakość rozumiana jako stopień poprawnie przełożonych pojęć).

Końcowa miara semantyki wynikowej modelu  $RS_{m^s, m^d}$  stanowi średnią arytmetyczną  $RSSA_{m^s, m^d}$  oraz  $RSC_{m^s, m^d}$ .

$$RS_{m^s, m^d} = \frac{RSSA_{m^s, m^d} + RSC_{m^s, m^d}}{2} \quad (6.39)$$

## 6.4. Podsumowanie

W rozdziale przedstawiono 30 miar oceny jakości translacji semantyki. Miary te dotyczą oceny poszczególnych wzorców mapowania, odwzorowywania semantyki metamodeli, jak również konkretnych modeli. W każdej z tych warstw podjęto zagadnienie oceny czterech aspektów: dokładnej translacji, zmiany, utraty oraz manifestacji nowej semantyki. Pomiar odbywał się w przestrzeni atomów semantycznych oraz odpowiadających sobie par pojęć.

Miary były poddane normalizacji oraz zostały zaprojektowane w sposób uwzględniający liniowość. Na bazie podstawowych miar zaproponowano złożone miary dążące do jednowymiarowej oceny jakości translacji, zarówno w dziedzinie metamodeli jak również modeli. Opracowane miary posiadają niską (liniową) złożoność obliczeniową oraz posiadają jednoznaczną interpretację. W związku z tym, stanowią przydatne narzędzie służące do ewaluacji jakości przeprowadzanych translacji, która jest zagadnieniem istotnym, co zostało wskazane we wstępie niniejszego rozdziału.

## 7. Ewaluacja translacji semantyki metamodeli i dyskusja otrzymanych wyników

Rozdział ten zawiera wyniki przeprowadzonej oceny jakości translacji semantyki AOM na metamodele EER, ORM oraz UML. Ewaluacja została przeprowadzona zgodnie z miarami zaproponowanymi w rozdziale 6. Ocenę przeprowadzono zarówno dla poszczególnych wzorców mapowania opartych o zaproponowane cząsteczki semantyczne metamodelu asocjacyjnego, jak również w postaci zagregowanej, tzn. uwzględniając całe metamodele.

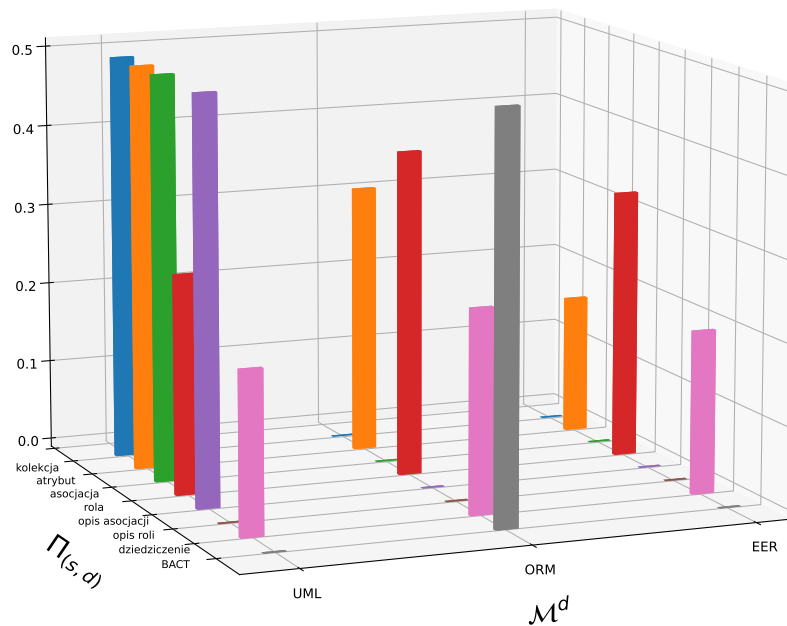
### 7.1. Miary pierwszego poziomu

Tablica 7.1. Wyniki ewaluacji miary  $TSA_p$

$\mathcal{M}^d$	kolekcja	atrybut	asocjacja	rola	opis asocjacji	opis roli	dziedziczenie	BACT
EER	0.0	0.17	0.0	0.33	0.0	0.0	0.20	0.0
ORM	0.0	0.33	0.0	0.40	0.0	0.0	0.25	0.5
UML	0.5	0.50	0.5	0.27	0.5	0.0	0.20	0.0

W kontekście miary  $TSA_p$  (rys. 7.1, tab. 7.1) należy zauważyć, iż cząsteczka semantyczna *opisu roli* posiada wartość tej miary równą 0.0 dla każdego z metamodeli docelowych. Oznacza to, że wszystkie atomy semantyczne związane z tą cząsteczką są nieprzekładalne w sposób bezpośredni. Kolejnym wartym zauważenia faktem jest ocena odwzorowania cząsteczki *opisu asocjacji* na UML, która posiada wartość miary  $TSA_p$  wynoszącą 0.5. Wynika to z faktu, iż odwzorowaniem atomu semantycznego  $\otimes_{\text{AOM}}^{[22]}$  jest pojęcie [uml:AssociationClass](#), którego semantyka pokrywa semantykę tego atomu.

Dla miary  $ASA_p$  (rys. 7.2, tab. 7.2), która reprezentuje zmienione atomy semantyczne, należy zauważyć wartości 0.0 dla wszystkich metamodeli w zakresie cząsteczki semantycznej *kolekcji* i *asocjacji*. Dla cząsteczki *opisu asocjacji*, można zauważyć, iż wartości miar  $TSA_p$  oraz  $ASA_p$  są przeciwne dla wszystkich metamodeli docelowych, tzn. gdy wartość  $TSA_p$  jest równa 1.0, to  $ASA_p$  jest równa 0.0 – i odwrotnie. Obrazuje to sytuację, w ramach której atomy cząsteczki semantycznej



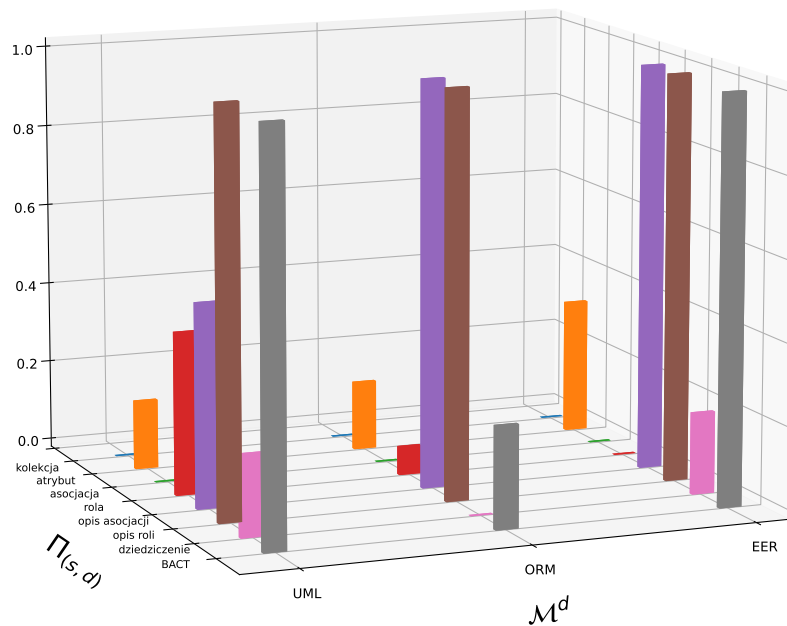
Rysunek 7.1. Wyniki ewaluacji miarą  $TSA_p$

Tablica 7.2. Wyniki ewaluacji miary  $ASA_p$

$\mathcal{M}^d$	kolekcja	atrybut	asocjacja	rola	opis asocjacji	opis roli	dziedziczenie BACT
EER	0.0	0.33	0.0	0.00	1.0	1.0	0.2
ORM	0.0	0.17	0.0	0.07	1.0	1.0	0.0
UML	0.0	0.17	0.0	0.40	0.5	1.0	0.2

zostały zrealizowane w każdym z metamodeli bez utraty semantyki. Należy dodatkowo zauważyć, iż cząsteczka *opisu roli* została w każdym z metamodeli zrealizowana w pełni ze zmianami, więc wartość miary  $ASA_p$  wynosi 1.0. Wynika to z przyjętej konwencji realizacji wzorców mapowania *opisu roli* polegającej na odwzorowaniu tej konstrukcji metamodelu asocjacyjnego w postaci pojęć reprezentujących rolę (dla EER: [eer:Role](#), dla ORM: [orm:Role](#), dla UML: [uml:Property](#)). Z tego też względu, semantyka *opisu roli* została przeniesiona, jednakże ze zmianami.

Ewaluacja przeprowadzona za pomocą miary  $FSA_p$  (rys. 7.3, tab. 7.3) dotyczy wymuszeń dla atomów semantycznych cząsteczek wynikowych. Charakterystyczna cząsteczka semantyczna w świetle tej miary to *kolekcja*. Można zauważyć, iż dla



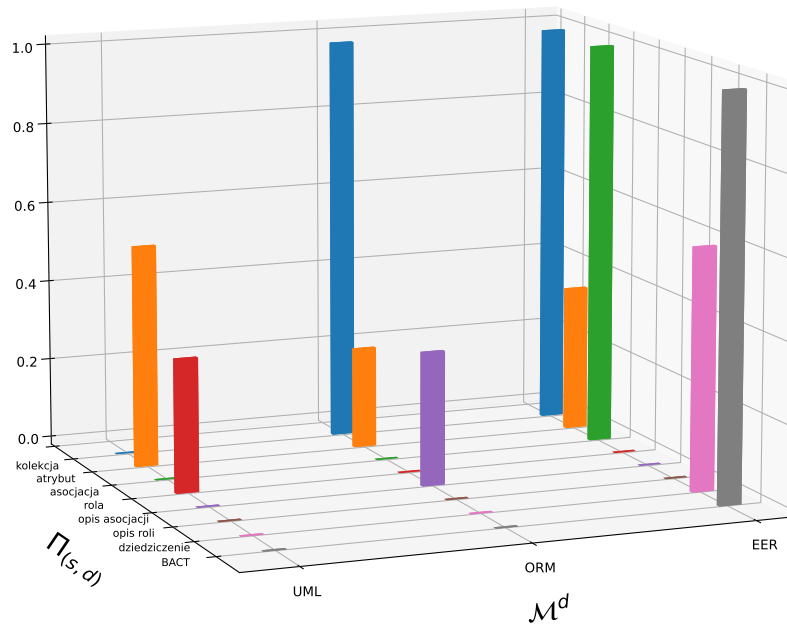
Rysunek 7.2. Wyniki ewaluacji miarą  $ASA_p$

Tablica 7.3. Wyniki ewaluacji miary  $FSA_p$

$\mathcal{M}^d$	kolekcja	atrybut	asocjacja	rola	opis asocjacji	opis roli	dziedziczenie BACT	
EER	1.0	0.36	1.0	0.00	0.00	0.0	0.6	1.0
ORM	1.0	0.25	0.0	0.00	0.33	0.0	0.0	0.0
UML	0.0	0.55	0.0	0.33	0.00	0.0	0.0	0.0

metamodeli EER i ORM wartość tej miary wynosi 1.0, a dla UML 0.0. Wynika to z faktu, iż zarówno EER jak również ORM wymaga zaakceptowania dodatkowej semantyki koniecznej do wyrażenia dla odpowiedników podstawowej kategorii reprezentującej typ bytu. W przypadku UML, bardzo wiele atomów semantycznych ma charakter opcjonalny, toteż rzadziej wymagana jest akceptacja dodatkowych, wymuszonych atomów. Wyjątkiem jest tutaj jedynie cząsteczka *atrybutu*, która wprowadza konieczność wymuszenia semantyki związanej z bardziej precyzyjnym odzwierciedleniem atomów semantycznych cząstki semantycznej *atrybutu*. Dla przykładu, wymagana jest akceptacja semantyki związanej z tym, iż uml:Property odpowiadające





Rysunek 7.3. Wyniki ewaluacji miarą  $FSA_p$

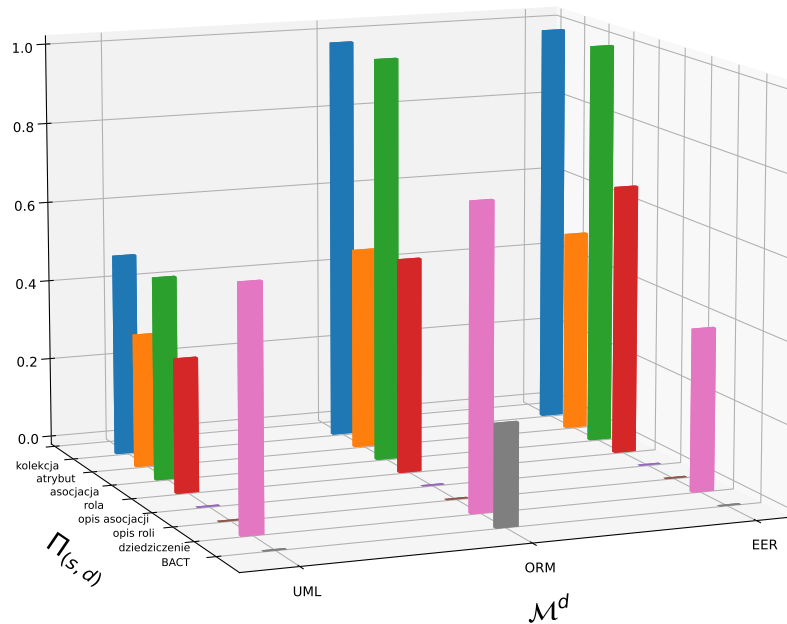
[aom:Attr](#) nie może być identyfikatorem, nie może być tylko do odczytu, etc. (zob. reguły R1.3-R1.8 wzorca mapowania  $*_{AOM}^{[atrybut]} \mapsto *_{UML_{AOM}}^{[atrybut]}$ ).

Tablica 7.4. Wyniki ewaluacji miary  $LSA_p$

$M^d$	kolekcja	atrybut	asocjacja	rola	opis asocjacji	opis roli	dziedziczenie	BACT
EER	1.0	0.50	1.0	0.67	0.0	0.0	0.40	0.00
ORM	1.0	0.50	1.0	0.53	0.0	0.0	0.75	0.25
UML	0.5	0.33	0.5	0.33	0.0	0.0	0.60	0.00

Ewaluacja przeprowadzona za pomocą miary  $LSA_p$  (rys. 7.4, tab. 7.4) dotyczy atomów semantycznych, których semantyka została w procesie translacji utracona. Dla cząsteczek semantycznych *opisu asocjacji* i *opisu roli* wartość tej miary wynosi 0.0, zatem żaden atom semantyczny nie został w pełni utracony. W przypadku EER i ORM, cząsteczkami, które utraciły względnie najwięcej atomów semantycznych były *kolekcja* i *asocjacja*, natomiast dla UML *dziedziczenie*.

Na rysunku 7.5 oraz w tabeli 7.5 przedstawiono wyniki ewaluacji miarą  $TC_p$ . Wychodząc od AOM, metamodels EER, ORM, UML posiadają stuprocentową bazę



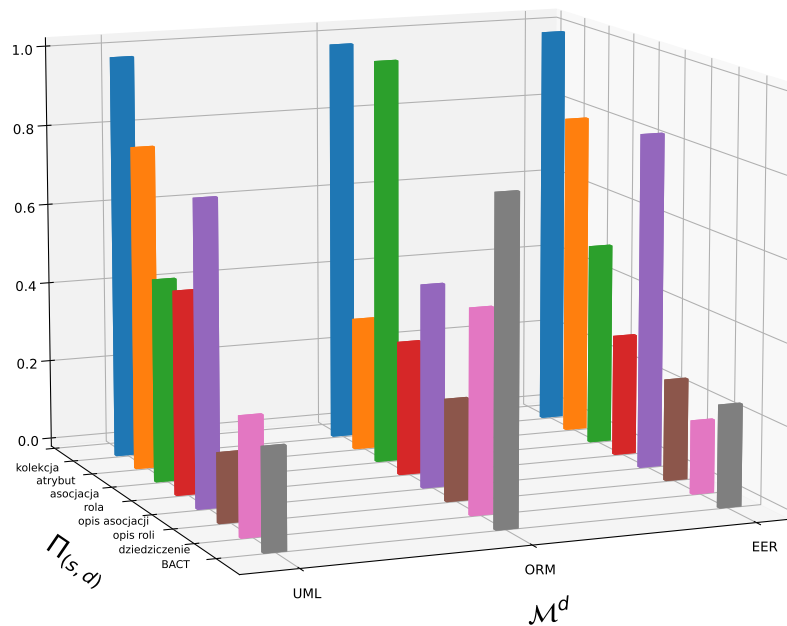
Rysunek 7.4. Wyniki ewaluacji miarą  $LSA_p$

Tablica 7.5. Wyniki ewaluacji miary  $TC_p$

$\mathcal{M}^d$	kolekcja	atrybut	asocjacja	rola	opis asocjacji	opis roli	dziedziczenie	BACT
EER	1.0	0.80	0.5	0.30	0.83	0.25	0.18	0.25
ORM	1.0	0.33	1.0	0.33	0.50	0.25	0.50	0.80
UML	1.0	0.80	0.5	0.50	0.75	0.17	0.29	0.25

pojęciową dla cząsteczki semantycznej *kolekcji*. Oznacza to, iż odpowiadające sobie pojęcia w ramach realizacji wzorców mapowania dla cząsteczki semantycznej *kolekcji* zostały dobrane w taki sposób, iż odpowiadają tym samym pojęciom CLoM. Należy dodatkowo zauważyć, iż brak wartości 0.0 w tabeli 7.5 oznacza, iż każda z cząsteczek semantycznych bazuje na wspólnej (mniej lub bardziej) przestrzeni pojęciowej.

Na rysunku 7.6 oraz w tabeli 7.6 przedstawiono wyniki ewaluacji przy użyciu miary  $AC_p$ . Można zauważyć, iż dla cząsteczek semantycznych *kolekcji* oraz *opisu roli* dla każdego z metamodeli wartość miary wynosi 0.0. W przypadku kolekcji wynika to z faktu, iż odwzorowane pojęcia sobie odpowiadały. Natomiast w przypadku *opisu roli*, pary pojęć zostały albo przełożone (tab. 7.5), wymuszone (tab. 7.7), bądź



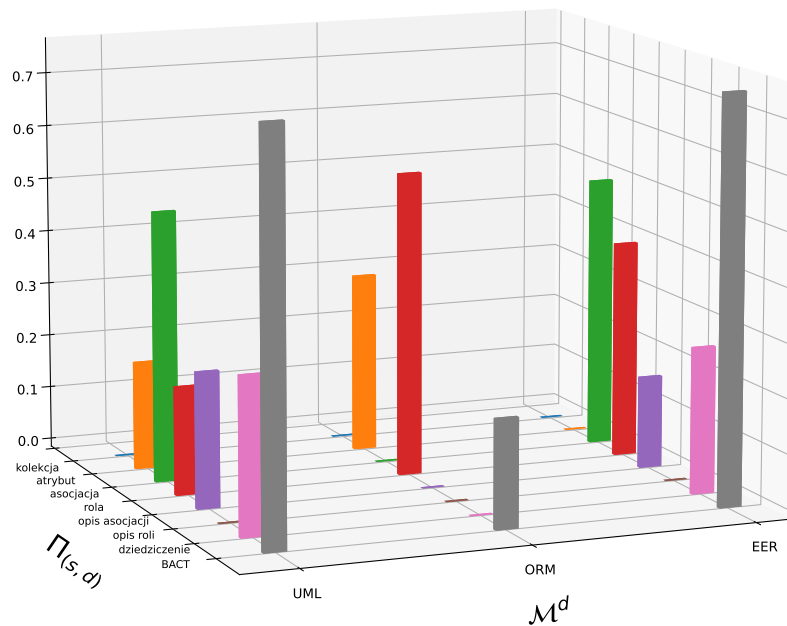
Rysunek 7.5. Wyniki ewaluacji miarą  $TC_p$

Tablica 7.6. Wyniki ewaluacji miary  $AC_p$

$\mathcal{M}^d$	kolekcja	atrybut	asocjacja	rola	opis asocjacji	opis roli	dziedziczenie	BACT
EER	0.0	0.00	0.5	0.40	0.17	0.0	0.27	0.75
ORM	0.0	0.33	0.0	0.56	0.00	0.0	0.00	0.20
UML	0.0	0.20	0.5	0.20	0.25	0.0	0.29	0.75

utraczone (tab. 7.8) – nie było jednak sytuacji w której odwzorowywane pojęcia zmieniały semantykę. Największa zmiana semantyki w zakresie odpowiadających sobie pojęć wystąpiła dla cząsteczki semantycznej *BACT*. W ramach odwzorowania *BACT* na metamodely UML i EER doszło do sytuacji, gdy pojęcia pełniące funkcje w CLoM inne niż typ bytu, były przyporządkowywane pojęciom metamodelu pełniącym funkcję typu bytu. W związku z tym, doszło do relatywnie dużej zmiany semantyki na gruncie odpowiadających sobie pojęć.

Rysunek 7.7 oraz tabela 7.7 przedstawiają wyniki ewaluacji przy użyciu miary  $FC_p$ . Miara przyjmuje największe wartości dla cząsteczki *opisu roli*. Wynika to z faktu związanego z realizacją wzorców mapowania dla tej cząsteczki we wszystkich



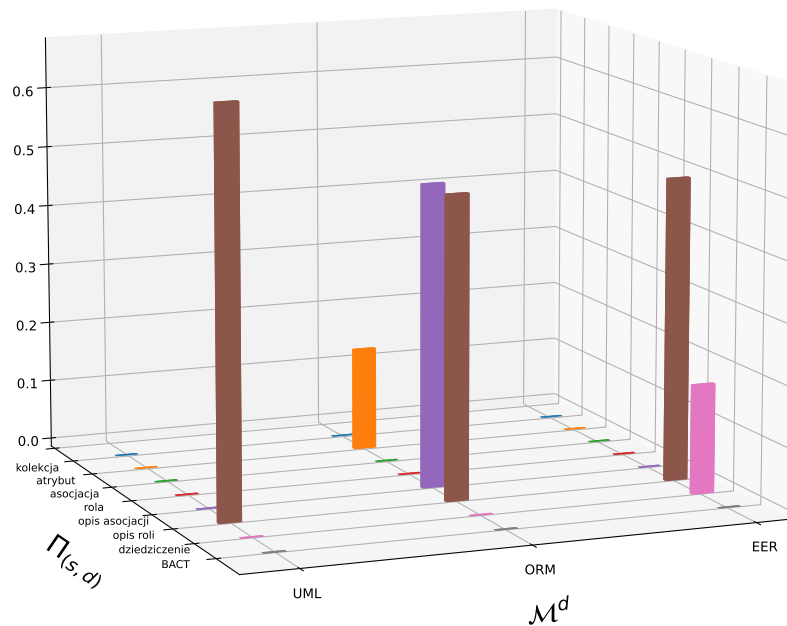
Rysunek 7.6. Wyniki ewaluacji miarą  $AC_p$

Tablica 7.7. Wyniki ewaluacji miary  $FC_p$

$\mathcal{M}^d$	kolekcja	atrybut	asocjacja	rola	opis asocjacji	opis roli	dziedziczenie BACT	
EER	0.0	0.00	0.0	0.0	0.0	0.50	0.18	0.0
ORM	0.0	0.17	0.0	0.0	0.5	0.50	0.00	0.0
UML	0.0	0.00	0.0	0.0	0.0	0.67	0.00	0.0

metamodelach docelowych. Implementacja ta polegała na utworzeniu odwzorowania [aom:Role](#) w pojęcie docelowe reprezentujące pojęciową rolę (dla EER: [eer:Role](#), dla ORM: [orm:Role](#), dla UML: [uml:Property](#)). Zostało to wymuszone przez wzorzec mapowania, ze względu na brak konstrukcji semantycznej reprezentującej opis roli w metamodelach docelowych.

Na rysunku 7.8 oraz w tabeli 7.8 przedstawiono wyniki ewaluacji przy użyciu miary  $LC_p$ . Dla cząsteczek semantycznych *kolekcja*, *asocjacja*, *opis asocjacji* oraz *BACT*, dla każdego z metamodeli docelowych wartość miary  $LC_p$  wynosi 0.0. Oznacza to, że w procesie odwzorowywania pojęć w ramach tych cząsteczek żadne pojęcie nie zostało utracone. Cząsteczką semantyczną, która uzyskała największą wartość



Rysunek 7.7. Wyniki ewaluacji miarą  $FC_p$

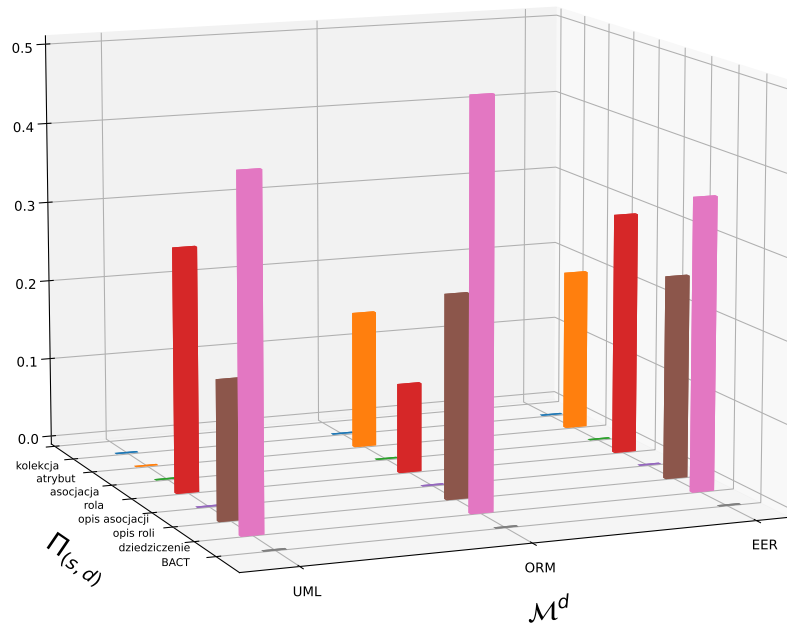
Tablica 7.8. Wyniki ewaluacji miary  $LC_p$

$\mathcal{M}^d$	kolekcja	atrybut	asocjacja	rola	opis asocjacji	opis roli	dziedziczenie	BACT
EER	0.0	0.20	0.0	0.30	0.0	0.25	0.36	0.0
ORM	0.0	0.17	0.0	0.11	0.0	0.25	0.50	0.0
UML	0.0	0.00	0.0	0.30	0.0	0.17	0.43	0.0

dla tej miary jest *dziedziczenie*. Wynika to z względnie wysokiej liczby pojęć pomocniczych w ramach asocjacyjnego mechanizmu dziedziczenia *asocjacji* i *kolekcji*, które nie mają odpowiedników w metamodelach docelowych.

## 7.2. Miary translacji semantyki metamodeli

Na rys. 7.9 oraz w tabeli 7.9 przedstawiono wartości zagregowanych miar  $TSA_{s,d}$ ,  $ASA_{s,d}$ ,  $FSA_{s,d}$ ,  $LSA_{s,d}$ ,  $TC_{s,d}$ ,  $AC_{s,d}$ ,  $FC_{s,d}$ ,  $LC_{s,d}$ , dla wszystkich metamodeli docelowych.



Rysunek 7.8. Wyniki ewaluacji miarą  $LC_p$

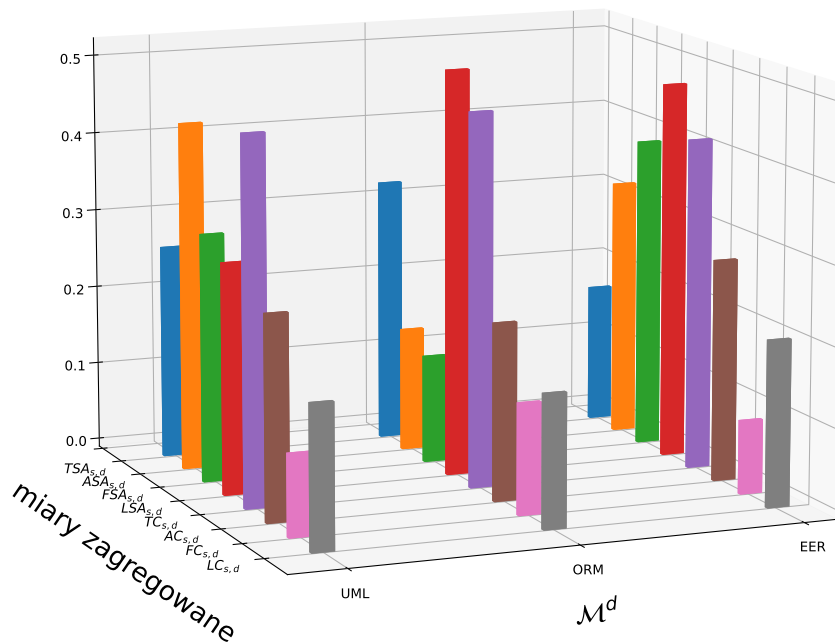
Tablica 7.9. Wyniki ewaluacji dla miar zagregowanych

$\mathcal{M}^d$	$TSA_{s,d}$	$ASA_{s,d}$	$FSA_{s,d}$	$LSA_{s,d}$	$TC_{s,d}$	$AC_{s,d}$	$FC_{s,d}$	$LC_{s,d}$
EER	0.174	0.324	0.391	0.476	0.418	0.279	0.093	0.208
ORM	0.333	0.156	0.136	0.512	0.471	0.223	0.139	0.167
UML	0.270	0.440	0.313	0.291	0.463	0.257	0.103	0.180

Miara  $TSA_{s,d}$  jest miarą najważniejszą ze względu na to, iż określa poziom bezpośrednio przełożonych atomów semantycznych. Wartość tej miary jest zdecydowanie największa dla metamodelu docelowego ORM (prawie dwa razy więcej od EER). Świadczy to o największej zgodności semantyki pomiędzy AOM a ORM w zakresie atomów semantycznych.

Miara  $ASA_{s,d}$  posiada zbliżone wartości dla EER oraz UML oraz zdecydowanie niższą wartość dla ORM. Oznacza to że poziom zmienności atomów semantycznych AOM podczas translacji jest podobny dla EER i UML oraz najniższy dla ORM.

W zakresie miary  $FSA_{s,d}$ , metamodel ORM powodował najmniejsze wymuszenia w zakresie nowej semantyki podczas translacji. Niewystępowanie wymuszeń w zakresie atomów semantycznych wynika z konstrukcji metamodelu. Konstrukcja ta może



Rysunek 7.9. Wyniki ewaluacji miarami zagregowanymi

powodować rozdrobnienie semantyki na poszczególne atomy, które w ramach cząstek mogą mieć możliwość występować opcjonalnie. ORM posiada wiele konstrukcji o charakterze opcjonalnie ograniczającym podstawowe, proste pojęcia metamodelu, dlatego też uzyskał względnie niską wartość tej miary.

Miara  $LSA_{s,d}$ , która określa utraconą semantykę posiada najwyższą wartość dla ORM, a najniższą dla UML. Jest to dokładnie odwrotnie do miary  $ASA_{s,d}$ . Należy zauważyć, że fakt ten pokrywa się z intuicją, tzn. w przypadku dużej utraty semantyki oczekuje się małego jej przełożenia i zmiany, oraz odwrotnie.

Podczas ewaluacji miarą  $TC_{s,d}$  największą wartość uzyskał ORM. Jest to zbieżne z wynikami ewaluacji miarą  $TSA_{s,d}$ . Należy jednak dostrzec fakt, iż różnica pomiędzy wartościami miary  $TC_{s,d}$  dla ORM oraz EER jest mniejsza niż w przypadku miary  $TSA_{s,d}$ . Wynika to z odmiennej semantyki tych miar, mimo iż obie dotyczą przełożenia semantyki z metamodelu źródłowego na docelowy. W przypadku  $TC_{s,d}$  należy zauważyć, iż mierzony jest dobór odpowiadających sobie pojęć metamodeli. Fakt uzyskania względnie wysokiej wartości przez EER oznacza, iż metamodel ten posiada pewną ilość pojęć, które semantycznie odpowiadają pojęciom asocjacyjnym.

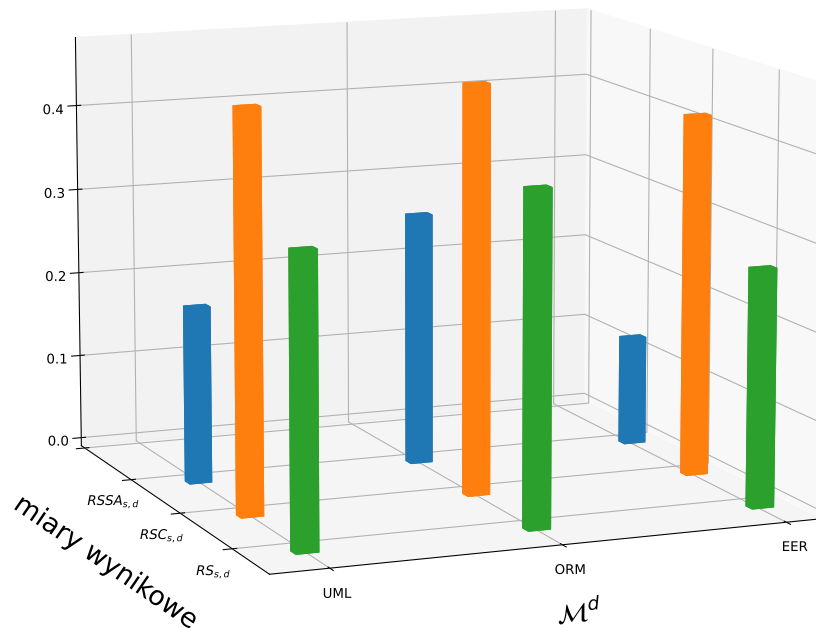
Jednakże niska wartość miary  $TSA_{s,d}$  wskazuje na fakt, iż zdolność tych pojęć do wyrażania określonej semantyki charakterystycznej dla AOM jest niewielka.

Ewaluacja miarami  $AC_{s,d}$ ,  $FC_{s,d}$ ,  $LC_{s,d}$  cechuje się niską wariancją dla przyjętych metamodeli docelowych. Oznacza to, że w zakresie zmienianych, wymuszanych oraz traconych w procesie translacji pojęć, każdy przyjętych metamodeli może być podobnie postrzegany.

### 7.3. Miary semantyki wynikowej

Tablica 7.10. Wyniki ewaluacji dla miar wynikowych

$\mathcal{M}^d$	$RSSA_{s,d}$	$RSC_{s,d}$	$RS_{s,d}$
EER	0.128	0.419	0.273
ORM	0.293	0.471	0.382
UML	0.205	0.461	0.333



Rysunek 7.10. Wyniki ewaluacji miarami semantyki wynikowej:  $RSSA_{s,d}$ ,  $RSC_{s,d}$ ,  $RSC_{s,d}$

W tabeli 7.10 oraz na rysunku 7.10 przedstawiono miary semantyki wynikowej:  $RSSA_{s,d}$ ,  $RSC_{s,d}$ ,  $RS_{s,d}$ . Są to miary najwyższego poziomu, ponieważ zostały zbu-



dowane z miar zagregowanych. Każda z tych miar wyraża stosunek wartości miary semantyki przełożonej na metamodel docelowy, bez utraty lub modyfikacji do sumy wartości wszystkich miar dotyczących atomów semantycznych (dla miary  $RSSA_{s,d}$ ) lub par pojęć (dla  $RSC_{s,d}$ ). Miara  $RS_{s,d}$  jest średnią arytmetyczną  $RSSA_{s,d}$  i  $RSC_{s,d}$ .

Ewaluacja miarą  $RSSA_{s,d}$  wskazuje na fakt, iż translacja na metamodel ORM wykazuje najwyższy poziom zgodności semantycznej z AOM. Natomiast zdecydowanie mniejszą zgodność wykazuje EER.

Należy zwrócić uwagę na interesującą własność w zakresie miar  $RSSA_{\mathcal{M}^s, \mathcal{M}^d}$  i  $RSC_{s,d}$  w kontekście wszystkich metamodeli docelowych. Z analizy wynika, iż porządek powstały poprzez zestawienie metamodeli docelowych według obu miar jest tożsamy, tj. metamodeli docelowe wykazują korelację w zakresie odpowiadających sobie par pojęć oraz realizacji atomów semantycznych.

Uwzględniając łącznie miary  $RSSA_{s,d}$  i  $RSC_{s,d}$ , metamodel EER wykazuje najniższą wartość dla miary  $RS_{s,d}$ . Metamodel ORM w zakresie obu miar  $RSSA_{s,d}$  i  $RSC_{s,d}$  osiągnął największą wartość, a tym samym również dla miary  $RS_{s,d}$ , co należy interpretować, iż uwzględniając zarówno poziom pojęciowy jak i poziom atomów semantycznych, zachowuje on relatywnie największą zgodność z metamodelem asocjacyjnym.

## 7.4. Podsumowanie

Przeprowadzona ewaluacja wskazała na szereg interesujących własności procesu translacji. Warty uwagi jest fakt, iż pewne cząsteczki semantyczne metamodelu asocjacyjnego posiadają lepsze odwzorowania w innych metamodelach (np. *kolekcja*, *rola*) niż inne cząsteczki (np. *dziedziczenie*). Obserwacja takich cząsteczek pozwala na wskazanie unikalnych właściwości metamodelu asocjacyjnego. Analizując miary semantyki wynikowej można również zauważyć, iż w ogólnym przypadku wartości miar dotyczących atomów semantycznych są niższe niż te dotyczące pojęć. Wynika to z oczywistego faktu, iż pojęcia w ramach reprezentacji semantycznych metamodeli są bytami lepiej odwzorowywalnymi niż atomy semantyczne, które przenoszą jednostkową, niepodzielną część semantyki.

Należy dodatkowo zauważyć, iż opracowane miary okazały się przydatne zarówno w zakresie określenia poszczególnych własności semantyki podczas translacji, tj. jej przeniesienia, zmiany, dodania lub utraty. Zaproponowane miary wynikowe mogą posłużyć również jako narzędzie pozwalające na m.in. określenie stopnia podobieństwa metamodeli w zakresie ich pojemności semantycznej.

## 8. Studium przypadku

W celu zobrazowania asocjacyjnych wzorców modelowania oraz translacji semantyki metamodelu asocjacyjnego w praktyce, przeprowadzono studium przypadku modelu. Wykorzystany model stanowi rozwiązanie faktycznego problemu modelowania z zakresu reprezentacji wiedzy w postaci sieci semantycznych.

### 8.1. Przedstawienie modelu

Model, który zostanie omówiony w ramach studium przypadku, jest fragmentem większego modelu: SKB. SKB stanowi hybrydowy system reprezentacji wiedzy, cechuje go modularna budowa. Poszczególne moduły SKB są dedykowane reprezentacji wiedzy określonego rodzaju:

**Structural Module (SM<sup>SKB</sup>)** – moduł strukturalny – stanowi zbiór podstawowych elementów definiujących strukturalne właściwości wiedzy, w tym dostarcza mechanizmy tworzenia pojęć i związków pomiędzy nimi,

**Ontological Core Module (OCM<sup>SKB</sup>)** – moduł rdzenia ontologicznego – przechowuje informacje o pojęciach oraz podstawowe powiązania pomiędzy nimi, takie jak *klasa-cecha*, *klasa-instancja*, *właściwość-cecha-wartość* [94],

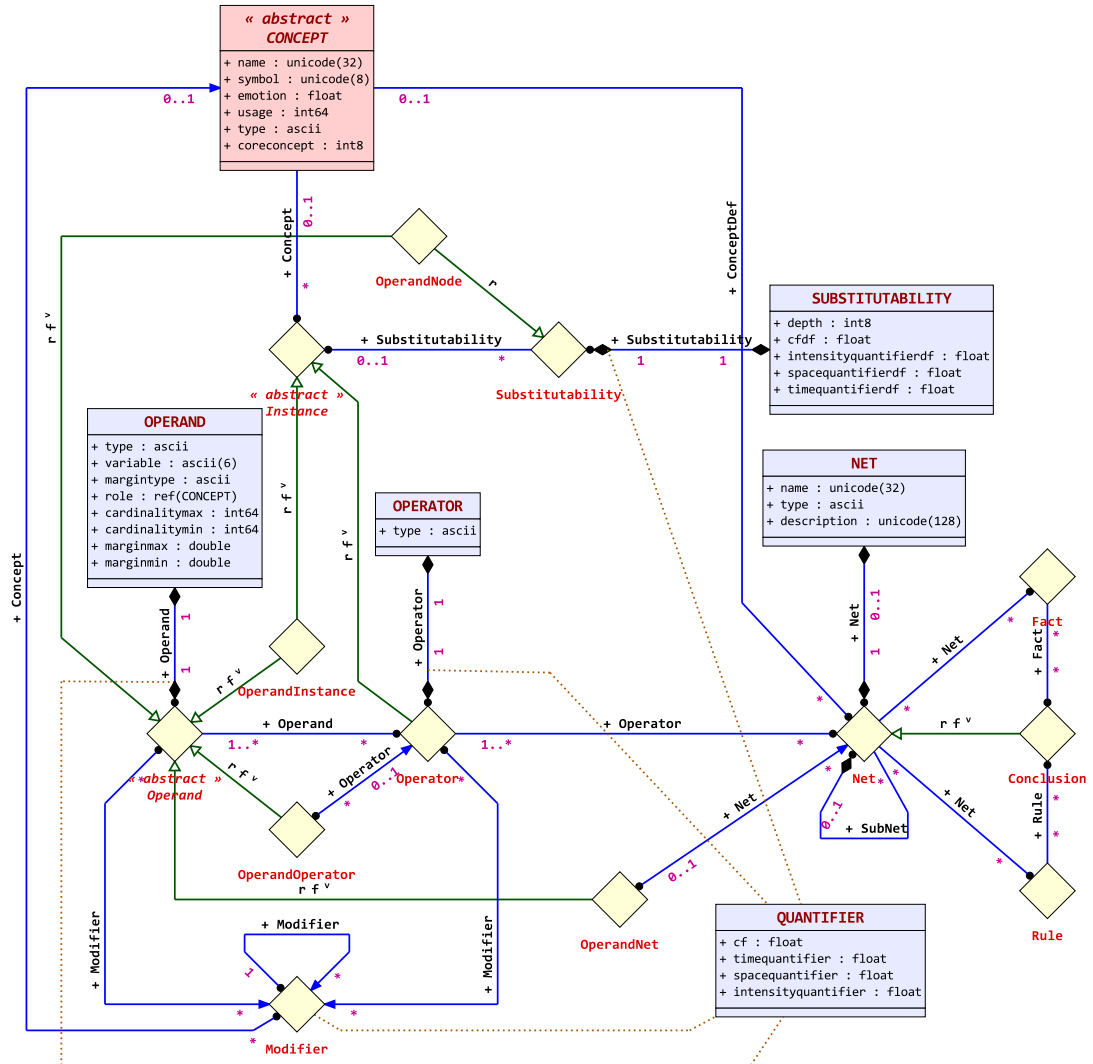
**Relationship Module (RM<sup>SKB</sup>)** – moduł związków – przechowuje informacje o związkach jako struktury hipergrafowe, zarówno w aspekcie intensjonalnym, jak również ekstensjonalnym [92],

**Cyclic Value Ranges Module (CVRM<sup>SKB</sup>)** – moduł zakresów cyklicznych wartości – przechowuje informacje o złożonych definicjach wartości o charakterze cyklicznym, interwałowym, np. w celu reprezentacji czasu [90],

**Dimension & Space Module (DSM<sup>SKB</sup>)** – moduł wymiarów i przestrzeni – przechowuje informacje odnośnie zdefiniowanych przestrzeni, które składają się z wymiarów, w celu możliwości wyrażenia zaistnienia pozostałych elementów SKB w ramach określonej przestrzeni,

**Linguistic Module (LM<sup>SKB</sup>)** – moduł lingwistyczny – przechowuje informacje w zakresie leksykalnych identyfikatorów dla pojęć, aby dać możliwość wydajnego odnoszenia się do nich przez analityków wiedzy, zapewnia również wsparcie dla wielojęzyczności [93],

**Extended Semantic Network Module (ESNM<sup>SKB</sup>)** – moduł rozszerzonych sieci semantycznych – przechowuje złożoną strukturalnie wiedzę w postaci sieci składających się z reguł i faktów [77, 78, 95].



Rysunek 8.1. Diagram AML zawierający model danych ESNM<sup>SKB</sup>. Źródło: [77]

Studium przypadku zostanie przeprowadzone w oparciu o model ESNM<sup>SKB</sup> (rys. 8.1). Rozszerzenie sieci semantycznych w SKB polega na wprowadzeniu kilku nowych koncepcji, m.in. ujednoznaczenie związku *ISA*, dowolność w definiowaniu powiązań, zdolność do ograniczania liczebności, kwantyfikatory i operatory logiczne, reprezentacja informacji niekompletnej i niepewnej. Wyrażenia w ramach ESNM<sup>SKB</sup> oparte są o konstrukcję *operator-operand*:

$$r(p_1, p_2, \dots, p_n) \quad (8.1)$$

gdzie:

- $r$  – operator,
- $p_k$  – operand  $k \in 1..n$ .

Należy zauważyć, iż reprezentacja takich struktur reprezentacji wiedzy pozwala efektywnie wykorzystać w warstwie modelu złożone wyrażenia oparte o związki  $n$ -arne oraz ich role. W związku z tym, analiza właściwości artefaktów powstałych w procesie modelowania jest bardzo interesująca i pozwala zwrócić uwagę na kluczowe elementy samego metamodelu asocjacyjnego. W rozdziale nie zawarto opisu semantyki samego modułu, gdyż wybiega ona poza ramy pracy oraz została szczegółowo opisana w publikacjach [77, 78, 95].

## 8.2. Opis modelu w świetle wzorców modelowania asocjacyjnego

Przedstawiony model składa się z:

- 6 kolekcji,
- 13 asocjacji, w tym:
  - 4 asocjacji unarnych,
  - 2 asocjacji binarnych,
  - 7 asocjacji  $n$ -arnych,
- 20 ról (definiowanych, nieodziedziczonych),
- 8 związków dziedziczenia,
- 1 opisu asocjacji,
- 3 opisów roli.

W kolejnych podrozdziałach omówiono instancje wzorców modelowania, opisanych w rozdziale 4, które zostały zidentyfikowane w modelu.

### 8.2.1. Lista

Do budowy modelu wykorzystano dwukrotnie wzorzec listy. Pierwsze wykorzystanie dotyczy zamodelowania *sieci* jako listy *operatorów*, drugie natomiast *instancji* jako listy *podstawień*.

#### 8.2.1.1. Sieć jako lista operatorów

$$\diamond Net [*] \xrightarrow{+Operator} [1..*] \diamond Operator \quad (8.2)$$

Sieć, jako lista operatorów, przypisuje asocjacji  $\diamond Net$  funkcję GRUPY oraz GRUPOWANIA. Sieć, jest związkiem, który łączy zbiorowość operatorów w jedną całość. Z kolei asocjacja  $\diamond Operator$  w ramach tej instancji wzorca, pełni rolę ELEMENTU.

Instancję wzorca cechuje własność możliwa zapełnienie listy, ze względu na konieczność znajdowania się co najmniej jednego operatora w sieci.

### 8.2.1.2. Instancja jako lista podstawień

$$\diamond Instance^{\emptyset} [0..1] \xrightarrow{+Substitutability} [1..*] \diamond Substitutability \quad (8.3)$$

Lista podstawień, którą stanowi instancja stanowi również wystąpienie wzorca modelowania listy. W tym przypadku rolę uczestnika GRUPY oraz związku GRUPOWANIA przypisano asocjacji  $\diamond Instance^{\emptyset}$ , reprezentującej instancje pojęć. Role ELEMENTU przypisano asocjacji  $\diamond Substitutability$ .

Wystąpienie wzorca posiada następujące własności możliwe:

- zapełnienie listy, ze względu na konieczność znajdowania się co najmniej jednego podstawienia w instancji,
- 1-ograniczona przynależność, ze względu na możliwość przypisania obiektu asocjacji  $\diamond Substitutability$  do co najwyżej jednego obiektu asocjacji  $\diamond Instance^{\emptyset}$ .

## 8.2.2. Słownik

Słownik został użyty w modelu ESNM<sup>SKB</sup> dwukrotnie, w obu przypadkach rolę WYLICZENIA BYTU pełni kolekcja  $\square CONCEPT^{\emptyset}$ . Kolekcja ta reprezentuje abstrakcyjne pojęcie, które posiada swoje konkretne specjalizacje w OCM<sup>SKB</sup>.

### 8.2.2.1. Pojęcie modyfikujące

$$\diamond Modifier [*] \xrightarrow{+Concept} [0..1] \square CONCEPT^{\emptyset} \quad (8.4)$$

W ramach tej instancji wzorca, wykorzystano pojęcie do opisanego asocjacji  $\diamond Modifier$ , stanowiącej ELEMENT OPISYWANY. Wykorzystano również negację własności możliwej binawigowalność słownika, aby uniemożliwić nawigowanie od pojęcia do modyfikatora.

### 8.2.2.2. Pojęcie opisujące instancję

$$\diamond Instance^{\emptyset} [*] \xrightarrow{+Concept} [0..1] \square CONCEPT^{\emptyset} \quad (8.5)$$

ELEMENT OPISYWANY dla tej instancji słownika to  $\diamond Instance^{\emptyset}$ . W odróżnieniu od poprzedniego przypadku, to wystąpienie wzorca nie posiada żadnych dodatkowych możliwych własności.

## 8.2.3. Autoreferencja

W konstrukcji modelu ESNM<sup>SKB</sup> wykorzystano dwa wystąpienia wzorca autoreferencji. Wystąpienia te, choć posiadają odmienne własności możliwe, oparte są

o koncepcję autoreferencyjnej roli, tj. takiej, dla której właściciel i cel roli są tą samą asocjacją.

### 8.2.3.1. Drzewo modyfikatorów

$$\diamond Modifier [1] \xrightarrow{+Modifier} [*] \diamond Modifier \quad (8.6)$$

Drzewo modyfikatorów zakłada, iż ELEMENTEM AUTOREFERENCYJNYM jest asocjacja  $\diamond Modifier$ . ZWIĄZKIEM AUTOREFERENCYJNYM z kolei jest jej rola: *Modifier*. Wykorzystanie wzorca pozwala określonym modyfikatorom przypisywać inne modyfikatory, w związku z tym, w naturalny sposób struktura taka ma postać drzewiastą. Należy zauważyć, iż w konstrukcji wzorca zanegowano własność możliwą binawigowalność roli autoreferencji w kierunku źródło  $\rightarrow$  cel, czyniąc autoreferencję nawigowaną jedynie w kierunku jej celu. Dodatkowo, określono 1-minimalność źródła autoreferencji i 1-maksymalność źródła autoreferencji w celu ograniczenia liczby węzłów nadrzędnych.

### 8.2.3.2. Drzewo podsieci

$$\diamond Net [0..1] \xrightarrow{+SubNet} [*] \diamond Net \quad (8.7)$$

Drzewo podsieci jest wystąpieniem wzorca autoreferencji, w ramach której rolę ELEMENTU AUTOREFERENCYJNEGO pełni asocjacja  $\diamond Net$ , a ZWIĄZKIEM AUTOREFERENCYJNYM jest jej rola *SubNet*. Mechanizm pozwala reprezentować sieci hierarchiczne, w ramach których definiowane są pewne podsieci. Własności możliwe wzorca, które wykorzystano to 1-maksymalność źródła autoreferencji oraz zależność istnienia elementu od autoreferencji dla źródła autoreferencji.

## 8.2.4. BACT

ESNM<sup>SKB</sup> zawiera trzy wystąpienia wzorca BACT. W sposób złożony funkcjonalnie ze względu na możliwość przechowywania danych oraz łączenia innych elementów zostały zamodelowane trzy byty: *operator*, *operand* oraz *podstawialność*. Należy wspomnieć, iż w modelu można odnaleźć jeszcze jeden fragment, który strukturalnie przypomina ten wzorzec (wyr. 8.8), lecz nim nie jest. Binawigowalna rola *Net* bardzo przypomina ten wzorzec, lecz ze względu na określenie kresu dolnego krotności roli po stronie związku jako 0, struktura ta nie spełnia koniecznych atomów semantycznych wzorca.

$$\diamond Net [1] \xrightarrow{+Net} [0..1] \square NET \quad (8.8)$$

### 8.2.4.1. Operator

$$\diamond Operator [1] \xrightarrow{+Operator} [1] \square OPERATOR \quad (8.9)$$

Kolekcja  $\square OPERATOR$  w tej instancji BACT jest ASPEKTEM DANYCH, natomiast ASPEKTEM ZWIĄZKU jest asocjacja  $\diamond Operator$ .

#### 8.2.4.2. Operand

$$\diamond Operand^{\emptyset} [1] \xrightarrow{+Operand} [1] \square OPERAND \quad (8.10)$$

W ramach tej instancji wzorca ASPEKT ZWIĄZKU to asocjacja  $\diamond Operand^{\emptyset}$ , a ASPEKT DANYCH – kolekcja  $\square OPERAND$ . Należy zauważyć, iż abstrakcyjność asocjacji nie wpływa na konieczne atomy semantyczne wzorca.

#### 8.2.4.3. Podstawialność

$$\diamond Substitutability [1] \xrightarrow{+Substitutability} [1] \square SUBSTITUTABILITY \quad (8.11)$$

W przypadku podstawialności, analogicznie jak w poprzednich wystąpieniach, kolekcja  $\square SUBSTITUTABILITY$  pełni rolę ASPEKTU DANYCH, a asocjacja  $\diamond Substitutability$  – ASPEKTU ZWIĄZKU.

#### 8.2.5. Abstrakcyjny cel roli

Wzorzec abstrakcyjnego celu roli został użyty w modelu jednokrotnie, w wersji, w której cel roli jest asocjacją. ARD został użyty do rozróżnienia strukturalnie odmiennych *operandów*, łączonych z *operatorem*.

##### 8.2.5.1. Abstrakcyjny Operand jako cel Operatora

$$\diamond Operator [*] \xrightarrow{+Operand} [1..*] \diamond Operand^{\emptyset} \quad (8.12)$$

$$\left\{ \begin{array}{l} \diamond OperandInstance \\ \diamond OperandOperator \\ \diamond OperandNet \\ \diamond OperandNode \end{array} \right\} \xrightarrow{rf^v} \diamond Operand^{\emptyset} \quad (8.13)$$

Abstrakcyjny cel roli w modelu ESNM<sup>SKB</sup> daje możliwość zamodelowania asocjacji  $\diamond Operand^{\emptyset}$  jako asocjacji abstrakcyjnej (pełniącej rolę ABSTRAKCYJNEGO UCZESTNIKA), która jest celem asocjacji  $\diamond Operator$  (pełniącej rolę ZWIĄZKU). Rolę KONKRETNEGO UCZESTNIKA pełnią asocjacje dziedziczące po  $\diamond Operand^{\emptyset}$ .

#### 8.2.6. Abstrakcyjny właściciel roli

W modelu został również wykorzystany wzorzec abstrakcyjnego właściciela roli. ARO został wykorzystany do zamodelowania instancji jako właściciela podstawień i pojęcia.

### 8.2.6.1. Instancja jako właściciel podstawień oraz pojęcia

$$\diamond Instance^{\emptyset} \left\{ \begin{array}{l} [*] \xrightarrow{+Concept} [0..1] \square CONCEPT \\ [0..1] \xrightarrow{+Substitutability} [*] \diamond Substitutability \end{array} \right\} \quad (8.14)$$

$$\left\{ \begin{array}{l} \diamond OperandInstance \\ \diamond Operator \end{array} \right\} \xrightarrow{rfv} \diamond Instance^{\emptyset} \quad (8.15)$$

Abstrakcyjny właściciel roli w modelu ESNM<sup>SKB</sup> posłużył do zamodelowania asocjacji  $\diamond Instance^{\emptyset}$  jako asocjacji abstrakcyjnej (pełniacej rolę ABSTRAKCYJNEGO ZWIĄZKU), która jest właścicielem dziedzicznych ról *Concept* oraz *Substitutability*. Ze względu na tryb dziedziczenia asocjacji, który uwzględnia dziedziczenie ról, role te zostały rozpropagowane do asocjacji pochodnych ( $\diamond OperandInstance$ ,  $\diamond Operator$ ), które pełnią rolę KONKRETNYCH ZWIĄZKÓW. Z kolei końcówki tych ról ( $\square Concept$ ,  $\diamond Substitutability$ ) pełnią rolę UCZESTNIKA.

### 8.2.7. Kaskada asocjacji

ESNM<sup>SKB</sup> zawiera również przykład zastosowania wzorca kaskady asocjacji. Kaskada posłużyła do zbudowania podstawowych konstrukcji wyrażanych w ramach modelu, tj. sieci semantycznych.

#### 8.2.7.1. Kaskadowa budowa sieci opartej o konstrukcję *operator-operand*

$$\diamond Net [*] \xrightarrow{+Operator} [1..*] \diamond Operator \quad (8.16)$$

$$\diamond Operator [*] \xrightarrow{+Operand} [1..*] \diamond Operand^{\emptyset} \quad (8.17)$$

Kaskada asocjacji obejmuje następujący łańcuch kolejnych asocjacji:  $\diamond Net$ ,  $\diamond Operator$ ,  $\diamond Operand^{\emptyset}$ , które stanowią trzon reprezentowanego modelu sieci semantycznych. Kres górny krotności określony jako *wiele* w każdej z ról realizujących kaskadę pozwala tworzyć wielooperatorowe zbiorowości  $n$ -arnych operatorów. Poziom kaskady w tym przypadku wynosi dwa. Asocjacja  $\diamond Net$  jest KORZENIEM KASKADY, natomiast asocjacje  $\diamond Operator$  i  $\diamond Operand^{\emptyset}$  stanowią związki kaskadowe kolejnych poziomów. UCZESTNIK KASKADY zależny jest od specjalizacji  $\diamond Operand^{\emptyset}$ , np. gdy obiekt asocjacji jest asocjacją  $\diamond OperandInstance$ , uczestnikiem kaskady jest  $\square CONCEPT$ .



### 8.3. Ekstrakcja semantyki modelu

Ekstrakcja semantyki modelu  $ESNM^{SKB}$  polegała na jego analizie i skutkowała opracowaniem zbioru skonkretyzowanych cząsteczek semantycznych metamodelu asocjacyjnego. Z uwagi na obszerność tego opracowania, wyniki ekstrakcji zostały zebrane i przedstawione w odrębnym raporcie [75]. Podczas ekstrakcji wyodrębniono 73 cząsteczki modelu asocjacyjnego, które symbolicznie przedstawiono w tabeli 8.1. Symbol  $N$  oznacza sumaryczną liczbę skonkretyzowanych cząsteczek semantycznych w modelu  $ESNM^{SKB}$  dla danej cząsteczki semantycznej metamodelu asocjacyjnego.

Tablica 8.1. Podsumowanie ekstrakcji semantyki modelu  $ESNM^{SKB}$

$N$	* <sub>AOM</sub> <sup>[kolekcja]</sup>	* <sub>AOM</sub> <sup>[atrybut]</sup>	* <sub>AOM</sub> <sup>[asocjacja]</sup>	* <sub>AOM</sub> <sup>[rola]</sup>	* <sub>AOM</sub> <sup>[opis asocjacji]</sup>	* <sub>AOM</sub> <sup>[opis roli]</sup>	* <sub>AOM</sub> <sup>[dziedziczenie]</sup>	* <sub>AOM</sub> <sup>[BACT]</sup>
	3	27	12	16	1	3	8	3

### 8.4. Translacje modelu

Translacje semantyki modelu  $ESNM^{SKB}$  przeprowadzono dla trzech metamodeli docelowych: EER, ORM, UML. Dla każdego z tych metamodeli dobór wzorców mapowania został przeprowadzony w analogiczny sposób. Zaproponowany w raporcie [75] opis cząsteczek semantycznych modelu pozwala na określenie jednoznacznego odwzorowania na gruncie opracowanej konwencji translacji na każdy z metamodeli. Podczas doboru uwzględniono wszystkie cząsteczki semantyczne  $ESNM^{SKB}$  wyrażonego w AOM, spełniając tym samym wymagania kompletności. Aplikacja zbioru wzorców mapowania pozwoliła na uzyskanie cząsteczek translacyjnych dla każdego z metamodeli docelowych. Modele stanowiące rekonstrukcje składniowe uwzględniające uzyskane cząsteczki semantyczne przedstawiono w kolejnych podrozdziałach.

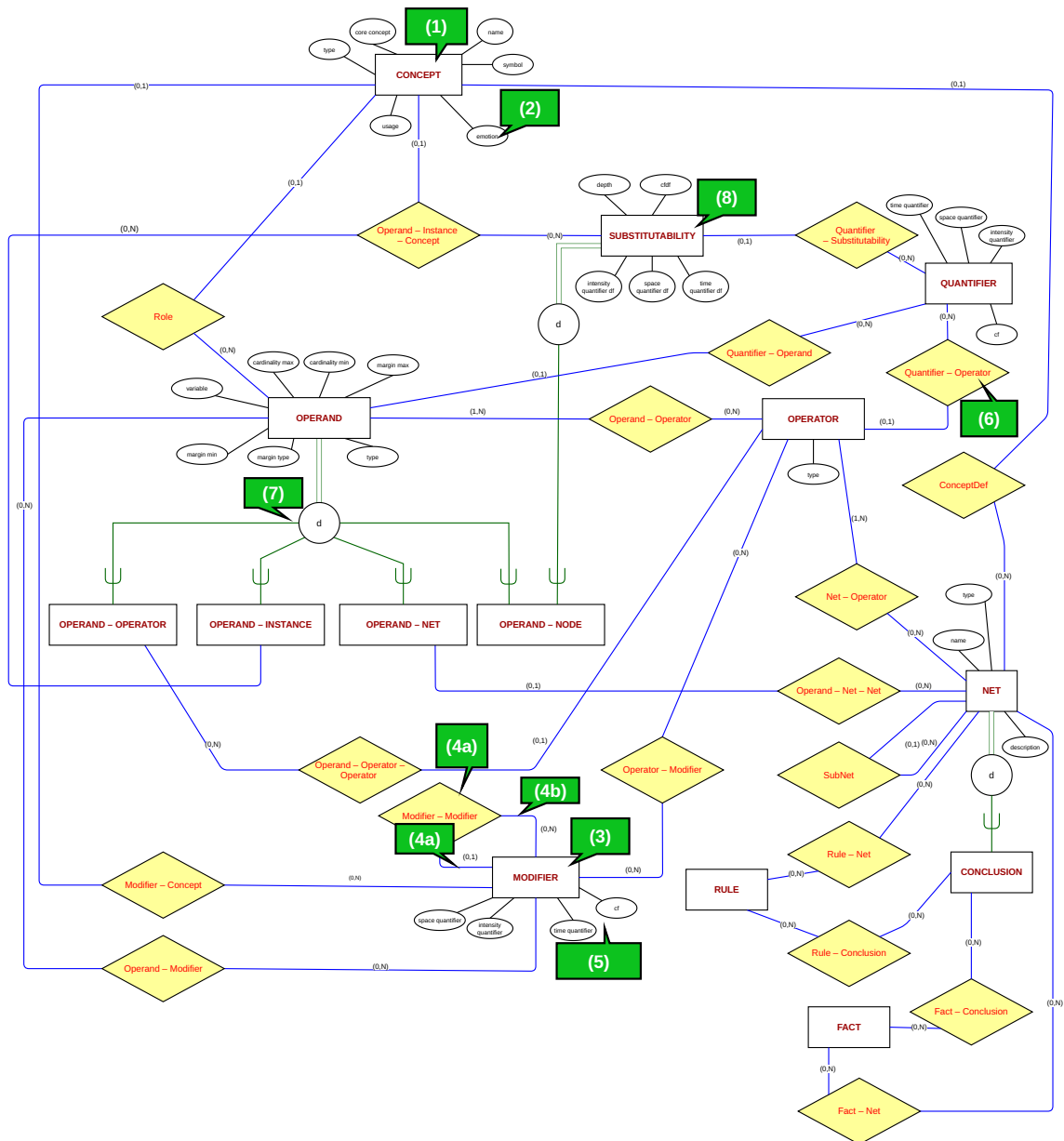
#### 8.4.1. Translacja semantyki do EER

Diagram na rysunku 8.2 przedstawia model  $ESNM^{SKB}$  wyrażony w ramach EER.

Kolejno przedstawione będą przykłady zastosowania wzorców mapowania dla poszczególnych rodzajów cząsteczek semantycznych metamodelu.

##### 8.4.1.1. Translacja cząsteczki semantycznej kolekcji

Podczas translacji semantyki modelu  $ESNM^{SKB}$  wzorzec mapowania zdefiniowany dla asocjacyjnej cząsteczki semantycznej kolekcji został wykorzystany trzy razy. Translacja została wykonana według wzorca mapowania opisanego w podroz-



Rysunek 8.2. Model ESNM<sup>SKB</sup> po translacji na EER

dziale 5.3.3.1. Przykładem zastosowania wzorca może być cząsteczka semantyczna reprezentująca kolekcję  $\square$ CONCEPT:

$$\ast_{\text{ESNM}^{\text{SKB}}}[\text{Concept}] \triangleleft_{\text{AOM}} \ast_{\text{AOM}}[\text{kolekcja}] \quad (8.18)$$

Dla tej cząsteczki aktywowany został atom semantyczny  $\ast_{\text{AOM}}^{[1]}$  oraz dezaktywowany atom semantyczny  $\ast_{\text{AOM}}^{[2]}$ . W wyniku zastosowania przekształcenia *ModelMapping*

określono następującą cząsteczkę translacyjną:

$$\begin{aligned} *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}_{\text{AOM}}}^{[\text{Concept}]} = & \left( \begin{array}{c} \emptyset, \\ \left\{ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{[\text{Concept}]} \triangleleft *_{\text{EER}}^{[\text{zbiór encji}]} \right\}, \\ \left\{ \odot_{\text{[esnm-eer:CONCEPT]}} \triangleleft \odot_{\text{[eer:EntitySet]}} \right\} \end{array} \right) \end{aligned} \quad (8.19)$$

Semantyka modelu docelowego reprezentowana jest przez cząsteczkę semantyczną, która reprezentuje semantykę pojęcia *CONCEPT*, odwzorowanego w modelu EER jako zbiór encji, którego reprezentację składniową na rysunku 8.2 oznaczono numerem **(1)**. Cząsteczka posiada dezaktywowany atom semantyczny  $*_{\text{EER}}^{[1]}$ , gdyż wymusił to zastosowany wzorec mapowania.

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{[\text{Concept}]} \triangleleft *_{\text{EER}}^{[\text{zbiór encji}]} \quad (8.20)$$

$$\begin{aligned} *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{[\text{Concept}]} = & \left( \begin{array}{c} *_{\text{EER}}^{\{\diamond 1\}}, \\ \emptyset, \\ \left\{ \odot_{\text{[esnm-eer:CONCEPT]}} \triangleleft \odot_{\text{[eer:EntitySet]}} \right\} \end{array} \right) \end{aligned} \quad (8.21)$$

#### 8.4.1.2. Translacja cząsteczki semantycznej atrybutu

Cząsteczka semantyczna atrybutu została zidentyfikowana w modelu źródłowym 27 razy. Podstawę jej odwzorowania stanowił wzorec mapowania opisany w podrozdziale 5.3.3.2.

Przykładem zastosowania wzorca może być cząsteczka semantyczna reprezentująca atrybut *emotion* kolekcji  $\square \text{CONCEPT}$ :

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{AOM}}^{[\text{Concept's emotion}]} \triangleleft *_{\text{AOM}}^{[\text{atrybut}]} \quad (8.22)$$

Atrybut ten jest dziedziczalny oraz posiadany przez kolekcję  $\square \text{CONCEPT}$ , w związku z tym posiada aktywowane odpowiednie atomy semantyczne. Nie jest również wirtualny, przez co atom semantyczny  $*_{\text{AOM}}^{[4]}$  jest dezaktywowany. W wyniku odwzorowania tej cząsteczki na EER określono cząsteczkę translacyjną:

$$\begin{aligned}
*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}_{\text{AOM}}}^{\text{[Concept's emotion]}} &= \left( \begin{array}{c} \emptyset, \\ \left\{ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{\text{[Concept's emotion]}} \triangleleft *_{\text{EER}}^{\text{[atrybut zbioru encji]}} \right\}, \\ \left\{ \begin{array}{l} \odot_{\text{[esnm-eer:Concept's emotion]}} \triangleleft \odot_{\text{[eer:Attribute]}} \\ \odot_{\text{[esnm-eer:CONCEPT]}} \triangleleft \odot_{\text{[eer:EntitySet]}} \\ \odot_{\text{[esnm-eer:Concept's emotion ValueSet]}} \triangleleft \odot_{\text{[eer:ValueSet]}} \end{array} \right\} \end{array} \right) \quad (8.23)
\end{aligned}$$

Semantyka modelu docelowego reprezentowana jest przez cząsteczkę semantyczną, która reprezentuje semantykę pojęcia reprezentowanego w modelu EER jako atrybut zbioru encji, którego reprezentację składniową na rysunku 8.2 oznaczono numerem **(2)**. Cząsteczka posiada dezaktywowane atomy semantyczne  $*_{\text{EER}}^{[4]}$ ,  $*_{\text{EER}}^{[5]}$ ,  $*_{\text{EER}}^{[7]}$ ,  $*_{\text{EER}}^{[11]}$ , gdyż wymusił to zastosowany wzorzec mapowania. Dezaktywacja atomu  $*_{\text{EER}}^{[3]}$ , wynika z równości kresu dolnego i górnego atrybutu, co z kolei zostało odwzorowane zgodnie z regułą wzorca mapowania.

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{\text{[Concept's emotion]}} \triangleleft *_{\text{EER}}^{\text{[atrybut zbioru encji]}} \quad (8.24)$$

$$\begin{aligned}
*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{\text{[Concept's emotion]}} &= \left( \begin{array}{c} *_{\text{EER}}^{\{\diamond 3, \diamond 4, \diamond 5, \diamond 7, \diamond 9, 10, \diamond 11, \diamond 12, \diamond 13\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot_{\text{[esnm-eer:Concept's emotion]}} \triangleleft \odot_{\text{[eer:Attribute]}} \\ \odot_{\text{[esnm-eer:CONCEPT]}} \triangleleft \odot_{\text{[eer:EntitySet]}} \\ \odot_{\text{[esnm-eer:Concept's emotion ValueSet]}} \triangleleft \odot_{\text{[eer:ValueSet]}} \\ \odot_{\underline{1}} \triangleleft \odot_{\text{[kres g\u00f3rny atrybutu]}} \\ \odot_{\underline{1}} \triangleleft \odot_{\text{[kres dolny atrybutu]}} \end{array} \right\} \end{array} \right) \quad (8.25)
\end{aligned}$$

#### 8.4.1.3. Translacja cząsteczki semantycznej asocjacji

Wzorzec mapowania oparty o cząsteczkę semantyczną asocjacji został wykorzystany 12 razy. Przykładem może być odwzorowanie asocjacji  $\diamond \text{Modifier}$ . Odwzorowanie semantyki wykonano według wzorca mapowania opisanego w podrozdziale 5.3.3.3. W ramach stosowania tego wzorca mapowania dla  $\text{ESNM}^{\text{SKB}}$  okazało się, iż każda instancja kategorii metamodelu asocjacyjnego *Assoc* stanowiąca podstawę syntaktyczną dla skonkretyzowanej cząsteczki semantycznej asocjacji, została odwzorowana do cząsteczki semantycznej, której podstawą był zbiór encji.

Przykładem może być cząsteczka semantyczna zbudowana wokół kategorii  $\diamond Modifier$ :

$$*_{ESNM^{SKB} \triangleleft AOM}^{[Modifier]} \triangleleft *_{AOM}^{[asocjacja]} \quad (8.26)$$

Cząsteczka ta ma zanegowane oba atomy semantyczne, które definiują jej charakterystykę:  $*_{AOM}^{[1]}$ ,  $*_{AOM}^{[2]}$ , tj. nie jest ani abstrakcyjna, ani uninawigowalna. Przekształcenie tej cząsteczki skutkuje powstaniem cząsteczki translacyjnej:

$$*_{ESNM^{SKB} \triangleleft EER_{AOM}}^{[Modifier]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{ESNM^{SKB} \triangleleft EER}^{[Modifier]} \triangleleft *_{EER}^{[zbiór encji]} \right\}, \\ \left\{ \odot_{[esnm-eer:MODIFIER]} \triangleleft \odot_{[eer:EntitySet]} \right\} \end{array} \right) \quad (8.27)$$

Cząsteczka ta opakowuje fragment semantyki docelowej modelu związków-encji, określony następująco:

$$*_{ESNM^{SKB} \triangleleft EER}^{[Modifier]} \triangleleft *_{EER}^{[zbiór encji]} \quad (8.28)$$

$$*_{ESNM^{SKB} \triangleleft EER}^{[Modifier]} = \left( \begin{array}{c} *_{EER}^{\{\diamond 1\}}, \\ \emptyset, \\ \left\{ \odot_{[esnm-eer:MODIFIER]} \triangleleft \odot_{[eer:EntitySet]} \right\} \end{array} \right) \quad (8.29)$$

Reprezentację syntaktyczną wynikowej cząsteczki semantycznej przedstawiono na diagramie symbolem **(3)**.

#### 8.4.1.4. Translacja cząsteczki semantycznej roli

Wzorzec mapowania zdefiniowany dla cząsteczki semantycznej roli wykorzystano 16 razy w ramach translacji semantyki modelu  $ESNM^{SKB}$ . Translację wykonano według wzorca mapowania opisanego w formie reguł w podrozdziale 5.3.3.4. Przykładem może być cząsteczka semantyczna oparta o rolę *Modifier* asocjacji  $\diamond Modifier$ :

$$*_{ESNM^{SKB} \triangleleft AOM}^{[Modifier's modifier]} \triangleleft *_{AOM}^{[rola]} \quad (8.30)$$

Jest to rola uninawigowalna, a co za tym idzie, skierowana do celu o krotnościach 1..1 po stronie związku oraz 0..\* po stronie elementu wiązanego. Dodatkowo, właściciel roli oraz jej cel są określone przez tę samą asocjację  $\diamond Modifier$ . Cząsteczka semantyczna  $*_{ESNM^{SKB} \triangleleft AOM}^{[Modifier's modifier]}$  posiada szereg atomów semantycznych aktywowanych oraz

dezaktywowanych oraz zdefiniowana została poniżej:

$$\begin{aligned}
*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{AOM}}^{\text{[Modifier's modifier]}} = & \left( \begin{array}{c} *_{\text{AOM}}^{\{\diamond 5, \diamond 6, \diamond 7, \diamond 8, \diamond 9, \diamond 10, \diamond 11, \diamond 12, 25, 26, 27, 28, \diamond 21, \diamond 23, \diamond 24\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot_{\text{[esnm-aom:Modifier's modifier (Role)]}} \triangleleft \odot_{\text{[aom:Role]}}, \\ \odot_{\text{[esnm-aom:Modifier (Assoc)]}} \triangleleft \odot_{\text{[aom:Assoc]}}, \\ \odot_{\text{[esnm-aom:Modifier (Assoc)]}} \triangleleft \odot_{\text{[końcówka roli]}}, \\ \odot_{\text{[1]}} \triangleleft \odot_{\text{[supremum krotności po str. związku]}}, \\ \odot_{\text{[1]}} \triangleleft \odot_{\text{[infimum krotności po str. związku]}}, \\ \odot_{\text{[*]}} \triangleleft \odot_{\text{[supremum krotności po str. el. związanego]}}, \\ \odot_{\text{[0]}} \triangleleft \odot_{\text{[infimum krotności po str. el. związanego]}} \end{array} \right\} \end{array} \right) \quad (8.31)
\end{aligned}$$

Przekształcenie tej cząsteczki skutkuje powstaniem cząsteczki translacyjnej:

$$\begin{aligned}
*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}_{\text{AOM}}}^{\text{[Modifier's modifier]}} = & \left( \begin{array}{c} \emptyset, \\ \left\{ \begin{array}{l} *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{\text{[Modifier - Modifier]}} \triangleleft *_{\text{EER}}^{\text{[zbiór związków]}}, \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{\text{[Modifier - Modifier r}_1\text{]}} \triangleleft *_{\text{EER}}^{\text{[rola związku]}}, \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{\text{[Modifier - Modifier r}_2\text{]}} \triangleleft *_{\text{EER}}^{\text{[rola związku]}} \end{array} \right\}, \\ \left\{ \begin{array}{l} \odot_{\text{[esnm-eer:Modifier - Modifier]}} \triangleleft \odot_{\text{[eer:RelationshipSet]}}, \\ \odot_{\text{[esnm-eer:Modifier - Modifier r}_1\text{]}} \triangleleft \left[ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{\text{[Modifier - Modifier r}_1\text{]}} \right] \odot_{\text{[eer:Role]}}, \\ \odot_{\text{[esnm-eer:Modifier - Modifier r}_2\text{]}} \triangleleft \left[ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{\text{[Modifier - Modifier r}_2\text{]}} \right] \odot_{\text{[eer:Role]}}, \\ \odot_{\text{[esnm-eer:MODIFIER]}} \triangleleft \odot_{\text{[eer:EntitySet]}}, \\ \odot_{\text{[eer:partial]}} \triangleleft \odot_{\text{[eer:ParticipationKind]}}, \\ \odot_{\text{[eer:one]}} \triangleleft \left[ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{\text{[Modifier - Modifier r}_1\text{]}} \right] \odot_{\text{[eer:CardinalityKind]}}, \\ \odot_{\text{[eer:many]}} \triangleleft \left[ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{\text{[Modifier - Modifier r}_2\text{]}} \right] \odot_{\text{[eer:CardinalityKind]}} \end{array} \right\} \end{array} \right) \quad (8.32)
\end{aligned}$$

Cząsteczka ta opakowuje fragment semantyki docelowej modelu związków-encji, określonej jako trzy cząsteczki modelu docelowego wskazane na rysunku 8.2 jako kolejno (4a), (4b), (4c):

(4a):

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{\text{[Modifier - Modifier]}} \triangleleft *_{\text{EER}}^{\text{[zbiór związków]}} \quad (8.33)$$

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{[\text{Modifier} - \text{Modifier}]} = \left( \begin{array}{c} *_{\text{EER}}^{\{\diamond 1\}} \\ \emptyset, \\ \left\{ \odot_{\text{esnm-eer:Modifier} - \text{Modifier}} \triangleleft \odot_{\text{eer:RelationshipSet}} \right\} \end{array} \right) \quad (8.34)$$

(4b):

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{[\text{Modifier} - \text{Modifier } r_1]} \triangleleft *_{\text{EER}}^{[\text{rola związku}]} \quad (8.35)$$

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{[\text{Modifier} - \text{Modifier } r_1]} = \left( \begin{array}{c} *_{\text{EER}}^{\{\diamond 8,15,16,17,\diamond 18\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot_{\text{esnm-eer:Modifier} - \text{Modifier}} \triangleleft \odot_{\text{eer:RelationshipSet}} \\ \odot_{\text{esnm-eer:Modifier} - \text{Modifier } r_1} \triangleleft \odot_{\text{eer:Role}}, \\ \odot_{\text{eer:partial}} \triangleleft \odot_{\text{eer:ParticipationKind}} \\ \odot_{\text{eer:one}} \triangleleft \odot_{\text{eer:CardinalityKind}} \\ \odot_{\text{esnm-eer:MODIFIER}} \triangleleft \odot_{\text{eer:EntitySet}} \end{array} \right\} \end{array} \right) \quad (8.36)$$

(4c):

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{[\text{Modifier} - \text{Modifier } r_2]} \triangleleft *_{\text{EER}}^{[\text{rola związku}]} \quad (8.37)$$

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{EER}}^{[\text{Modifier} - \text{Modifier } r_2]} = \left( \begin{array}{c} *_{\text{EER}}^{\{\diamond 8,15,16,17,\diamond 18\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot_{\text{esnm-eer:Modifier} - \text{Modifier}} \triangleleft \odot_{\text{eer:RelationshipSet}} \\ \odot_{\text{esnm-eer:Modifier} - \text{Modifier } r_2} \triangleleft \odot_{\text{eer:Role}}, \\ \odot_{\text{eer:partial}} \triangleleft \odot_{\text{eer:ParticipationKind}} \\ \odot_{\text{eer:many}} \triangleleft \odot_{\text{eer:CardinalityKind}} \\ \odot_{\text{esnm-eer:MODIFIER}} \triangleleft \odot_{\text{eer:EntitySet}} \end{array} \right\} \end{array} \right) \quad (8.38)$$

#### 8.4.1.5. Translacja cząsteczki semantycznej opisu asocjacji

W modelu  $\text{ESNM}^{\text{SKB}}$  zidentyfikowano cząsteczkę semantyczną opisu asocjacji tylko raz. Opis asocjacji dotyczy konstrukcji użytej do przypisania asocjacji  $\diamond \text{Modifier}$

kolekcji opisującej  $\square QUANTIFIER$ .

$$*_{ESNM^{SKB} \triangleleft AOM}^{[Description\ of\ Modifier]} \triangleleft *_{AOM}^{[opis\ asocjacji]} \quad (8.39)$$

Odwzorowanie cząsteczki przeprowadzono według wzorca mapowania określonego w podrozdziale 5.3.3.5. Zastosowanie wzorca polega na przeniesieniu atrybutów elementu opisującego do elementu opisywanego. W tym przypadku, asocjacja  $\diamond Modifier$  została odwzorowana jako zbiór encji, więc atrybuty kolekcji opisującej stały się atrybutami zbioru encji. Skonkretyzowana cząsteczka translacyjna przyjmuje następującą postać:

$$*_{ESNM^{SKB} \triangleleft EER_{AOM}}^{[Description\ of\ Modifier]} = \left( \begin{array}{l} \emptyset, \\ \left\{ \begin{array}{l} *_{ESNM^{SKB} \triangleleft EER}^{[Modifier's\ space\ quantifier]} \triangleleft *_{EER}^{[atrybut\ zbioru\ encji]}, \\ *_{ESNM^{SKB} \triangleleft EER}^{[Modifier's\ intensity\ quantifier]} \triangleleft *_{EER}^{[atrybut\ zbioru\ encji]}, \\ *_{ESNM^{SKB} \triangleleft EER}^{[Modifier's\ time\ quantifier]} \triangleleft *_{EER}^{[atrybut\ zbioru\ encji]}, \\ *_{ESNM^{SKB} \triangleleft EER}^{[Modifier's\ cf]} \triangleleft *_{EER}^{[atrybut\ zbioru\ encji]}, \end{array} \right\}, \\ \left\{ \begin{array}{l} \odot_{[esnm-eer:Modifier's\ space\ quantifier]} \triangleleft \odot_{[eer:Attribute]} \\ \odot_{[esnm-eer:Modifier's\ space\ ValueSet]} \triangleleft \odot_{[eer:ValueSet]} \\ \odot_{[esnm-eer:Modifier's\ intensity\ quantifier]} \triangleleft \odot_{[eer:Attribute]} \\ \odot_{[esnm-eer:Modifier's\ intensity\ ValueSet]} \triangleleft \odot_{[eer:ValueSet]} \\ \odot_{[esnm-eer:Modifier's\ time\ quantifier]} \triangleleft \odot_{[eer:Attribute]} \\ \odot_{[esnm-eer:Modifier's\ time\ ValueSet]} \triangleleft \odot_{[eer:ValueSet]} \\ \odot_{[esnm-eer:Modifier's\ cf]} \triangleleft \odot_{[eer:Attribute]} \\ \odot_{[esnm-eer:Modifier's\ cf\ ValueSet]} \triangleleft \odot_{[eer:ValueSet]} \\ \odot_{[esnm-eer:MODIFIER]} \triangleleft \odot_{[eer:EntitySet]} \\ \odot_{[1]} \triangleleft \odot_{[kres\ g\u00f3rny\ atrybutu]} \\ \odot_{[1]} \triangleleft \odot_{[kres\ dolny\ atrybutu]} \end{array} \right\} \end{array} \right) \quad (8.40)$$

Cząsteczki modelu wynikowego będą reprezentować atrybuty zbioru encji, a zatem będą miały postać analogiczną do wyrażenia 8.25. Reprezentację gramatyczną powstałej struktury oznaczono na diagramie symbolem (5).

#### 8.4.1.6. Translacja cząsteczki semantycznej opisu roli

Jednym spośród trzech przykładów cząsteczki semantycznej opisu roli w  $ESNM^{SKB}$  jest opisanie roli *Operator* asocjacji  $\diamond Operator$  przy pomocy kolekcji  $\square QUANTIFIER$ . Przyjęty wzorec mapowania opisany w podrozdziale 5.3.3.6



traktuje opis roli jako dodatkową rolę asocjacji, której właścicielem jest opisywana rola. Dlatego odwzorowanie cząsteczki semantycznej

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \triangleleft \text{AOM} \end{matrix} \begin{matrix} [\text{Operator's operand}] \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{AOM} \end{matrix} \begin{matrix} [\text{rola}] \end{matrix} \quad (8.41)$$

przeprowadzono w taki właśnie sposób, aby translacyjna cząsteczka przyjęła następującą postać:

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \triangleleft \text{EER}_{\text{AOM}} \end{matrix} \begin{matrix} [\text{Description of Operator}] \\ = \end{matrix} \left( \begin{array}{c} \emptyset, \\ \left\{ \begin{array}{l} * \\ \text{ESNM}^{\text{SKB}} \triangleleft \text{EER} \end{array} \begin{matrix} [\text{Description of Operator}] \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{EER} \end{matrix} \begin{matrix} [\text{zbiór związków}] \\ \end{matrix}, \\ * \\ \text{ESNM}^{\text{SKB}} \triangleleft \text{EER} \end{array} \begin{matrix} [\text{Description of Operator's } r_1] \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{EER} \end{matrix} \begin{matrix} [\text{rola związku}] \\ \end{matrix}, \\ * \\ \text{ESNM}^{\text{SKB}} \triangleleft \text{EER} \end{array} \begin{matrix} [\text{Description of Operator's } r_2] \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{EER} \end{matrix} \begin{matrix} [\text{rola związku}] \\ \end{matrix} \right), \\ \left\{ \begin{array}{l} \odot \begin{matrix} \underline{\text{esnm-eer:Quantifier – Operator}} \\ \triangleleft \end{matrix} \odot \begin{matrix} \underline{\text{eer:RelationshipSet}} \\ \end{matrix} \\ \odot \begin{matrix} \underline{\text{esnm-eer:OPERATOR}} \\ \triangleleft \end{matrix} \odot \begin{matrix} \underline{\text{eer:EntitySet}} \\ \end{matrix} \\ \odot \begin{matrix} \underline{\text{esnm-eer:QUANTIFIER}} \\ \triangleleft \end{matrix} \odot \begin{matrix} \underline{\text{eer:EntitySet}} \\ \end{matrix} \\ \odot \begin{matrix} \underline{\text{esnm-eer:Description of Operator's } r_1} \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \triangleleft \text{EER} \end{matrix} \begin{matrix} [\text{Description of Operator's } r_1] \\ \triangleleft \end{matrix} \odot \begin{matrix} \underline{\text{eer:Role}} \\ \end{matrix} \\ \odot \begin{matrix} \underline{\text{eer:one}} \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \triangleleft \text{EER} \end{matrix} \begin{matrix} [\text{Description of Operator's } r_1] \\ \triangleleft \end{matrix} \odot \begin{matrix} \underline{\text{eer:CardinalityKind}} \\ \end{matrix} \\ \odot \begin{matrix} \underline{\text{esnm-eer:Description of Operator's } r_2} \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \triangleleft \text{EER} \end{matrix} \begin{matrix} [\text{Description of Operator's } r_2] \\ \triangleleft \end{matrix} \odot \begin{matrix} \underline{\text{eer:Role}} \\ \end{matrix} \\ \odot \begin{matrix} \underline{\text{eer:many}} \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \triangleleft \text{EER} \end{matrix} \begin{matrix} [\text{Description of Operator's } r_2] \\ \triangleleft \end{matrix} \odot \begin{matrix} \underline{\text{eer:CardinalityKind}} \\ \end{matrix} \\ \odot \begin{matrix} \underline{\text{eer:partial}} \\ \triangleleft \end{matrix} \odot \begin{matrix} \underline{\text{eer:ParticipationKind}} \\ \end{matrix} \end{array} \right) \quad (8.42)$$

Cząsteczki modelu wynikowego reprezentują binarny zbiór związków, a zatem mają postać podobną do wyrażeń 8.34, 8.36, 8.38. Reprezentację gramatyczną powstałej struktury oznaczono na diagramie symbolem **(6)**.

#### 8.4.1.7. Translacja cząsteczki semantycznej dziedziczenia

Cząsteczka semantyczna opisująca dziedziczenie w modelu będącym przedmiotem studium przypadku wystąpiła osiem razy. Jednym z przykładów jej wystąpienia może być związek generalizacji-specjalizacji pomiędzy asocjacją  $\diamond \text{Operand}$  a pochodnymi asocjacjami, np.  $\diamond \text{OperandOperator}$ . W ramach odwzorowywania semantyki wykorzystany został wzorec przedstawiiony w podrozdziale 5.3.3.7. Odwzorowywana jest cząsteczka semantyczna:

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \triangleleft \text{AOM} \end{matrix} \begin{matrix} [\text{OperandOperator's inheritance from Operand}] \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{AOM} \end{matrix} \begin{matrix} [\text{dziedziczenie}] \end{matrix} \quad (8.43)$$

Ponieważ pojęcia reprezentujące  $\diamond Operand$  i  $\diamond OperandOperator$  zostały przeniesione do modelu docelowego jako zbiory encji, zgodnie z wzorcem możliwe jest zastosowanie związku generalizacji będącego częścią semantyki EER. W efekcie, otrzymano następującą cząsteczkę translacyjną:

$$\begin{aligned}
 & *_{ESNM^{SKB} \triangleleft EER_{AOM}} [OperandOperator's inheritance from Operand] = \\
 & \left( \begin{array}{c} \emptyset, \\ \left\{ *_{ESNM^{SKB} \triangleleft EER} [OperandOperator's gen/spec from Operand] \triangleleft *_{EER} [generalizacja-specjalizacja] \right\}, \\ \left\{ \begin{array}{l} \odot [esn\text{-}eer:OperandOperator \text{ gen/spec from Operand}] \triangleleft \odot [eer:Generalization/Specialization] \\ \odot [esn\text{-}eer:Operand] \triangleleft \odot [generalizacja] \\ \odot [esn\text{-}eer:OperandOperator] \triangleleft \odot [specjalizacja] \\ \odot [eer:disjoint] \triangleleft \odot [eer:DisjointnessKind], \\ \odot [eer:total] \triangleleft \odot [eer:ParticipationKind] \end{array} \right\} \end{array} \right) \quad (8.44)
 \end{aligned}$$

Cząsteczka translacyjna definiuje postać następującej skonkretyzowanej cząsteczki modelu docelowego:

$$*_{ESNM^{SKB} \triangleleft EER} [OperandOperator's gen/spec from Operand] \triangleleft *_{EER} [generalizacja-specjalizacja] \quad (8.45)$$

$$\begin{aligned}
 & *_{ESNM^{SKB} \triangleleft EER} [OperandOperator's gen/spec from Operand] = \\
 & \left( \begin{array}{c} *_{EER}^{\{19,20,21,22\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot [esn\text{-}eer:OperandOperator \text{ gen/spec from Operand}] \triangleleft \odot [eer:Generalization/Specialization] \\ \odot [esn\text{-}eer:Operand] \triangleleft \odot [generalizacja] \\ \odot [esn\text{-}eer:OperandOperator] \triangleleft \odot [specjalizacja] \\ \odot [eer:disjoint] \triangleleft \odot [eer:DisjointnessKind], \\ \odot [eer:total] \triangleleft \odot [eer:ParticipationKind] \end{array} \right\} \end{array} \right) \quad (8.46)
 \end{aligned}$$

W składni konkretnej przedstawionej na diagramie 8.2 wynikowe struktury oznaczono symbolem (7).

#### 8.4.1.8. Translacja cząsteczki semantycznej BACT

W modelu  $ESNM^{SKB}$  cząsteczka semantyczna opisująca wzorzec modelowania BACT wystąpiła trzy razy. Przykładem może być instancja BACT dotycząca podstawialności (*Substitutability*) opisana wyrażeniem 8.11. Dla tej struktury zdefiniowano następującą cząsteczkę semantyczną:

$$\ast_{ESNM^{SKB} \triangleleft AOM}^{[Substitutability]} \triangleleft \ast_{AOM}^{[BACT]} \quad (8.47)$$

Została ona odwzorowana zgodnie z wzorcem mapowania opisanym w podrozdziale 5.3.3.8. W wyniku otrzymano następującą cząsteczkę translacyjną:

$$\ast_{ESNM^{SKB} \triangleleft EER_{AOM}}^{[Substitutability]} = \left( \begin{array}{c} \emptyset, \\ \left\{ \ast_{ESNM^{SKB} \triangleleft EER}^{[Substitutability]} \triangleleft \ast_{EER}^{[zbiór encji]} \right\}, \\ \left\{ \odot_{[esnm-eer:SUBSTITUTABILITY]} \triangleleft \odot_{[eer:EntitySet]} \right\} \end{array} \right) \quad (8.48)$$

Cząsteczka ta z kolei jest nośnikiem semantyki skonkretyzowanej cząsteczki modelu docelowego:

$$\ast_{ESNM^{SKB} \triangleleft EER}^{[Substitutability]} \triangleleft \ast_{EER}^{[zbiór encji]} \quad (8.49)$$

$$\ast_{ESNM^{SKB} \triangleleft EER}^{[Substitutability]} = \left( \begin{array}{c} \ast_{EER}^{\{\diamond 1\}}, \\ \emptyset, \\ \left\{ \odot_{[esnm-eer:SUBSTITUTABILITY]} \triangleleft \odot_{[eer:EntitySet]} \right\} \end{array} \right) \quad (8.50)$$

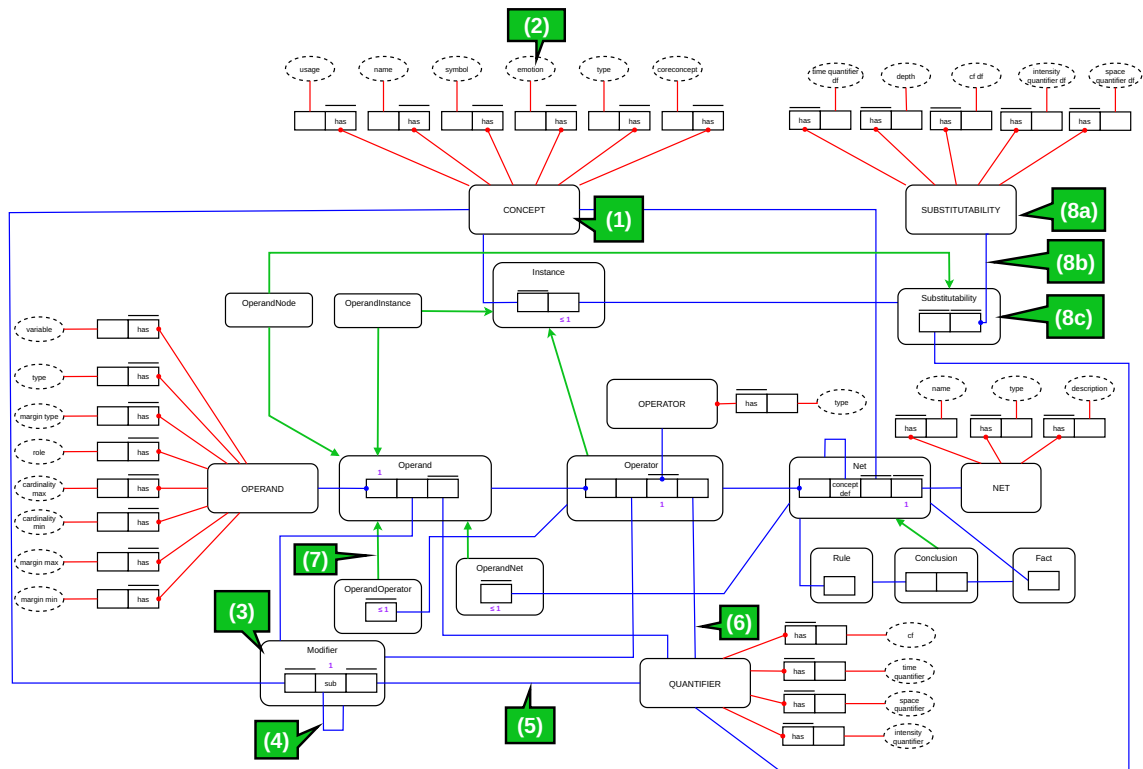
Wynikowe struktury modelu odpowiadające za realizację uzyskanej semantyki przedstawiono na diagramie, oznaczając je symbolem (8).

#### 8.4.2. Translacja semantyki do ORM

Diagram na rysunku 8.3 przedstawia model  $ESNM^{SKB}$  wyrażony jako model ORM. Przykłady zastosowania wzorców mapowania dla poszczególnych typów cząsteczek semantycznych AOM przedstawiono w kolejnych podrozdziałach.

##### 8.4.2.1. Translacja cząsteczki semantycznej kolekcji

Cząsteczka semantyczna modelu docelowego, która powstała poprzez odwzorowanie  $\ast_{ESNM^{SKB} \triangleleft AOM}^{[Concept]}$  została przedstawiona poniżej. Translację wykonano zgodnie z opisem wzorca w rozdziale 5.4.3.1:



Rysunek 8.3. Model ESNM<sup>SKB</sup> po translacji na ORM

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}}[\text{Concept}] \triangleleft *_{\text{ORM}}[\text{obiek-t-byt}] \quad (8.51)$$

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}}[\text{Concept}] = \left( \begin{array}{c} *_{\text{ORM}}\{\diamond 2, \diamond 3, \}, \\ \emptyset, \\ \left\{ \odot_{\text{[esnm-orm:Concept]}} \triangleleft \odot_{\text{[orm:Entity Type]}} \right\} \end{array} \right) \quad (8.52)$$

Reprezentację gramatyczną powstałej struktury oznaczono na diagramie symbolem (1).

#### 8.4.2.2. Translacja części semantycznej atrybutu

Translacja części semantycznych atrybutu odbyła się zgodnie z wzorcem przedstawionym w rozdziale 5.4.3.2. Biorąc pod uwagę fakt, iż ORM jest metamodeliem beztrybutowym, w wyniku powstała translacyjna część semantyczna składająca się z konkretyzacji:

- części realizującej byt posiadający atrybut:  $*_{\text{ORM}}^{\text{[obiek-t-byt]}}$ ,
- części realizującej wartość reprezentowaną przez atrybut:  $*_{\text{ORM}}^{\text{[obiek-t-wartość]}}$ ,
- części realizującej fakt posiadania atrybutu:  $*_{\text{ORM}}^{\text{[fakt]}}$ ,

— dwóch cząsteczek realizujących role obiektów:  $*_{\text{ORM}}^{\text{rola}}$ .

W wyniku zastosowania wzorca powstaje cząsteczka translacyjna składająca się z konkretyzacji wymienionych powyżej. Przykładową cząsteczką translacyjną dla atrybutu *emotion* kolekcji  $\square \text{CONCEPT}$  jest poniższa cząsteczka:

$$\begin{aligned}
 & *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}_{\text{AOM}}}^{\text{[Concept's emotion]}} = \\
 & \left( \begin{array}{c} \emptyset, \\ \left\{ \begin{array}{l} *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}}^{\text{[Concept]}} \triangleleft *_{\text{ORM}}^{\text{[obiekt-byt]}}, \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}}^{\text{[Concept's emotion]}} \triangleleft *_{\text{ORM}}^{\text{[obiekt-wartość]}}, \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}}^{\text{[having emotion by concept's concept]}} \triangleleft *_{\text{ORM}}^{\text{[rola]}}, \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}}^{\text{[having emotion by concept's emotion]}} \triangleleft *_{\text{ORM}}^{\text{[rola]}}, \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}}^{\text{[having emotion by concept]}} \triangleleft *_{\text{ORM}}^{\text{[fakt]}} \end{array} \right\}, \\ \left\{ \begin{array}{l} \odot_{\text{[esnm-orm:Concept]}} \triangleleft \odot_{\text{[byt]}} \\ \odot_{\text{[esnm-orm:emotion]}} \triangleleft \odot_{\text{[atrybut]}} \\ \odot_{\text{[esnm-orm:having emotion by concept's concept]}} \triangleleft \odot_{\text{[rola bytu]}} \\ \odot_{\text{[esnm-orm:having emotion by concept's emotion]}} \triangleleft \odot_{\text{[rola atrybutu]}} \\ \odot_{\text{[esnm-orm:having emotion by concept]}} \triangleleft \odot_{\text{[orm:FactType]}} \end{array} \right\} \end{array} \right) \quad (8.53)
 \end{aligned}$$

Reprezentację gramatyczną powstałej struktury oznaczono na diagramie symbolem (2).

### 8.4.2.3. Translacja cząsteczki semantycznej asocjacji

Translacja zidentyfikowanych cząsteczek semantycznych asocjacji na ORM została przeprowadzona zgodnie z wzorcem mapowania opisanym w rozdziale 5.4.3.3. Pojęciu asocjacji odpowiada typ faktu, względnie reifikowanego, w przypadku, gdy jego asocjacyjna reprezentacja stanowi zakończenie innych ról. Przykładową cząsteczką translacyjną przedstawiono poniżej:

$$\begin{aligned}
 & *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}_{\text{AOM}}}^{\text{[Modifier]}} = \\
 & \left( \begin{array}{c} \emptyset, \\ \left\{ \begin{array}{l} *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}}^{\text{[Modifier (fact)]}} \triangleleft *_{\text{ORM}}^{\text{[fakt]}}, \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}}^{\text{[Modifier]}} \triangleleft *_{\text{ORM}}^{\text{[reifikacja]}}, \end{array} \right\}, \\ \left\{ \begin{array}{l} \odot_{\text{[esnm-orm:Modifier (fact)]}} \triangleleft \odot_{\text{[orm:FactType]}} \\ \odot_{\text{[esnm-orm:Modifier (entity)]}} \triangleleft \odot_{\text{[orm:EntityType]}} \end{array} \right\} \end{array} \right) \quad (8.54)
 \end{aligned}$$

Reprezentację gramatyczną powstałej struktury oznaczono na diagramie symbolem (3).

#### 8.4.2.4. Translacja cząsteczki semantycznej roli

Wzorzec mapowania cząsteczki semantycznej opisany w rozdziale 5.4.3.4 stanowił podstawę przeprowadzenia odwzorowania ról. Cząsteczki semantyczne roli odwzorowywane są na cząsteczki translacyjne składające się z roli w rozumieniu ORM.

Przykładem roli w modelu ESNM<sup>SKB</sup> jest rola *Modifier* asocjacji  $\diamond$ *Modifier*. Jej odwzorowanie w ORM realizuje poniższa cząsteczka semantyczna:

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \triangleleft \text{ORM} \end{matrix} \begin{matrix} [\text{Modifier's modifier}] \\ \text{ORM} \end{matrix} \triangleleft \begin{matrix} * \\ \text{ORM} \end{matrix} \begin{matrix} [\text{rola}] \\ \text{ORM} \end{matrix} \quad (8.55)$$

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \triangleleft \text{ORM} \end{matrix} \begin{matrix} [\text{Modifier's modifier}] \\ \text{ORM} \end{matrix} = \left( \begin{matrix} * \\ \text{ORM} \end{matrix} \begin{matrix} \{\diamond 4, \diamond 5, \diamond 8, 9, \diamond 14, \diamond 15, \diamond 16, \diamond 17, \diamond 18, \diamond 19, \diamond 20, \diamond 21, \diamond 22, \} \\ \text{ORM} \end{matrix}, \right. \\ \left. \begin{matrix} \emptyset, \\ \left\{ \begin{matrix} \odot \underline{[\text{esnm-orm:Modifier's modifier}]} \triangleleft \odot \underline{[\text{orm:Role}]}, \\ \odot \underline{[\text{esnm-orm:Modifier (fact)}]} \triangleleft \odot \underline{[\text{związek}]}, \\ \odot \underline{[\text{esnm-orm:Modifier (entity)}]} \triangleleft \odot \underline{[\text{orm:ObjectType}]} \end{matrix} \right\} \end{matrix} \right) \quad (8.56)$$

Reprezentację gramatyczną powstałej struktury oznaczono na diagramie symbolem (4).

#### 8.4.2.5. Translacja cząsteczki semantycznej opisu asocjacji

Opis asocjacji  $\diamond$ *Modifier* został zrealizowany poprzez zwiększenie arności odpowiadającego mu typu faktu. W wyniku pojawiła się dodatkowa rola zakończona w obiekcie, reprezentującym kolekcję opisującą. Reprezentuje ten fakt poniższa cząsteczka translacyjna zgodna z wzorcem mapowania:

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \triangleleft \text{ORM} \end{matrix} \begin{matrix} [\text{Description of modifier}] \\ \text{ORM} \end{matrix} \triangleleft \begin{matrix} * \\ \text{ORM} \end{matrix} \begin{matrix} [\text{rola}] \\ \text{ORM} \end{matrix} \quad (8.57)$$

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \triangleleft \text{ORM} \end{matrix} \begin{matrix} [\text{Description of modifier}] \\ \text{ORM} \end{matrix} = \left( \begin{matrix} * \\ \text{ORM} \end{matrix} \begin{matrix} \{\diamond 4, \diamond 5, \diamond 8, 9, \diamond 14, \diamond 15, \diamond 16, \diamond 17, \diamond 18, \diamond 19, \diamond 20, \diamond 21, \diamond 22, \} \\ \text{ORM} \end{matrix}, \right. \\ \left. \begin{matrix} \emptyset, \\ \left\{ \begin{matrix} \odot \underline{[\text{esnm-orm:Description of modifier}]} \triangleleft \odot \underline{[\text{orm:Role}]}, \\ \odot \underline{[\text{esnm-orm:Modifier (fact)}]} \triangleleft \odot \underline{[\text{związek}]}, \\ \odot \underline{[\text{esnm-orm:Quantifier}]} \triangleleft \odot \underline{[\text{orm:ObjectType}]} \end{matrix} \right\} \end{matrix} \right) \quad (8.58)$$

Reprezentację gramatyczną powstałej struktury oznaczono na diagramie symbolem (5).

#### 8.4.2.6. Translacja cząsteczki semantycznej opisu roli

Opisy roli, analogicznie do opisu asocjacji zostały odwzorowane w postaci dodatkowych ról w faktach odwzorowujących asocjacje zawierające opisywane role. Przykładem jest tutaj opis roli *Operator* asocjacji  $\diamond Operator$ , który został zrealizowany przy pomocy następującej cząsteczki semantycznej:

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \end{matrix} \begin{matrix} [\text{Description of operator}] \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{ORM} \end{matrix} \begin{matrix} [\text{rola}] \\ \end{matrix} \quad (8.59)$$

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \end{matrix} \begin{matrix} [\text{Description of operator}] \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{ORM} \end{matrix} = \left( \begin{matrix} * \\ \text{ORM} \end{matrix} \begin{matrix} \{\diamond 4, \diamond 5, \diamond 8, 9, \diamond 14, \diamond 15, \diamond 16, \diamond 17, \diamond 18, \diamond 19, \diamond 20, \diamond 21, \diamond 22, \} \\ \end{matrix} , \right. \\ \left. \begin{matrix} \emptyset, \\ \left\{ \begin{matrix} \odot \underline{[esnm-orm:Description of operator]} \triangleleft \odot \underline{[orm:Role]}, \\ \odot \underline{[esnm-orm:Modifier (fact)]} \triangleleft \odot \underline{[związek]}, \\ \odot \underline{[esnm-orm:Operator (entity)]} \triangleleft \odot \underline{[orm:ObjectType]} \end{matrix} \right\} \end{matrix} \right) \quad (8.60)$$

Reprezentację gramatyczną powstałej struktury oznaczono na diagramie symbolem (6).

#### 8.4.2.7. Translacja cząsteczki semantycznej dziedziczenia

Cząsteczka semantyczna dziedziczenia została odwzorowana z wykorzystaniem związku podtypowania (ang. *subtyping*) istniejącego w ORM. W przypadku dziedziczenia asocjacji, przyjęto, iż związek podtypowania tworzony będzie dla obiektów reifikujących fakty. W przypadku dziedziczenia asocjacji  $\diamond OperandOperator$  po  $\diamond Operand$  powstała następująca cząsteczka semantyczna:

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \end{matrix} \begin{matrix} [\text{OperandOperator's subtyping from Operand}] \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{ORM} \end{matrix} \begin{matrix} [\text{podtyp}] \\ \end{matrix} \quad (8.61)$$

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \end{matrix} \begin{matrix} [\text{OperandOperator's subtyping from Operand}] \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{ORM} \end{matrix} = \left( \begin{matrix} * \\ \text{ORM} \end{matrix} \begin{matrix} \{11,12\} \\ \end{matrix} , \right. \\ \left. \begin{matrix} \emptyset, \\ \left\{ \begin{matrix} \odot \underline{[esnm-orm:Description of operator]} \triangleleft \odot \underline{[orm:Role]} \\ \odot \underline{[esnm-orm:Modifier (fact)]} \triangleleft \odot \underline{[związek]}, \\ \odot \underline{[esnm-orm:Operator (entity)]} \triangleleft \odot \underline{[orm:ObjectType]} \end{matrix} \right\} \end{matrix} \right) \quad (8.62)$$

Reprezentację gramatyczną powstałej struktury oznaczono na diagramie symbolem (7).

### 8.4.2.8. Translacja cząsteczki semantycznej BACT

Reprezentacja cząsteczki semantycznej BACT w ORM zakłada bezpośrednio odwzorowanie jego aspektów do kategorii semantycznych ORM z minimalizacją utraty semantyki metamodelu asocjacyjnego, tj. aspekt danych odwzorowany jest w obiekt, natomiast aspekt związku w zreifikowany fakt. Przykładową cząsteczkę translacyjną dla tego wzorca przedstawiono poniżej:

$$\begin{aligned}
 & *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}_{\text{AOM}}}^{[\text{Substitutability}]} = \\
 & \left( \begin{array}{c} \emptyset, \\ \left\{ \begin{array}{l} *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}}^{[\text{Substitutability (entity)}]} \triangleleft *_{\text{ORM}}^{[\text{obekt-byt}]}, \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}}^{[\text{Substitutability (role)}]} \triangleleft *_{\text{ORM}}^{[\text{rola}]}, \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{ORM}}^{[\text{Substitutability (fact)}]} \triangleleft *_{\text{ORM}}^{[\text{fakt}]} \end{array} \right\}, \\ \left\{ \begin{array}{l} \odot_{\text{esnm-orm:Substitutability (entity)}} \triangleleft \odot_{\text{orm:EntityType}} \\ \odot_{\text{esnm-orm:Substitutability (role)}} \triangleleft \odot_{\text{orm:Role}} \\ \odot_{\text{esnm-orm:Substitutability (fact)}} \triangleleft \odot_{\text{orm:FactType}} \end{array} \right\} \end{array} \right) \quad (8.63)
 \end{aligned}$$

Reprezentację gramatyczną powstałej struktury oznaczono na diagramie symbolami (8a), (8b), (8c).

### 8.4.3. Translacja semantyki do UML

Diagram na rysunku 8.4 przedstawia model  $\text{ESNM}^{\text{SKB}}$  wyrażony w ramach UML.

Przykłady zastosowania wzorców mapowania dla poszczególnych typów cząstek semantycznych AOM przedstawiono w kolejnych podrozdziałach.

#### 8.4.3.1. Translacja cząsteczki semantycznej kolekcji

Cząsteczka semantyczna modelu docelowego, która powstała poprzez odwzorowanie

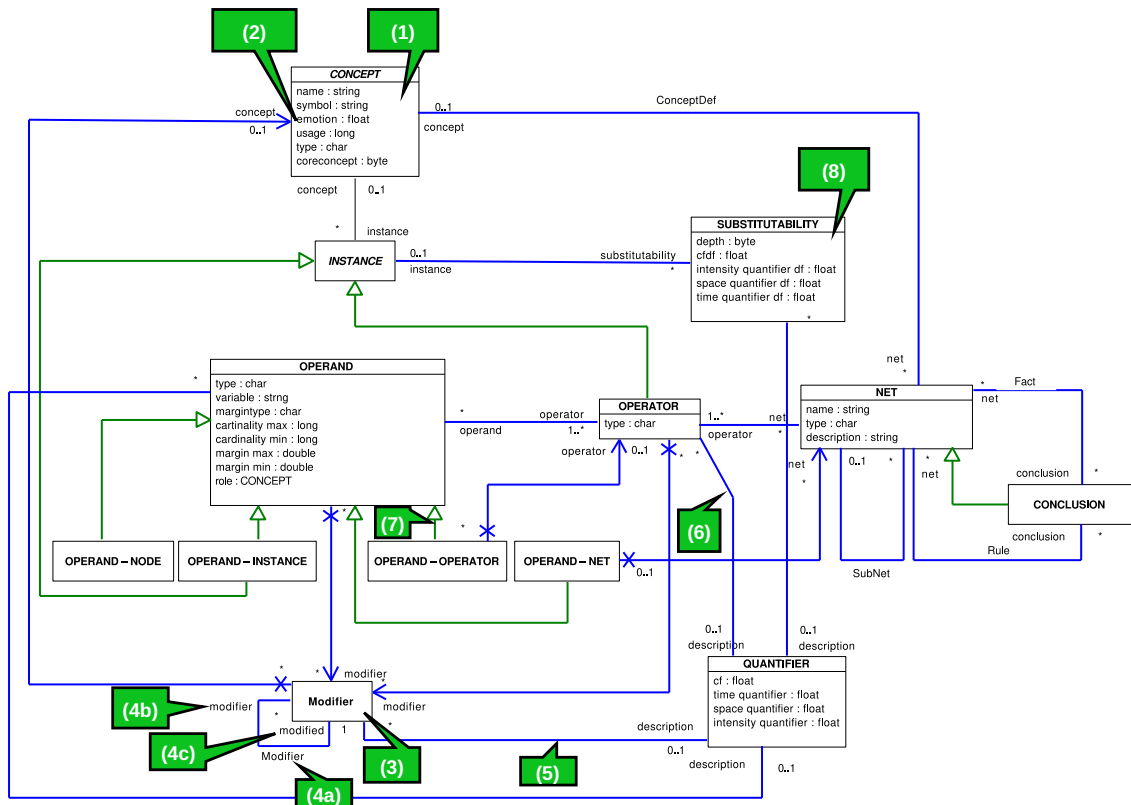
$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{AOM}}^{[\text{Concept}]}$  została przedstawiona poniżej:

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Concept}]} \triangleleft *_{\text{UML}}^{[\text{klasa}]} \quad (8.64)$$

$$\begin{aligned}
 & *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Concept}]} = \\
 & \left( \begin{array}{c} *_{\text{UML}}^{\{\odot 1, \odot 23\}}, \\ \emptyset, \\ \left\{ \odot_{\text{esnm-uml:Concept}} \triangleleft \odot_{\text{uml:Class}} \right\} \end{array} \right) \quad (8.65)
 \end{aligned}$$

Reprezentację gramatyczną powstałej struktury oznaczono na diagramie symbolem (1).





Rysunek 8.4. Model ESNM<sup>SKB</sup> po translacji na UML

### 8.4.3.2. Translacja cząsteczki semantycznej atrybutu

Przykładem cząsteczki semantycznej reprezentującej atrybut jest cząsteczka  $*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{AOM}} [\text{Concept's emotion}]$ . Jej reprezentacja w modelu docelowym została przedstawiona poniżej:

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Concept's emotion}] \triangleleft *_{\text{UML}} [\text{atrybut}] \quad (8.66)$$

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Concept's emotion}] = \left( \begin{array}{c} \{ \diamond_{12}, \diamond_{13}, \diamond_{14}, \diamond_{15}, \diamond_{16}, \diamond_{17}, \diamond_{18}, \diamond_{19}, \diamond_{20}, \diamond_{23} \} \\ \text{UML} \\ \emptyset, \\ \left\{ \begin{array}{l} \odot [\text{esnm-uml:Concept}] \triangleleft \odot [\text{uml:Class}], \\ \odot [\text{esnm-uml:Concept's emotion}] \triangleleft \odot [\text{uml:Property}], \\ \odot [\text{float}] \triangleleft \odot [\text{uml:Type}] \end{array} \right\} \end{array} \right) \quad (8.67)$$

Na diagramie atrybut oznaczono symbolem (2).

### 8.4.3.3. Translacja cząsteczki semantycznej asocjacji

Przykład cząsteczki semantycznej asocjacji zobrazowano dla asocjacji  $\diamond \text{Modifier}$ . Odzworowanie semantyki wykonano według wzorca mapowania, zgodnie z którym

część docelowa modelu UML może być klasą. Część semantyczna modelu docelowego została przedstawiona poniżej:

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Modifier (class)}]} \triangleleft *_{\text{UML}}^{[\text{klasa}]} \quad (8.68)$$

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Modifier (class)}]} = \left( \begin{array}{c} *_{\text{UML}}^{\{\diamond 1, \diamond 23\}}, \\ \emptyset, \\ \left\{ \odot_{\text{esnm-uml:Modifier (class)}} \triangleleft \odot_{\text{uml:Class}} \right\} \end{array} \right) \quad (8.69)$$

Wynikowe struktury oznaczono na diagramie symbolem **(3)**.

#### 8.4.3.4. Translacja części semantycznej roli

Część semantyczną reprezentującą rolę *Modifier* asocjacji  $\diamond$  *Modifier* reprezentuje przedstawiona poniżej część translacyjna. Ze względu na to, iż asocjacyjna rola została odwzorowana w asocjację, część ta składa się z trzech części modelu docelowego, jednej reprezentującej asocjację oraz dwóch reprezentujących jej role.

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}_{\text{AOM}}}^{[\text{Modifier's modifier}]} = \left( \begin{array}{c} \emptyset, \\ \left\{ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Modifier (assoc)}]} \triangleleft *_{\text{UML}}^{[\text{asocjacja}]}, \right. \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Modifier's modifier}]} \triangleleft *_{\text{UML}}^{[\text{rola}]}, \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Modifier's modified}]} \triangleleft *_{\text{UML}}^{[\text{rola}]} \left. \right\}, \\ \left\{ \odot_{\text{esnm-uml:Modifier (assoc)}} \triangleleft \odot_{\text{uml:Association}} \right. \\ \odot_{\text{esnm-uml:MODIFIER}} \triangleleft \odot_{\text{uml:Type}} \\ \odot_{\text{esnm-uml:Modifier's modifier}} \triangleleft [*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Modifier's modifier}]}] \odot_{\text{uml:Property}} \\ \left. \odot_{\text{esnm-uml:Modifier's modified}} \triangleleft [*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Modifier's modified}]}] \odot_{\text{uml:Property}} \right\} \end{array} \right) \quad (8.70)$$

Części modelu docelowego przedstawiono poniżej:

**(4a):**

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Modifier (assoc)}]} \triangleleft *_{\text{UML}}^{[\text{asocjacja}]} \quad (8.71)$$

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Modifier (assoc)}]} = \left( \begin{array}{c} *_{\text{UML}}^{\{\diamond 1, \diamond 23\}}, \\ \emptyset, \\ \left\{ \odot_{\text{esnm-uml:Modifier (assoc)}} \triangleleft \odot_{\text{uml:Association}} \right\} \end{array} \right) \quad (8.72)$$

(4b):

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Modifier's modifier}]} \triangleleft *_{\text{UML}}^{[\text{rola}]} \quad (8.73)$$

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Modifier's modifier}]} = \left( \begin{array}{c} *_{\text{UML}}^{\{\diamond 13, \diamond 14, \diamond 15, \diamond 2, \diamond 3, \diamond 4, \diamond 5, \diamond 6, \diamond 7, \diamond 8, \diamond 16, \diamond 17, \diamond 18, \diamond 19, \diamond 23\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot [\text{esnm-uml:Modifier (assoc)}] \triangleleft \odot [\text{uml:Association}] \\ \odot [\text{esnm-uml:Modifier's modifier}] \triangleleft \odot [\text{uml:Property}] \\ \odot [\text{esnm-uml:Modifier (class)}] \triangleleft \odot [\text{uml:Property}] \\ \odot [\underline{0}] \triangleleft \odot [\text{liczba całkowita}] \\ \odot [\underline{*}] \triangleleft \odot [\text{nieograniczona liczba całkowita}] \end{array} \right\} \end{array} \right) \quad (8.74)$$

(4c):

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Modifier's modified}]} \triangleleft *_{\text{UML}}^{[\text{rola}]} \quad (8.75)$$

$$*_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}}^{[\text{Modifier's modified}]} = \left( \begin{array}{c} *_{\text{UML}}^{\{\diamond 13, \diamond 14, \diamond 15, \diamond 2, \diamond 3, \diamond 4, \diamond 5, \diamond 6, \diamond 7, \diamond 8, \diamond 16, \diamond 17, \diamond 18, \diamond 19, \diamond 23\}}, \\ \emptyset, \\ \left\{ \begin{array}{l} \odot [\text{esnm-uml:Modifier (assoc)}] \triangleleft \odot [\text{uml:Association}] \\ \odot [\text{esnm-uml:Modifier's modified}] \triangleleft \odot [\text{uml:Property}] \\ \odot [\text{esnm-uml:Modifier (class)}] \triangleleft \odot [\text{uml:Property}] \\ \odot [\underline{1}] \triangleleft \odot [\text{liczba całkowita}] \\ \odot [\underline{1}] \triangleleft \odot [\text{nieograniczona liczba całkowita}] \end{array} \right\} \end{array} \right) \quad (8.76)$$

Symbolami (4a), (4b) i (4c) oznaczono wynikowe struktury na diagramie.

#### 8.4.3.5. Translacja cząsteczki semantycznej opisu asocjacji

Opis asocjacji  $\diamond \text{Modifier}$  został zrealizowany poprzez utworzenie asocjacji pomiędzy klasą reprezentującą opis, a reprezentacją asocjacji  $\diamond \text{Modifier}$  w postaci klasy. Reprezentuje to cząsteczka translacyjna:

$$\begin{aligned}
& *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}_{\text{AOM}}} [\text{Description of Modifier}] = \\
& \left( \begin{array}{c} \emptyset, \\ \left\{ \begin{array}{l} *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Description of Modifier}] \triangleleft *_{\text{UML}} [\text{asocjacja}], \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Description of Modifier's } r_1] \triangleleft *_{\text{UML}} [\text{rola}], \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Description of Modifier's } r_2] \triangleleft *_{\text{UML}} [\text{rola}], \end{array} \right\}, \\ \left\{ \begin{array}{l} \odot [\text{esnm-uml:modifier-description}] \triangleleft \odot [\text{uml:Association}] \\ \odot [\text{esnm-uml:modifier-description's modifier}] \triangleleft *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Description of Modifier's } r_1] \odot [\text{uml:Property}] \\ \odot [\text{esnm-uml:MODIFIER}] \triangleleft *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Description of Modifier's } r_1] \odot [\text{uml:Type}] \\ \odot [\text{esnm-uml:description of modifier}] \triangleleft *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Description of Modifier's } r_2] \odot [\text{uml:Property}] \\ \odot [\text{esnm-uml:QUANTIFIER}] \triangleleft *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Description of Modifier's } r_2] \odot [\text{uml:Type}] \end{array} \right\} \end{array} \right) \quad (8.77)
\end{aligned}$$

Cząsteczka ta składa się z trzech skonkretyzowanych cząsteczek semantycznych modelu docelowego, dwóch reprezentujących rolę oraz jedną reprezentującą asocjację. Asocjację wynikową oznaczono na diagramie symbolem (5).

#### 8.4.3.6. Translacja cząsteczki semantycznej opisu roli

Opis roli *Operator* asocjacji  $\diamond$ *Operator* poprzez kolekcję  $\square$ *QUANTIFIER* stanowi podstawę budowy jednej z cząsteczek źródłowych. Jej odwzorowanie w cząsteczkę translacyjną UML można przedstawić następująco:

$$\begin{aligned}
& *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}_{\text{AOM}}} [\text{Description of Operator}] = \\
& \left( \begin{array}{c} \emptyset, \\ \left\{ \begin{array}{l} *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Description of Operator}] \triangleleft *_{\text{UML}} [\text{asocjacja}], \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Description of Operator's } r_1] \triangleleft *_{\text{UML}} [\text{rola}], \\ *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Description of Operator's } r_2] \triangleleft *_{\text{UML}} [\text{rola}], \end{array} \right\}, \\ \left\{ \begin{array}{l} \odot [\text{esnm-uml:operator-description}] \triangleleft \odot [\text{uml:Association}] \\ \odot [\text{esnm-uml:operator-description's operator}] \triangleleft *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Description of Operator's } r_1] \odot [\text{uml:Property}] \\ \odot [\text{esnm-uml:OPERATOR}] \triangleleft *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Description of Operator's } r_1] \odot [\text{uml:Type}] \\ \odot [\text{esnm-uml:description of operator}] \triangleleft *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Description of Operator's } r_2] \odot [\text{uml:Property}] \\ \odot [\text{esnm-uml:QUANTIFIER}] \triangleleft *_{\text{ESNM}^{\text{SKB}} \triangleleft \text{UML}} [\text{Description of Operator's } r_2] \odot [\text{uml:Type}] \end{array} \right\} \end{array} \right) \quad (8.78)
\end{aligned}$$

Cząsteczka ta składa się z trzech skonkretyzowanych cząsteczek semantycznych modelu docelowego, dwóch reprezentujących rolę oraz jedną reprezentującą asocjację. Asocjację wynikową oznaczono na diagramie symbolem (6).

### 8.4.3.7. Translacja cząsteczki semantycznej dziedziczenia

Cząsteczka semantyczna dziedziczenia reprezentująca dziedziczenie  $\diamond$ *OperandOperator* po  $\diamond$ *Operand* została odwzorowana do następującej skonkretyzowanej cząstki modelu docelowego:

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \end{matrix} \begin{matrix} [\text{OperandOperator's generalization from Operand}] \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{UML} \end{matrix} \begin{matrix} [\text{generalizacja}] \end{matrix} \quad (8.79)$$

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \end{matrix} \begin{matrix} [\text{OperandOperator's generalization from Operand}] \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{UML} \end{matrix} = \left( \begin{matrix} \begin{matrix} \begin{matrix} \otimes \\ \text{UML} \end{matrix} \{ \diamond 21, \diamond 22 \}, \\ \emptyset, \\ \left\{ \begin{matrix} \odot \underline{[\text{esnm-uml:OperandOperator's gen from Operand}]} \triangleleft \odot \underline{[\text{uml:Generalization}]} \\ \odot \underline{[\text{esnm-uml:OPERAND}]} \triangleleft \odot \underline{[\text{generalizacja}]} \\ \odot \underline{[\text{esnm-uml:OPERAND-OPERATOR}]} \triangleleft \odot \underline{[\text{specjalizacja}]} \end{matrix} \right\} \end{matrix} \end{matrix} \right) \quad (8.80)$$

Na diagramie 8.4 wynikowe struktury oznaczono symbolem (7).

### 8.4.3.8. Translacja cząsteczki semantycznej BACT

Cząsteczka BACT reprezentująca substytucyjność (wyr. 8.11) została odwzorowana do następującej skonkretyzowanej cząsteczki modelu docelowego:

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \end{matrix} \begin{matrix} [\text{Substitutability}] \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{UML} \end{matrix} \begin{matrix} [\text{klasa}] \end{matrix} \quad (8.81)$$

$$\begin{matrix} * \\ \text{ESNM}^{\text{SKB}} \end{matrix} \begin{matrix} [\text{Substitutability}] \\ \triangleleft \end{matrix} \begin{matrix} * \\ \text{UML} \end{matrix} = \left( \begin{matrix} \begin{matrix} \begin{matrix} \otimes \\ \text{UML} \end{matrix} \{ \diamond 1, \diamond 23 \}, \\ \emptyset, \\ \left\{ \odot \underline{[\text{esnm-uml:SUBSTITUTABILITY}]} \triangleleft \odot \underline{[\text{uml:Class}]} \right\} \end{matrix} \end{matrix} \right) \quad (8.82)$$

Na diagramie 8.4 wynikowe struktury oznaczono symbolem (8).

## 8.5. Ewaluacja translacji semantyki modelu

Zdefiniowane w rozdziale 6.3 miary zostały użyte do oceny jakości translacji semantyki modelu  $\text{ESNM}^{\text{SKB}}$  dla wszystkich metamodeli wynikowych. Ewaluacja polegała na ocenie odwzorowania każdej skonkretyzowanej cząsteczki semantycznej w świetle realizacji przez nią atomów semantycznych, jak również możliwości doboru odpowia-

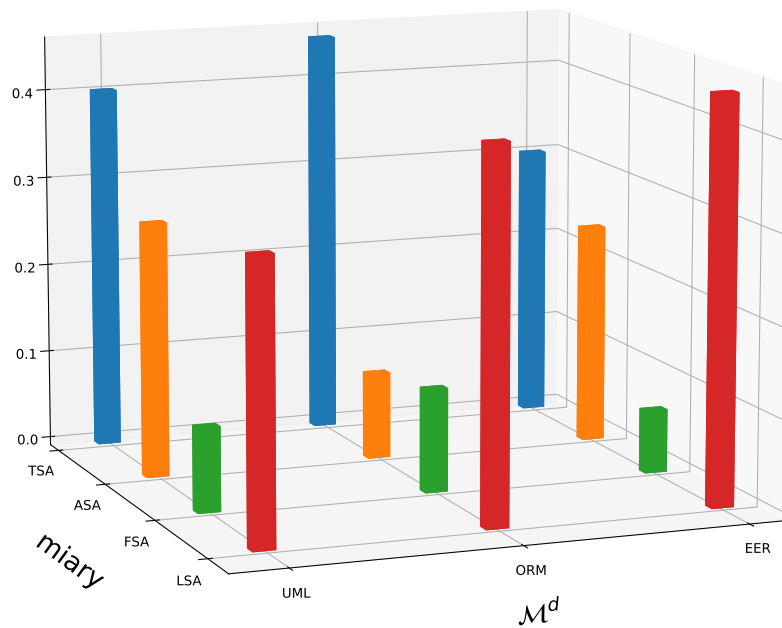
dających sobie par pojęć. Zebrane dane pozwoliły dokonać ewaluacji, której wyniki przedstawiono w niniejszym podrozdziale.

### 8.5.1. Miary pierwszego poziomu

Tabela 8.2 oraz rysunki 8.5 i 8.6 zawierają wyniki ewaluacji dla miar pierwszego poziomu.

Tablica 8.2. Wyniki ewaluacji translacji modelu ESNM<sup>SKB</sup> miarami pierwszego poziomu

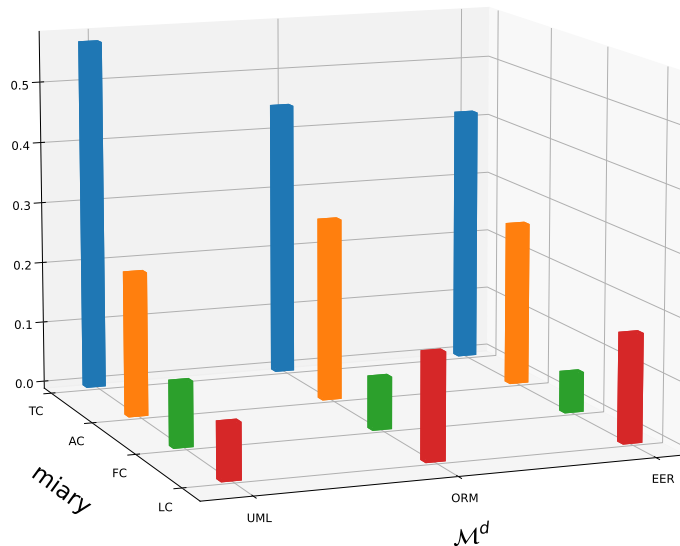
$\mathcal{M}^d$	TSA	ASA	FSA	LSA	TC	AC	FC	LC
EER	0.303	0.245	0.070	0.450	0.417	0.266	0.065	0.175
ORM	0.450	0.097	0.115	0.413	0.448	0.296	0.085	0.172
UML	0.404	0.284	0.094	0.312	0.571	0.233	0.105	0.090



Rysunek 8.5. Wyniki ewaluacji miarami  $TSA_{m^s, m^d}$ ,  $ASA_{m^s, m^d}$ ,  $FSA_{m^s, m^d}$  oraz  $LSA_{m^s, m^d}$

Najwyższą wartość miary  $TSA_{m^s, m^d}$  zauważono dla modelu wyrażonego w ORM. Na tę sytuację wpłynął z pewnością fakt, iż semantyka asocjacyjnego pojęcia [aom:Role](#) posiadającego dużą liczbę atomów semantycznych została dobrze odwzorowane w ramach ORM. Najniższą wartość zaobserwowano dla EER, na który

z kolei okazało się trudno przełożyć semantykę pojęcia aom:Attr, które to występowało w modelu najliczniej. Miara  $ASA_{m^s, m^d}$  osiągnęła najwyższą wartość dla UML. Najprawdopodobniej, odpowiada za to fakt, iż semantyka uml:Property UML jest zbliżona do asocjacyjnej roli, lecz spora część atomów semantycznych ją zmienia. Miara  $FSA_{m^s, m^d}$  ma niską wartość dla każdego metamodelu. ORM osiągnął najwyższą wartość dla tej miary, ze względu na wymuszenia semantyczne związane z realizacją licznych atrybutów. EER osiągnął najwyższą wartość miary  $LSA_{m^s, m^d}$  ze względu na względnie najniższą możliwość realizacji asocjacyjnej semantyki roli i atrybutu. Miary  $TC_{m^s, m^d}$ ,  $AC_{m^s, m^d}$ ,  $FC_{m^s, m^d}$ ,  $LC_{m^s, m^d}$  dotyczyły odwzorowywania



Rysunek 8.6. Wyniki ewaluacji miarami  $TC_{m^s, m^d}$ ,  $AC_{m^s, m^d}$ ,  $FC_{m^s, m^d}$  oraz  $LC_{m^s, m^d}$

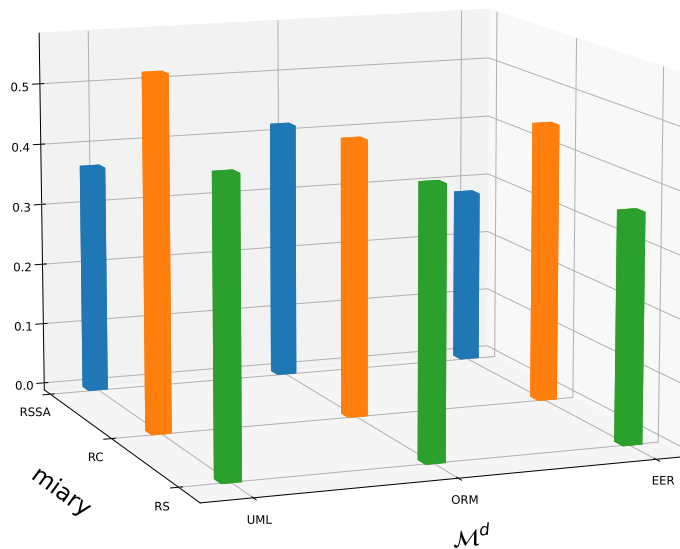
odpowiadających sobie pojęć. UML został najwyżej oceniony miarą  $TC_{m^s, m^d}$  oraz najniżej miarą  $LC_{m^s, m^d}$ . Wynika to ze względnie najlepszego dopasowania pojęć UML uwzględniających obiektowy paradygmat modelowania do modelowania asocjacyjnego, które stanowi pewnego rodzaju rozwinięcie obiektowości. Miara zmienionych pojęć modelu  $ASA_{m^s, m^d}$  jest najwyższa dla ORM, który stanowi model bezatrybutowy, w związku z tym koncepcje asocjacyjne zawarte w rozpatrywanym modelu zostały odwzorowane w sposób najbardziej zmieniony. W ramach ORM doszło również do największej liczby wymuszeń nowej semantyki, w związku z tym miara  $FC_{m^s, m^d}$  osiągnęła dla tego metamodelu największą wartość.

### 8.5.2. Miary semantyki wynikowej

Tabela 8.3 oraz rysunek 8.7 przedstawiają wyniki ewaluacji miarami pochodnymi, które posłużyły do oceny jakości translacji semantyki w modelach wynikowych.

Tablica 8.3. Wyniki ewaluacji translacji modelu  $ESNM^{SKB}$  miarami wynikowymi

$\mathcal{M}^d$	RSSA	RC	RS
EER	0.284	0.452	0.368
ORM	0.419	0.448	0.433
UML	0.369	0.571	0.470



Rysunek 8.7. Wyniki ewaluacji miarami  $RSSA$ ,  $RC$  oraz  $RS$

Miara  $RSSA_{m^s, m^d}$  jest najwyższa dla ORM, który najlepiej oddał atomy semantyczne AOM w przypadku realizacji modelu  $ESNM^{SKB}$ . Miara  $RC_{m^s, m^d}$  dotycząca par pojęć najkorzystniej oceniła UML, co wpłynęło również na najwyższy wynik miary  $RS_{m^s, m^d}$ . Należy zatem stwierdzić, iż odwzorowanie w UML całościowo najlepiej oddało pierwotną semantykę modelu  $ESNM^{SKB}$  z uwzględnieniem dopasowania pojęć modelu, choć ORM lepiej odwzorował poszczególne atomy semantyczne. Najmniej przystająca pod tym względem jest implementacja  $ESNM^{SKB}$  w metamodelu EER.



## 8.6. Podsumowanie

W niniejszym rozdziale przedstawiono studium przypadku metod modelowania i translacji semantyki opracowanych w niniejszej pracy. Studium zostało przeprowadzone w oparciu o fragment systemu reprezentacji wiedzy ESNM<sup>SKB</sup>. W kontekście wzorców modelowania asocjacyjnego, należy zauważyć, iż ESNM<sup>SKB</sup> nie był budowany przy ich użyciu a analiza wykonana w ramach niniejszej pracy stanowi pewnego rodzaju „nałożenie” warstwy wzorców na istniejący model. Liczbę w ten sposób zidentyfikowanych wzorców należy uznać za wysoką, co skłania do konkluzji, że modelowanie asocjacyjne samoistnie sprzyja rozwiązywaniu problemów w sposób uwzględniający ponowne użycie. Również bardzo istotny jest fakt, że przedstawiony w niniejszej pracy zestaw wzorców modelowania asocjacyjnego jest reprezentatywny w świetle problemów powstających podczas modelowania.

Przedstawione studium przypadku zawiera również egzemplifikację metody translacji semantyki do metamodeli EER, ORM, UML. Każda translacja została wzbogacona o przedstawienie diagramu zawierającego wynikowy model. Modele te, będące wynikiem translacji semantyki, zdecydowanie lepiej przedstawiają semantykę źródłowego metamodelu, niż w przypadku zastosowania mechanicznych odwzorowań opartych jedynie o kwestie syntaktyczne. Poziom zgodności odwzorowania został zmierzony za pomocą opracowanych miar, co stanowi istotną wartość dodaną przedstawionego rozwiązania.

## 9. Podsumowanie

Myślą przewodnią, która przyświecała prowadzonym badaniom, była chęć zaspokojenia potrzeby w zakresie specyfikacji semantyki metamodeli i modeli danych w sposób precyzyjny, jednoznaczny i mierzalny. W związku z tym, pierwszym elementem proponowanego rozwiązania stał się system pojęć **CLoM**. Stanowi on podstawę dalszych rozważań i rozwiązań. Zapełnia on lukę badawczą dotyczącą *problemu specyfikacji konceptualizacji domeny modelowania danych*. System CLoM składa się ze zdefiniowanych pojęć oraz związków między nimi. Każdemu pojęciu przypisano terminy umożliwiające odwoływanie się do nich, np. w ramach definicji i ograniczeń wyrażonych przy użyciu SBVR. Pojęcia zostały podzielone na obszary tematyczne dotyczące takich zagadnień jak modelowanie *klasyfikatorów, kategoryzacji, związków, bytów, zbiorowości, cech, metamodelowanie*. CLoM jest systemem otwartym o charakterze półformalnym, który nie stanowi kolejnego metamodelu, lecz definiuje pewien punkt widzenia na dziedzinę modelowania. Został on zdefiniowany i opisany w rozdziale 3. System nie stanowi jedynie nadzbioru rozwiązań istniejących w konkretnych metamodelach, lecz dotyka istotny modelowania, dokonując m.in. abstrakcji istniejących rozwiązań. Istotność zaproponowanego rozwiązania wynika zarówno z faktu braku specyfikacji konceptualizacji domeny modelowania, jak również potrzeby zapewnienia precyzyjnego opisu semantyki m.in. w celu oceny ilościowej jakości procesu translacji modeli. Jak wcześniej wspomniano, CLoM jest systemem otwartym i może być uzupełniany o kolejne pojęcia występujące w domenie.

Kolejnym istotnym elementem wkładu naukowego jest **specyfikacja wzorców modelowania** opartych o system CLoM. Proponowane rozwiązanie dotyczy *problemu wzorców dla metamodelu asocjacyjnego*. Zaproponowano szereg wzorców modelowania o charakterze zarówno abstrakcyjnym, jak również dedykowanym metamodelowi asocjacyjnemu. Opis każdego z wzorców składa się z opisu problemu, który rozwiązuje, jego elementów składowych (uczestników) oraz własności, specyfikacji w systemie pojęć uwzględniającej konieczne atomy semantyczne wzorca, jak również specyfikację atomów semantycznych w ramach własności możliwych. Każdy wzorzec posiada opis implementacji w AOM odnoszący się do jego specyfikacji semantycznej oraz przystępny przykład zastosowania. Szczegółowy opis opracowanych wzorców można odnaleźć w rozdziale 4. Wzorce modelowania asocjacyjnego pozwalają na do-

głębne zrozumienie wewnętrznych własności modelu asocjacyjnego poprzez analizę możliwości realizacji konkretnych rozwiązań w ramach tworzenia modeli. Wzorce są elementem wartościowym, gdyż dostarczają rozwiązania typowych, powtarzających się problemów w ramach modelowania, tym samym wpływają na wydajność i jakość procesu modelowania. W ramach dalszych prac możliwe jest tworzenie wzorców, w szczególności wzorców bardziej złożonych, składających się z już opracowanych.

W pracy zaproponowano również **metodę ekstrakcji semantyki metamodeli danych**. Element ten dotyka zarówno zagadnienia konkretyzacji metamodeli w kontekście CLoM, jak również umożliwia dostarczenie nośników semantyki: pojęć, atomów semantycznych i cząsteczek semantycznych w celu rozwiązania *problemu translacji semantyki modeli*. Istotną luką badawczą, która została wypełniona za pomocą zaproponowanego rozwiązania, jest problem określania zdolności metamodeli do wyrażania semantyki. Szczególnie ważnym podkreślenia jest fakt, iż rozwiązanie to skupia się na dekompozycji semantyki do poziomu atomów semantycznych oraz samych pojęć. Szczegółowy opis metody zawarto w rozdziale 5.1. Metodę tę zastosowano dla następujących metamodeli: AOM (rozdz. 5.2) (w zakresie pełnego opisu części intencjonalnej), EER (rozdz. 5.3.1), ORM (rozdz. 5.4.1) oraz UML (rozdz. 5.5.1). Opisy semantyki EER, ORM oraz UML zostały dostosowane do ram semantycznych AOM. Rozwiązanie to jest istotne ze względu na możliwość dotarcia do istoty konstrukcji semantycznych poszczególnych metamodeli. Kluczowa w tym momencie jest dekompozycja do poziomu atomowego tak, aby nie operować na pojęciach i mechanizmach różnie rozumianych w różnych metamodelach, a skupić się na możliwie jak najbardziej elementarnych koncepcjach, z których są one zbudowane. Naturalnym kierunkiem dalszych prac jest ekstrakcja semantyki dla innych, istniejących metamodeli danych oraz ewentualnie tworzenia nowych metamodeli w oparciu o specyfikację ich semantyki przy użyciu CLoM.

**Metoda translacji semantyki** stanowi istotny element wkładu naukowego proponowanego rozwiązania. Rozwiązuje ona problem badawczy określony w rozdziale 1.4 jako *problem translacji semantyki modeli*. W ramach opisu metody przedstawiono pełną koncepcję oraz w sposób formalny i precyzyjny wyspecyfikowano kolejne przekształcenia. Metodę tę zastosowano w celu dokonania translacji metamodelu asocjacyjnego do następujących metamodeli: EER, ORM, UML. Do najważniejszych właściwości metody należy jej podział na zagadnienie dotyczące translacji semantyki metamodeli oraz translacji modeli. W centrum metody znajdują się wzorce mapowania składające się z reguł translacji oparte ściśle o CLoM. Metodę wraz z zastosowaniem dla wskazanych metamodeli przedstawiono w rozdziale 5. Istotność opracowanej metody leży głównie w tym, iż translacja oparta o semantykę z zało-

żenia ma za zadanie jak najwierniejsze odwzorowanie koncepcji myślowych zawartych w metamodelu/modelu źródłowym na postać metamodelu/modelu docelowego. Dalsze kierunki prac w tym obszarze mogą dotyczyć translacji semantyki AOM do kolejnych metamodeli, jak również opracowanie translacji dla innych metamodeli.

Ostatnim istotnym elementem wkładu naukowego jest **metoda ewaluacji translacji semantyki**. Dotyczy ona problemu badawczego określonego jako *problem miary zmiany semantyki*. Jak już wspomniano, każda translacja wiąże się potencjalnie ze zmianą semantyki, a pomiar tej zmiany nie ma swojego odzwierciedlenia w literaturze przedmiotu. W ramach prowadzonych badań opracowano szereg metryk pozwalających dokonywać oceny jakości przeniesienia semantyki, zarówno na poziomie analitycznym, jak również syntetycznym. Poziom analityczny dotyczy utraty semantyki, zmiany semantyki, wymuszenia nowej semantyki oraz zgodności semantycznej, natomiast poziom syntetyczny dostarcza rozwiązań agregujących wymienione elementy. Wspomniane miary funkcjonują na różnych poziomach, takich jak: poziom cząsteczek semantycznych zasilających wzorce mapowania, poziom metamodelu oraz modelu. Szczegółowy opis tego zagadnienia znajduje się w rozdziale 6, a jej zastosowanie dla ewaluacji translacji semantyki metamodelu asocjacyjnego zawarto w rozdziale 7. Istotność podejmowanego zagadnienia polega na tym, iż dzięki dostarczonemu rozwiązaniu możliwe jest porównywanie zgodności semantycznej modeli, metamodeli, jak również poszczególnych cząsteczek semantycznych. Dalsze prace mogą iść w kierunku budowania bardziej złożonych miar na bazie zaproponowanych w niniejszym opracowaniu.

Cel pracy, określony jako rozwiązanie problemów badawczych zdefiniowanych w rozdziale 1.4 został osiągnięty. Rezultaty prowadzonych badań poddano weryfikacji poprzez wykonanie analizy studium przypadku. W toku prowadzonych badań i przeprowadzonych analiz konkretnych przypadków, przyjęte w rozdziale 1.3 założenia pracy okazały się trafne i uzasadnione.

# **Dodatki**

## A. Przyjęta konwencja użycia wyrażen SBVR

Niniejszy dodatek specyfikuje wykorzystany w pracy sposób odnoszenia się do pojęć zdefiniowanych w ramach standardu SBVR przy użyciu języka polskiego. Jest to zabieg techniczny, którego zadaniem jest ułatwić czytelność przygotowanych definicji oraz reguł biznesowych w taki sposób, by mapować czytelne i jednoznaczne sformułowania w języku naturalnym na reprezentacje zdefiniowane w [113]. Na potrzeby swobody reprezentacji przyjęto, iż konwencja użycia SBVR nie obejmuje zagadnień fleksji występującej w języku polskim. Wszelkie formy wyrazowe powstałe poprzez koniugację, bądź deklinację mają znaczenie zgodne z formą zlematyzowaną.

W pracy następuje odwołanie do szeregu pojęć, których definicji nie dostarczono. Dotyczy to pojęć o powszechnie rozumianej semantyce, którą można rozumieć niejawnie. Takie pojęcia należy rozumieć jako [implicitly-understood concept](#). Szereg pojęć ze słownika zdefiniowanego w ramach SBVR zostało użytych w sposób przetłumaczony na języka polski.

### Wyrażenia

Deklaracje oparte o SBVR rozróżnialne są ze względu na stylizację czcionki, która wizualnie określa formalny charakter stwierdzenia. Poniżej przedstawiono wykorzystane rodzaje czcionek zaadaptowanych z [113]:

[termin](#)

**[Nazwa](#)**

*[czasownik](#)*

**[słowo kluczowe](#)**

### Słowa kluczowe i frazy określające sformułowania logiczne

Poniżej przedstawione frazy oraz słowa kluczowe określają rodzaje sformułowań logicznych. Wykorzystane symbole  $n$  oraz  $m$  reprezentują użycie literałów zawierających liczby całkowite. Symbole  $p$  i  $q$  reprezentują wyrażenia zdań logicznych.

## Kwantyfikacje

każdy, wszystkie	<a href="#">universal quantification</a>
pewien	<a href="#">existential quantification</a>
przynajmniej jeden	<a href="#">existential quantification</a>
przynajmniej $n$	<a href="#">at-least-n quantification</a>
co najwyżej jeden	<a href="#">at-most-one quantification</a>
co najwyżej $n$	<a href="#">at-most-n quantification</a>
dokładnie jeden	<a href="#">exactly-one quantification</a>
dokładnie $n$	<a href="#">exactly-n quantification</a>
przynajmniej $n$ i co najwyżej $m$	<a href="#">numeric range quantification</a>
więcej niż jeden	<a href="#">at-least-n quantification</a> przy $n = 2$
żaden	<a href="#">logical negation</a> , w ramach którego <a href="#">logical operand</a> stanowi <a href="#">existential quantification</a>

## Operacje logiczne

nieprawda, że	<a href="#">logical negation</a>
$p$ i $q$	<a href="#">conjunction</a>
$p$ oraz $q$	<a href="#">conjunction</a>
$p$ lub $q$	<a href="#">disjunction</a>
albo $p$ albo $q$	<a href="#">exclusive disjunction</a>
jeżeli $p$ to $q$	<a href="#">implication</a>
$p$ wtedy i tylko wtedy gdy $q$	<a href="#">equivalence</a>
ani $p$ ani $q$	<a href="#">nor formulation</a>

Jeżeli podmiot w wyrażeniu używającym ‘i’ (‘oraz’) lub ‘lub’ powtarza się, może zostać pominięty. Przykładowo, stwierdzenie, że „Metamodel posiada właściwość<sub>1</sub> i metamodel posiada właściwość<sub>2</sub>” można zapisać jako „Metamodel posiada właściwość<sub>1</sub> i właściwość<sub>2</sub>”. Słowo kluczowe ‘nie’ użyte przed czasownikiem wprowadza [logical negation](#).

## Operacje modalne

jest obowiązkowe, aby $p$	<a href="#">obligation formulation</a>	
jest zabronione, aby $p$	<a href="#">obligation formulation</a>	z zagnieżdżonym
	<a href="#">logical negation</a>	

jest konieczne, aby $p$	<u>necessity formulation</u>		
jest niemożliwe, aby $p$	<u>necessity formulation</u>	z	zagnieżdżonym
	<u>logical negation</u>		
jest dozwolone, aby $p$	<u>permissibility formulation</u>		
jest możliwe, aby $p$	<u>possibility formulation</u>		

Następujące słowa kluczowe używane są w ramach wyrażeń posiadających czasownik, aby utworzyć czasowniki złożone dodające operację modalną.

... musi ...	<u>obligation formulation</u>		
... nie może ...	<u>obligation formulation</u>	z	zagnieżdżonym
	<u>logical negation</u>		
... nie musi ...	zanegowanie <u>obligation formulation</u>		
... zawsze ...	<u>necessity formulation</u>		
... nigdy ...	<u>necessity formulation</u>	z	zagnieżdżonym
	<u>logical negation</u>		
... może ...	<u>possibility formulation</u>		
... posiada możliwość ...	<u>permissibility formulation</u>		

### Pozostałe słowa kluczowe

taki, że, który	użyte po desygnacji pojęcia rzeczownikowego i przed desygnacją pojęcia czasownikowego, oznacza wprowadzenie ograniczenia na rzeczy oznaczonej poprzez poprzednią desygnację bazującą na faktach o tejże rzeczy, zamiennie używana jest forma imiesłowowa,
dany	<u>universal quantification</u> , który został wyciągnięty poza logiczne sformułowanie w celu reprezentacji jednej rzeczy w czasie.



## B. Opis notacji symbolicznej

W niniejszym dodatku opisane zostaną zastosowane konwencje opisu symbolicznego w ramach opracowanej metody translacji.

### atom semantyczny

$\text{atom}_{\text{kontekst}}$  – atom semantyczny *atom* zdefiniowany w kontekście *kontekst*, gdzie kontekst oznacza metamodel lub wzorzec modelowania.

*Przykład:*

osoba jest konkretyzacją typu bytu

$\text{prze}^{\text{[1]}}_{\text{siębiorstwo}}$

Oznacza atom semantyczny numer 1 w kontekście przedsiębiorstwa, który jest werbalizowany jako osoba jest konkretyzacją typu bytu.

### możliwy atom semantyczny

$\text{atom}_{\text{kontekst}}$  – możliwy atom semantyczny *atom* zdefiniowany w kontekście *kontekst*, gdzie kontekst oznacza metamodel lub wzorzec modelowania.

*Przykład:*

osoba może posiadać stanowisko

$\text{prze}^{\text{[1]}}_{\text{siębiorstwo}}$

Oznacza możliwy atom semantyczny numer 1 w kontekście przedsiębiorstwa, który jest werbalizowany jako osoba może posiadać stanowisko.

### aktywowany atom semantyczny

$\text{atom}_{\text{kontekst}}$  – aktywowany atom semantyczny *atom* zdefiniowany w kontekście *kontekst*, gdzie kontekst oznacza metamodel lub wzorzec modelowania.

*Przykład:*

$\text{AOM}^3$  (B.1)

Oznacza, iż atom możliwy  $\text{AOM}^3$  został aktywowany.

### dezaktywowany atom semantyczny

$\text{atom}_{\text{kontekst}}$  – dezaktywowany atom semantyczny *atom* zdefiniowany w kontekście *kontekst*, gdzie kontekst oznacza metamodel lub wzorzec modelowania.

*Przykład:*

$\text{AOM}^3$  (B.2)

Oznacza, iż atom możliwy  $\ast_{\text{AOM}}^{\diamond 3}$  został dezaktywowany.

### odwołanie się do pojęcia

$\odot_{[\text{pojęcie}]}$  – odwołanie się do pojęcia [pojęcie](#).

*Przykład:*

$$\odot_{[\text{aom:Assoc}]} \quad (\text{B.3})$$

Oznacza odwołanie się do pojęcia [aom:Assoc](#).

### kategoryzacja pojęć

$<$  – operator podstawienia pojęcia ze względu na zależność kategoryzacyjną.

*Przykład:*

$$\odot_{[\text{aom:Assoc}]} < \odot_{[\text{aom:Basenode}]} \quad (\text{B.4})$$

Oznacza, że  $\odot_{[\text{aom:Assoc}]}$  jest kategoryzowana przez  $\odot_{[\text{aom:Basenode}]}$ .

### konkretyzacja pojęć

$\triangleleft$  – operator podstawienia pojęcia ze względu na konkretyzację.

*Przykład:*

$$\odot_{[\text{osoba}]} \triangleleft \odot_{[\text{aom:Assoc}]} \quad (\text{B.5})$$

Oznacza, że  $\odot_{[\text{osoba}]}$  konkretyzuje  $\odot_{[\text{aom:Assoc}]}$ .

### wynikanie semantyczne

$\vDash^{\mathcal{M}}$  – operator wynikania semantycznego użyty do wykazania konsekwencji atomu semantycznego abstrahującego od konkretnego metamodelu. Wykazanie to ma na celu wyrażenie semantyki atomu w ramach metamodelu  $\mathcal{M}$ .

*Przykład:*

$$\ast_{\text{przedsiębiorstwo}}^{[1]} \vDash^{\text{AOM}} \odot_{[\text{osoba}]} \triangleleft \odot_{[\text{aom:Assoc}]} \quad (\text{B.6})$$

Oznacza, że semantyka  $\ast_{\text{przedsiębiorstwo}}^{[1]}$  dla metamodelu AOM jest wyrażona jako

$\odot_{[\text{osoba}]}$  konkretyzuje  $\odot_{[\text{aom:Assoc}]}$ .

### cząsteczka semantyczna

$\ast_{\mathcal{M}}^{[\text{cząsteczka}]}$  – cząsteczka semantyczna *cząsteczka* zdefiniowana w kontekście metamodelu  $\mathcal{M}$ .

*Przykład:*

$$\ast_{\text{AOM}}^{[\text{osoba}]} \quad (\text{B.7})$$

Oznacza, że cząsteczkę semantyczną *osoba* zdefiniowaną w kontekście metamodelu

AOM.

*Przykład:*

$$\ast_{\text{AOM}}^{[\text{osoba}]} = \langle \emptyset, \{\ast_{\text{AOM}}^{[\text{asocjacja}]}\}, \{\ominus_{[\text{osoba}]} \triangleleft \ominus_{[\text{aom:Assoc}]}\} \rangle \quad (\text{B.8})$$

Oznacza, że  $\ast_{\text{AOM}}^{[\text{osoba}]}$  posiada definicję w postaci trójki: pustego zbioru atomów semantycznych metamodelu, jednoelementowego zbioru cząsteczek semantycznych składającego się z  $\ast_{\text{AOM}}^{[\text{kolekcja}]}$  oraz jednoelementowego zbioru pojęć składającego się z konkretyzacji  $\ominus_{[\text{osoba}]} \triangleleft \ominus_{[\text{aom:Coll}]}$ .

### aktywacja lub dezaktywacja atomu w cząsteczce semantycznej

$\ast_{\mathcal{M}}^{[\text{cząsteczka}]} [\text{wyrażenie}]$  – cząsteczka semantyczna *cząsteczka* zdefiniowana w kontekście metamodelu  $\mathcal{M}$ , w ramach której obowiązuje *wyrażenie* będące zestawem aktywacji lub dezaktywacji atomów semantycznych.

*Przykład:*

$$\ast_{\text{AOM}}^{[\text{rola}]} \left[ \ast_{\text{AOM}}^{\diamond 5 \diamond 6} \right] \quad (\text{B.9})$$

Oznacza, że cząsteczkę semantyczną *rola* zdefiniowaną w kontekście metamodelu AOM, w ramach której aktywowano możliwy atom semantyczny  $\ast_{\text{AOM}}^{\diamond 5}$  oraz dezaktywowano możliwy atom semantyczny  $\ast_{\text{AOM}}^{\diamond 6}$ .

### określenie kontekstu konkretyzacji

$\triangleleft_{[\text{kontekst}]}$  – modyfikator operatora konkretyzacji ze względu na określony kontekst *kontekst*, gdzie *kontekst* stanowi zbiór atomów semantycznych, w ramach którego dane podstawienie obowiązuje.

*Przykład:*

$$\ominus_{[\text{osoba}]} \triangleleft_{[\ast_{\text{AOM}}^{[2]}]} \ominus_{[\text{aom:Assoc}]} \quad (\text{B.10})$$

Oznacza, że konkretyzacja  $\ominus_{[\text{aom:Assoc}]}$  do  $\ominus_{[\text{osoba}]}$  obowiązuje w ramach atomu semantycznego  $\ast_{\text{AOM}}^{[2]}$ .

### określenie kontekstu kategoryzacji

$<_{[\text{kontekst}]}$  – modyfikator operatora kategoryzacji ze względu na określony kontekst *kontekst*, gdzie *kontekst* stanowi zbiór atomów semantycznych, w ramach którego dane podstawienie obowiązuje.

*Przykład:*

$$\ominus_{[\text{aom:Assoc}]} <_{[\ast_{\text{AOM}}^{[13]}]} \ominus_{[\text{aom:Basenode}]} \quad (\text{B.11})$$

Oznacza, że kategoryzacja  $\ominus_{[\text{aom:Basenode}]}$  jako  $\ominus_{[\text{aom:Assoc}]}$  obowiązuje w ramach atomu semantycznego  $\ast_{\text{AOM}}^{[13]}$ .

## Bibliografia

- [1] D. H. Akehurst, S. Kent i O. Patrascoiu. „A relational approach to defining and implementing transformations between metamodels”. W: *Softw. Syst. Model.* 2.4 (2003), s. 215–239.
- [2] G. N. Allen i S. T. March. „A Critical Assessment of the Bunge-Wand-Weber Ontology for Conceptual Modeling”. W: *SSRN Electronic Journal* (2007).
- [3] P. Atzeni, P. Cappellari, R. Torlone, P. A. Bernstein i G. Gianforme. „Model-independent schema translation”. W: *The VLDB Journal* 17.6 (lip. 2008), s. 1347–1370.
- [4] A. Bader, N. Reneacute i T. Bernhard. „Pattern-Based Conceptual Data Modelling”. W: *Frontiers in Artificial Intelligence and Applications* 272. Information Modelling and Knowledge Bases XXVI (2014), s. 1–20. ISSN: 0922-6389.
- [5] S. Bagui. „Extended Entity Relationship Modeling”. W: *Encyclopedia of Database Technologies and Applications*. IGI Global, 2005, s. 233–239.
- [6] R. Barker. *Case Method: Entity Relationship Modelling*. 1st. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1990. ISBN: 0201416964.
- [7] D. Berardi, D. Calvanese i G. D. Giacomo. „Reasoning on UML class diagrams”. W: *Artificial Intelligence* 168.1-2 (paź. 2005), s. 70–118.
- [8] J. Bézivin. „In search of a basic principle for Model Driven Engineering”. W: *Special Novatica Issue - UML and Model Engineering* (2004).
- [9] G. Bierman i A. Wren. „First-class relationships in an object-oriented language”. W: *ECOOP 2005-Object-Oriented Programming* 3586.642 (2005), s. 262–286. ISSN: 354027992X.
- [10] L. Bird, A. Goodchild i T. A. Halpin. „Object Role Modelling and XML-Schema”. W: *Conceptual Modeling - ER 2000, 19th International Conference on Conceptual Modeling, Salt Lake City, Utah, USA, October 9-12, 2000, Proceedings*. 2000, s. 309–322.
- [11] M. Blaha. *Patterns of Data Modeling (Emerging Directions in Database Systems and Applications)*. 1 wyd. 2010. ISBN: 9781439819890.
- [12] H. Boley, S. Tabet i G. Wagner. „Design Rationale of RuleML: A Markup Language for Semantic Web Rules”. W: *Proceedings of the First In-*

- ternational Conference on Semantic Web Working*. SWWS'01. California: CEUR-WS.org, 2001, s. 381–401.
- [13] M. Boyd i P. McBrien. „Comparing and Transforming Between Data Models Via an Intermediate Hypergraph Data Model”. W: *Journal on Data Semantics IV. Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2005, s. 69–109.
- [14] A. Bretto. „Hypergraph theory”. W: *An introduction. Mathematical Engineering. Cham: Springer* (2013).
- [15] A. Bundy i L. Wallen. „Partitioned Semantic Net”. W: *Catalogue of Artificial Intelligence Tools*. Springer Berlin Heidelberg, 1984, s. 89–89.
- [16] M. Bunge. *Treatise on basic philosophy: Ontology I: the furniture of the world*. T. 3. Springer Science & Business Media, 1977.
- [17] M. Bunge. *Treatise on basic philosophy: Volume 6: Epistemology & methodology II: Understanding the world*. T. 6. Springer Science & Business Media, 1983.
- [18] R. G. G. Cattell, R. Cattell, D. K. Barry, D. K. Barry, M. Berler, J. Eastman, D. Jordan, C. Russell, O. Schadow, T. Stanienda i in. *The object data standard: ODMG 3.0*. Morgan Kaufmann, 2000.
- [19] P. P.-S. Chen. „The entity-relationship model—toward a unified view of data”. W: *ACM Transactions on Database Systems* 1.1 (mar. 1976), s. 9–36. ISSN: 03625915.
- [20] V. Chernenkiy, Y. Gapanyuk, A. Nardid, M. Skvortsova, A. Gushcha, Y. Fedorenko i R. Picking. „Using the metagraph approach for addressing RDF knowledge representation limitations”. W: *2017 Internet Technologies and Applications (ITA)*. IEEE. 2017, s. 47–52.
- [21] R. J. Cloutier i D. Verma. „Applying the concept of patterns to systems architecture”. W: *Systems engineering* 10.2 (2007), s. 138–154.
- [22] E. F. Codd. „A relational model of data for large shared data banks”. W: *Communications of the ACM* 13.6 (1970), s. 377–387.
- [23] B. Combemale, R. France, J.-M. Jézéquel, B. Rumpe, J. Steel i D. Vojtisek. *Engineering modeling languages: Turning domain knowledge into tools*. CRC Press, 2016.
- [24] D. Cuadra, P. Martínez, E. Castro i H. Al-Jumaily. „Guidelines for representing complex cardinality constraints in binary and ternary relationships”. W: *Software and Systems Modeling* (2013). ISSN: 16191366.
- [25] D. Cuyler i T. Halpin. „Metamodels for object-role modeling.” W: 2003.

- [26] K. Czarnecki i S. Helsen. „Classification of model transformation approaches”. W: *Proceedings of the 2nd OOPSLA Workshop on Generative Techniques in the Context of the Model Driven Architecture*. T. 45. 3. USA. 2003, s. 1–17.
- [27] M. Dahchour i A. Pirotte. „The Semantics of Reifying n-ary Relationships as Classes.” W: *ICEIS*. T. 2. 2002, s. 580–586.
- [28] M. Dahchour, A. Pirotte i E. Zimányi. „Generic relationships in information modeling”. W: *Journal on Data Semantics IV*. Springer, 2005, s. 1–34.
- [29] P. Danenas, T. Skersys i R. Butleris. „Enhancing the extraction of SBVR business vocabularies and business rules from UML use case diagrams with natural language processing”. W: *Proceedings of the 23rd Pan-Hellenic Conference on Informatics*. 2019, s. 1–8.
- [30] G. Daniel, G. Sunyé i J. Cabot. „UMLtographDB: Mapping conceptual schemas to graph databases”. W: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2016. ISBN: 9783319463964.
- [31] I. Davies, P. Green, M. Rosemann, M. Indulska i S. Gallo. „How Do Practitioners Use Conceptual Modeling in Practice?” W: *Data Knowl. Eng.* 58.3 (wrz. 2006), s. 358–380. ISSN: 0169-023X.
- [32] A. De Lucia, C. Gravino, R. Oliveto i G. Tortora. „An experimental comparison of ER and UML class diagrams for data modeling”. W: *Empirical Software Engineering* (2010). ISSN: 13823256.
- [33] R. Elmasri i S. B. Navathe. „Fundamentals of Database Systems 4th edition”. W: *Database* (2003). ISSN: 19406029.
- [34] D. W. Embley i B. Thalheim, red. *Handbook of Conceptual Modeling*. Springer Berlin Heidelberg, 2011.
- [35] G. Engels, M. Gogolla, U. Hohenstein, K. Hülsmann, P. Löhr-Richter, G. Saake i H.-D. Ehrich. „Conceptual modelling of database applications using an extended ER model”. W: *Data & Knowledge Engineering* 9.2 (grud. 1992), s. 157–204.
- [36] H.-E. Eriksson i M. Penker. „Business Modeling With UML”. W: *Business Patterns at Work* (2000). ISSN: 00104531.
- [37] A. Evans, R. France, K. Lano i B. Rumpe. „Meta-Modelling Semantics of UML”. W: *Behavioral Specifications of Businesses and Systems*. Springer US, 1999, s. 45–60.

- [38] A. Evans, R. France, K. Lano i B. Rumpe. „The UML as a Formal Modeling Notation”. W: *The Unified Modeling Language*. «UML» '98: *Beyond the Notation*. Springer Berlin Heidelberg, 1999, s. 336–348.
- [39] C. Fahrner i G. Vossen. „A survey of database design transformations based on the Entity-Relationship model”. W: *Data and Knowledge Engineering* (1995).
- [40] S. Ferg. „Cardinality Concepts in Entity-Relationship Modeling”. W: *10th International Conference on the Entity-Relationship Approach*. Red. T. J. Teorey. San Mateo: ER Institute, 1991, s. 1–33.
- [41] R. D. N. Fidalgo, E. M. D. Souza, S. España, J. B. D. Castro i O. Pastor. „EERMM: A Metamodel for the Enhanced Entity-Relationship Model”. W: *Conceptual Modeling*. Springer Berlin Heidelberg, 2012, s. 515–524.
- [42] C. Finkelstein. „Information Engineering Methodology”. W: *Handbook on Architectures of Information Systems*. Springer Berlin Heidelberg, 1998, s. 405–427.
- [43] J. Fong, A. Fong, H. K. Wong i P. Yu. „Translating relational schema with constraints into XML Schema”. W: *International Journal of Software Engineering and Knowledge Engineering* 16.02 (kw. 2006), s. 201–243.
- [44] E. Gamma, R. Helm, R. Johnson i J. M. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. 1 wyd. Addison-Wesley Professional, 1994. ISBN: 0201633612.
- [45] A. Gangemi i V. Presutti. „Ontology design patterns”. W: *Handbook on ontologies*. Springer, 2009, s. 221–243.
- [46] G. Genova, J. Llorens i P. Martenez. „The meaning of multiplicity of n-ary associations in UML”. W: *Software and Systems Modeling* 1.2 (grud. 2002), s. 86–97.
- [47] M. Glinz. „Problems and deficiencies of UML as a requirements specification language”. W: *10th International Workshop on Software Specification and Design, IWSSD 2000*. 2000. ISBN: 0769508847.
- [48] H. Gomma. *Software modeling and design: UML, use cases, patterns, and software architectures*. Cambridge University Press, 2011.
- [49] C. Griffo, J. P. A. Almeida i G. Guizzardi. „A Pattern for the Representation of Legal Relations in a Legal Core Ontology.” W: *JURIX*. 2016, s. 191–194.
- [50] N. Guarino i G. Guizzardi. „“We need to discuss the Relationship”: Revisiting Relationships as Modeling Constructs”. W: *International Conference on Advanced Information Systems Engineering*. Springer. 2015, s. 279–294.

- [51] G. Guizzardi. „Ontological foundations for structural conceptual models”. Prac. dokt. 2005.
- [52] G. Guizzardi, R. Falbo i R. Guizzardi. „Grounding Software Domain Ontologies in the Unified Foundational Ontology (UFO): The case of the ODE Software Process Ontology.” W: sty. 2008, s. 127–140.
- [53] G. Guizzardi i G. Wagner. „Towards ontological foundations for agent modelling concepts using the unified foundational ontology (UFO)”. W: *International Bi-Conference Workshop on Agent-Oriented Information Systems*. Springer. 2004, s. 110–124.
- [54] J. Guttag. *Introduction to computation and programming using Python*. Cambridge, MA: The MIT Press, 2013. ISBN: 9780262519632.
- [55] T. Halpin. „Entity Relationship modeling from an ORM perspective: Part 1”. W: *Journal of Conceptual Modeling* 13 (2000).
- [56] T. Halpin. „Entity Relationship modeling from an ORM perspective: Part 2”. W: (2004).
- [57] T. Halpin. *Information modeling and relational databases*. Burlington, MA: Morgan Kaufmann Publishers, 2008. ISBN: 9780080568737.
- [58] T. Halpin. „Metaschemas for ER, ORM and UML Data Models”. W: *Journal of Database Management* 13.2 (kw. 2002), s. 20–30.
- [59] T. Halpin. „Object-Role Modeling”. W: *Encyclopedia of Database Systems*. Springer New York, 2016, s. 1–7.
- [60] T. Halpin. *Object-Role Modeling Fundamentals: A practical guide to data modeling with ORM*. Technics Publications, 2015.
- [61] T. Halpin. „ORM 2”. W: *On the Move to Meaningful Internet Systems 2005: OTM 2005 Workshops*. Red. R. Meersman, Z. Tari i P. Herrero. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, s. 676–687. ISBN: 978-3-540-32132-3.
- [62] T. Halpin i A. Bloesch. „Modeling Collections in UML and ORM”. W: *Proc. EMMSAD00: 5th IFIP WG8 1* (2000).
- [63] T. Halpin i T. Morgan. *Information modeling and relational databases*. Morgan Kaufmann, 2010.
- [64] T. Halpin i H. A. Proper. „Database schema transformation and optimization”. W: *Proceedings of the OOER'95, 14th International Object-Oriented and Entity-Relationship Modelling Conference, Gold Coast, Queensland, Australia*. 1995. ISBN: 3540606726.
- [65] T. Halpin i H. A. Proper. „Subtyping and polymorphism in object-role modelling”. W: *Data and Knowledge Engineering* (1995). ISSN: 0169023X.



- [66] B. Henderson-Sellers. *On the Mathematics of Modelling, Metamodeling, Ontologies and Modelling Languages*. Springer Berlin Heidelberg, 2012.
- [67] G. G. Hendrix. „Encoding knowledge in partitioned networks”. W: *Associative networks*. Elsevier, 1979, s. 51–92.
- [68] P. Hitzler, A. Gangemi i K. Janowicz. *Ontology engineering with ontology design patterns: foundations and applications*. T. 25. IOS Press, 2016.
- [69] B. Hnatkowska, Z. Huzar i L. Tuzinkiewicz. „Data modeling with UML 2.0”. W: *Proceedings of the 2005 conference on Software Engineering: Evolution and Emerging Technologies*. 2005, s. 63–74.
- [70] C. Ireland, D. Bowers, M. Newton i K. Waugh. „A Classification of Object-Relational Impedance Mismatch”. W: *2009 First International Conference on Advances in Databases, Knowledge, and Data Applications*. IEEE, 2009.
- [71] C. Ireland, D. Bowers, M. Newton i K. Waugh. „Understanding object-relational mapping: A framework based approach”. W: *Int J Adv Softw* 2 (sty. 2009).
- [72] M. Jodłowiec. „Modeling with association inheritance – a case study on Extended Semantic Network Module of Semantic Knowledge Base”. W: *Zeszyty Naukowe. Elektryka / Politechnika Opolska* z. 75 (2017), s. 27–30. ISSN: 1429-1533.
- [73] M. Jodłowiec. „Complex Relationships Modeling in Association-Oriented Database Metamodel”. W: *Intelligent Information and Database Systems - 10th Asian Conference, ACIIDS 2018, Dong Hoi City, Vietnam, March 19-21, 2018, Proceedings, Part II*. Red. N. T. Nguyen, D. H. Hoang, T.-P. Hong, H. Pham i B. Trawinski. T. 10752. Lecture Notes in Computer Science. Springer, 2018, s. 46–56.
- [74] M. Jodłowiec. „Design Pattern for Modeling Complex Entities in Association-Oriented Database Metamodel”. W: *Zeszyt poświęcony studiom doktoranckim na Wydziale Elektrotechniki, Automatyki i Informatyki*. Opole, 2016.
- [75] M. Jodłowiec. *Ekstrakcja semantyki modelu ESNM<sup>SKB</sup>*. Spraw. tech. Wrocław University of Science i Technology, 2021.
- [76] M. Jodłowiec i M. Krótkiewicz. „An Approach to Expressing Metamodels’ Semantics in a Concept System”. W: *Advances and Trends in Artificial Intelligence. Artificial Intelligence Practices - 34th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEAAIE 2021, Kuala Lumpur, Malaysia, July 26-29, 2021, Proceedings*,

- Part I*. Red. H. Fujita, A. Selamat, J. C.-W. Lin i M. Ali. T. 12798. Lecture Notes in Computer Science. Springer, 2021, s. 274–282.
- [77] M. Jodłowiec, M. Krótkiewicz i K. Wojtkiewicz. „Defining semantic networks using association-oriented metamodel”. W: *Journal of Intelligent & Fuzzy Systems* 37.6 (grud. 2019), s. 7453–7464. ISSN: 10641246, 18758967.
- [78] M. Jodłowiec, M. Krótkiewicz i K. Wojtkiewicz. „Semantic Networks Modeling with Operand-Operator Structures in Association-Oriented Metamodel”. W: *Computational Collective Intelligence - 9th International Conference, ICCCI 2017, Nicosia, Cyprus, September 27-29, 2017, Proceedings, Part I*. Red. N. T. Nguyen, G. A. Papadopoulos, P. Jędrzejowicz, B. Trawinski i G. Vossen. T. 10448. Lecture Notes in Computer Science. Springer, 2017, s. 24–33.
- [79] M. Jodłowiec, M. Krótkiewicz i P. Zabawa. „Fundamentals of generalized and extended graph-based structural modeling”. W: *ICCCI 2020*. Springer. 2020.
- [80] P. Johannesson. „A method for transforming relational schemas into conceptual schemas”. W: *Proceedings of 1994 IEEE 10th International Conference on Data Engineering*. IEEE. 1994, s. 190–201.
- [81] J. Johnson. „Hypernetworks: Multidimensional relationships in multilevel systems”. W: *The European Physical Journal Special Topics* 225.6-7 (wrz. 2016), s. 1037–1052.
- [82] C. Joslyn i K. Nowak. „Ubergraphs: A Definition of a Recursive Hypergraph Structure”. W: *arXiv preprint arXiv:1704.05547* (2017).
- [83] D. Karagiannis i H. Kühn. „Metamodelling platforms”. W: *EC-Web*. T. 2455. 2002, s. 182.
- [84] J. Karpovič, G. Krikščiūnienė, L. Ablonskis i L. Nemuraitė. „The Comprehensive Mapping of Semantics of Business Vocabulary and Business Rules (SBVR) to OWL 2 Ontologies”. W: *Information Technology And Control* 43.3 (wrz. 2014).
- [85] C. M. Keet i P. R. Fillottrani. „An ontology-driven unifying metamodel of UML Class Diagrams, EER, and ORM2”. W: *Data & Knowledge Engineering* 98 (lip. 2015), s. 30–53.
- [86] A. W. Kiwelekar i R. K. Joshi. „An Object-Oriented Metamodel for Bunge-Wand-Weber Ontology”. W: *Workshop on Semantic Web for Collaborative Knowledge Acquisition at IJCAI*. arXiv: 1004.3640. 2010.
- [87] M. Krótkiewicz. „A novel inheritance mechanism for modeling knowledge representation systems”. W: *Comput. Sci. Inf. Syst.* 15.1 (2018), s. 51–78.

- [88] M. Krótkiewicz. *Asocjacyjny metamodel baz danych. Definicja formalna oraz analiza porównawcza metamodeli baz danych (eng. Association-Oriented Database Metamodel)*. Studia i Monografie z. 444. Opole: Oficyna Wydawnicza Politechniki Opolskiej, 2016, s. 269. ISBN: 978-83-65235-50-3.
- [89] M. Krótkiewicz. „Association-Oriented Database Model -  $n$ -ary Associations”. W: *International Journal of Software Engineering and Knowledge Engineering* 27.2 (2017), s. 281–320.
- [90] M. Krótkiewicz. „Cyclic value ranges model for specifying flowing resources in unified process metamodel”. W: *Enterprise Information Systems* 13.7-8 (2019), s. 1046–1068.
- [91] M. Krótkiewicz. „Formal Definition and Modelling Language of Association-Oriented Database Metamodel (AssoBase)”. W: *Vietnam Journal of Computer Science* (list. 2018).
- [92] M. Krótkiewicz. „Hypergraph Approach Towards Ontology Design in Association-Oriented Metamodel”. W: *Cybernetics and Systems* 50.2 (2019), s. 132–153.
- [93] M. Krótkiewicz, M. Jodłowiec i K. Wojtkiewicz. „Introduction to Semantic Knowledge Base: Multilanguage Support of Linguistic Module”. W: *Third European Network Intelligence Conference, ENIC 2016, Wrocław, Poland, September 5-7, 2016*. IEEE Computer Society, 2016, s. 188–194.
- [94] M. Krótkiewicz i K. Wojtkiewicz. „Functional and structural integration without competence overstepping in structured semantic knowledge base system”. W: *Journal of Logic, Language and Information* 23.3 (2014), s. 331–345.
- [95] M. Krótkiewicz, K. Wojtkiewicz, M. Jodłowiec i W. Pokuta. „Semantic Knowledge Base: Quantifiers and Multiplicity in Extended Semantic Networks Module”. W: *Knowledge Engineering and Semantic Web - 7th International Conference, KESW 2016, Prague, Czech Republic, September 21-23, 2016, Proceedings*. Red. A.-C. N. Ngomo i P. Kremen. T. 649. Communications in Computer and Information Science. Springer, 2016, s. 173–187.
- [96] T. Levendovszky, G. Karsai, M. Maroti, A. Ledeczi i H. Charaf. „Model reuse with metamodel-based transformations”. W: *International Conference on Software Reuse*. Springer. 2002, s. 166–178.
- [97] X. Li. „Using UML for conceptual modeling: towards an ontological core”. Includes bibliographical references (leaves 139-147). 2007.

- [98] X. Li i J. Parsons. „Ontological Semantics for the Use of UML in Conceptual Modeling.” W: *ER (Tutorials, Posters, Panels & Industrial Contributions)*. 2007, s. 179–184.
- [99] M. H. Linehan. „SBVR use cases”. W: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. T. 5321 LNCS. 2008, s. 182–196. ISBN: 3540888071.
- [100] B. H. Liskov i J. M. Wing. „A behavioral notion of subtyping”. W: *ACM Transactions on Programming Languages and Systems (TOPLAS)* 16.6 (1994), s. 1811–1841.
- [101] C.-M. Lo i H.-Y. Hung. „Towards a UML profile to relational database modeling”. W: *Applied Mathematics & Information Sciences* 8.2 (2014), s. 733.
- [102] D. Martinez-Mosquera, S. Luján-Mora i H. Recalde. „Conceptual modeling of big data extract processes with UML”. W: *2017 International Conference on Information Systems and Computer Science (INCISCOS)*. IEEE. 2017, s. 207–211.
- [103] H. C. Mayr i B. Thalheim. „The triptych of conceptual modeling”. W: *Software and Systems Modeling* 20.1 (list. 2020), s. 7–24.
- [104] D. L. McGuinness, F. Van Harmelen i in. „OWL web ontology language overview”. W: *W3C recommendation* 10.10 (2004), s. 2004.
- [105] S. T. McQuade, N. J. Merrill i B. Piccoli. „Metabolic graphs, LIFE method and the modeling of drug action on Mycobacterium tuberculosis”. W: *arXiv preprint arXiv:2003.12400* (2020).
- [106] E. Mickeviciute, R. Butleris, S. Gudas i E. Karciauskas. „Transforming BPMN 2.0 Business Process Model into SBVR Business Vocabulary and Rules”. W: (2017), s. 360–371.
- [107] „Modeling Autoreferential Relationships in Association-Oriented Database Metamodel”. W: t. 656. 2018, s. 49–62. ISBN: 9783319672281.
- [108] E. J. Naiburg i R. A. Maksimchuk. „UML for Database Design”. W: *Addison-Wesley* 27.1 (2003), s. 66–67.
- [109] S. Nelson, D. J. Pearce i J. Noble. „First Class Relationships for OO Languages”. W: *Proceedings of the 6th International Workshop on Multiparadigm Programming with Object-Oriented Languages (MPOOL 2008)*. 2008, s. 33.
- [110] L. Nemuraite i T. Skersys. „VETIS tool for editing and transforming SBVR business vocabularies and business rules into UML&OCL models”. W: *16th International Conference on Information and Software Technologies, Kaunas: Kaunas University of Technology* (2010), s. 377–384.

- [111] P. B. F. Njonko i W. El Abed. „From natural language business requirements to executable models via SBVR”. W: *2012 International Conference on Systems and Informatics, ICSAI 2012* Icsai (2012), s. 2453–2457.
- [112] C. K. Ogden i I. A. Richards. *The Meaning of Meaning: A Study of the Influence of Language upon Thought and of the Science of Symbolism*. T. 29. K. Paul, Trench, Trubner & Company, Limited, 1923. URL: <https://archive.org/details/TheMeaningOfMeaningC.K.OgdenAndI.A.Richards/>.
- [113] OMG. *Object Management Group, Semantics Of Business Vocabulary And Rules 1.5*. Object Management Group, 2019. URL: <http://www.omg.org/spec/SBVR/1.5/>.
- [114] OMG. *Object Management Group, Unified Modeling Language (UML) superstructure version 2.5*. Object Management Group, 2015. URL: <http://www.omg.org/spec/UML/2.5/>.
- [115] J. Osis i U. Donins. „Formalization of the UML Class Diagrams”. W: *Communications in Computer and Information Science*. Springer Berlin Heidelberg, 2010, s. 180–192.
- [116] P. Pal i N. Vyas. „Data Driven Decision Making for Patient-Oriented Systems Using Knowledge Engineering”. W: *International Journal of Scientific and Engineering Research* (wrz. 2013).
- [117] W.-l. Pan i D.-x. Liu. „Abstract syntax of object role modeling”. W: *Fourth International Conference on Machine Vision (ICMV 2011): Computer Vision and Image Analysis: Pattern Recognition and Basic Technologies*. Red. Z. Zeng i Y. Li. SPIE, grud. 2011.
- [118] W.-l. Pan i D.-x. Liu. „ORM-ODM: Ontology Definition Metamodel for Object Role Modeling”. W: *2010 3rd International Conference on Computer Science and Information Technology*. IEEE, lip. 2010.
- [119] P. Papotti i R. Torlone. „Automatic techniques for data model translation”. W: *Proceedings of VLDB 2005 PhD Workshop*. Citeseer. 2005, s. 21.
- [120] S. Patig. „Evolution of entity–relationship modelling”. W: *Data & Knowledge Engineering* 56.2 (lut. 2006), s. 122–138.
- [121] A. Pereira Toledo, A. Rodriguez Morffi, A. Pérez Alonso, A. Morfa Hernández i L. M. Gonzalez Gonzalez. „A Method for Expressing Integrity Constraints in Database Conceptual Modeling”. W: *Computación y Sistemas* 1.1 (2020).
- [122] B. Qu, P. Kumar, E. Zhang, P. Jaiswal, L. Cooper, J. Elser i Y. Zhang. „Interactive design and visualization of N-ary relationships”. W: *SIGGRAPH Asia 2017 Symposium on Visualization*. ACM, list. 2017.

- [123] C. Qu, M. Tao i R. Yuan. „A hypergraph-based blockchain model and application in Internet of Things-enabled smart homes”. W: *Sensors* 18.9 (2018), s. 2784.
- [124] A. Raj, T. V. Prabhakar i S. Hendryx. „Transformation of SBVR Business Design to UML Models”. W: *Business* (2008), s. 29–38.
- [125] E. Reynares, M. L. Caliusco i M. R. Galli. „SBVR to OWL 2 Mappings: An Automatable and Structural-Rooted Approach”. W: *CLEI Electronic Journal* 17.3 (grud. 2014).
- [126] A. Rodrigues da Silva. „Model-driven engineering: A survey supported by the unified conceptual model”. W: *Computer Languages, Systems & Structures* 43 (paź. 2015), s. 139–155.
- [127] J. Rouces, G. de Melo i K. Hose. „Framebase: Representing n-ary relations using semantic frames”. W: *European Semantic Web Conference*. Springer. 2015, s. 505–521.
- [128] S. Si-Said Cherfi, J. Akoka i I. Comyn-Wattiau. „Measuring UML conceptual modeling quality: method and implementation”. W: *BDA 2002 : 18èmes Journées Bases de Données Avancées*. Evry, France, paź. 2002. URL: <https://hal.archives-ouvertes.fr/hal-01124871>.
- [129] E. Seidewitz. „What models mean”. W: *IEEE Software* 20.5 (wrz. 2003), s. 26–32.
- [130] M. Sergievskiy. „N-ary Relations of Association in Class Diagrams: Design Patterns”. W: *International Journal of Advanced Computer Science and Applications* 7.2 (2016).
- [131] P. Severi, J. Fiadeiro i D. Ekserdjian. „Guiding the representation of n-ary relations in ontologies through aggregation, generalisation and participation”. W: *Web Semantics: Science, Services and Agents on the World Wide Web* 9.2 (2011), s. 83–98.
- [132] A. J. Simons, I. Graham i C. M. Bank. „37 things that don’t work in object-oriented modelling with UML”. W: *Proc. 2nd ECOOP Workshop on Precise Behavioural Semantics*. Citeseer. 1998, s. 209–232.
- [133] T. Skersys, P. Danenas i R. Butleris. „Extracting SBVR business vocabularies and business rules from UML use case diagrams”. W: *Journal of Systems and Software* 141 (2018), s. 111–130.
- [134] T. Skersys, L. Tutkute i R. Butleris. „The enrichment of BPMN business process model with SBVR business vocabulary and rules”. W: *Journal of Computing and Information Technology* 20.3 (2012), s. 143–150. ISSN: 13301136.

- [135] I.-Y. Song, M. Evans i E. K. Park. „A comparative analysis of entity-relationship diagrams”. W: *Journal of Computer and Software Engineering* 3.4 (1995), s. 427–459.
- [136] G. Sparks. „Database modelling in UML”. W: *Methods & Tools* 9.1 (2001), s. 10–23.
- [137] E. Stathopoulou i P. Vassiliadis. „Design Patterns for Relational Databases”. W: 2009.
- [138] V. C. Storey, J. C. Trujillo i S. W. Liddle. „Research on conceptual modeling: Themes, topics, and introduction to the special issue”. W: *Data & Knowledge Engineering* 98 (lip. 2015), s. 1–7.
- [139] A. Szalas. „Revisiting Object-Rule Fusion in Query Languages”. W: *Procedia Computer Science* 176 (2020), s. 50–59.
- [140] T. J. Teorey, D. Yang i J. P. Fry. „A logical design methodology for relational databases using the extended entity-relationship model”. W: *ACM Computing Surveys* (1986). ISSN: 03600300. arXiv: arXiv:1011.1669v3.
- [141] B. Thalheim. *Entity-Relationship Modeling: Foundations of Database Technology*. 1 wyd. Springer-Verlag Berlin Heidelberg, 2000. ISBN: 978-3-642-08480-5.
- [142] B. Thalheim. „Fundamentals of cardinality constraints”. W: *Entity-Relationship Approach – ER – 92*. Springer Berlin Heidelberg, 1992, s. 7–23.
- [143] B. Thalheim. „Semiotics in Databases”. W: *Model and Data Engineering*. Springer International Publishing, 2019, s. 3–19.
- [144] B. Thalheim. „The enhanced entity-relationship model”. W: *Handbook of Conceptual Modeling*. Springer, 2011, s. 165–206.
- [145] S. Ullmann. *Semantics: an introduction to the science of meaning*. Barnes & Noble, 1962. URL: <https://archive.org/details/semanticsintrodu0000ullm/>.
- [146] H. Vera, W. Boaventura, M. Holanda, V. Guimaraes i F. Hondo. „Data modeling for NoSQL document-oriented databases”. W: *CEUR Workshop Proceedings*. T. 1478. 2015, s. 129–135.
- [147] E. van der Vlist. *XML Schema: The W3C’s Object-Oriented Descriptions for XML*. Beijing: O’Reilly, 2002. ISBN: 978-0-596-00252-7.
- [148] V. I. Voloshin. *Introduction to graph and hypergraph theory*. Nova Science Publ., 2009.
- [149] Y. Wand i R. Weber. „An ontological model of an information system”. W: *IEEE transactions on software engineering* 16.11 (1990), s. 1282–1292.

- [150] K. Wojtkiewicz. „Zunifikowany Metamodel Procesów”. Prac. dokt. Opole University of Technology, 2019, s. 1–172.
- [151] Z. Wu, J. Liao, W. Song, H. Mao, Z. Huang, X. Li i H. Mao. „Semantic hyper-graph-based knowledge representation architecture for complex product development”. W: *Computers in Industry* 100 (2018), s. 43–56.
- [152] S. M. Yacoub i H. H. Ammar. *Pattern-oriented analysis and design : composing patterns to design software systems*. Boston, MA: Addison-Wesley, 2004. ISBN: 9780201776409.



# Spis tablic

2.1.	Przyporządkowanie rzeczy do poszczególnych kategorii w SBVR. Przykłady w dziedzinie modelowania. . . . .	28
2.2.	Podsumowanie istotnych własności wybranych metamodeli danych . . . . .	41
3.1.	Tabela obrazująca zależności między pojęciami abstrakcyjnymi, konkretnymi i ich ekstensjami . . . . .	49
7.1.	Wyniki ewaluacji miary $TSA_p$ . . . . .	229
7.2.	Wyniki ewaluacji miary $ASA_p$ . . . . .	230
7.3.	Wyniki ewaluacji miary $FSA_p$ . . . . .	231
7.4.	Wyniki ewaluacji miary $LSA_p$ . . . . .	232
7.5.	Wyniki ewaluacji miary $TC_p$ . . . . .	233
7.6.	Wyniki ewaluacji miary $AC_p$ . . . . .	234
7.7.	Wyniki ewaluacji miary $FC_p$ . . . . .	235
7.8.	Wyniki ewaluacji miary $LC_p$ . . . . .	236
7.9.	Wyniki ewaluacji dla miar zagregowanych . . . . .	237
7.10.	Wyniki ewaluacji dla miar wynikowych . . . . .	239
8.1.	Podsumowanie ekstrakcji semantyki modelu $ESNM^{SKB}$ . . . . .	248
8.2.	Wyniki ewaluacji translacji modelu $ESNM^{SKB}$ miarami pierwszego poziomu . . . . .	269
8.3.	Wyniki ewaluacji translacji modelu $ESNM^{SKB}$ miarami wynikowymi . . . . .	271

## Spis rysunków

2.1.	Warstwy modelowania wraz z przykładem ich zastosowania dla AOM. . . . .	18
2.2.	Komponenty metod modelowania. Opracowano na podstawie [83] . . . . .	19
2.3.	Zobrazowanie różnic pomiędzy <i>look-here semantics</i> oraz <i>look-across semantics</i> . W zależności od przyjętej semantyki, oba diagramy mogą reprezentować dwa zdania: „Jedna klasa klasyfikuje wiele obiektów. Jeden obiekt jest klasyfikowany przez jedną klasę.” Opracowano na podstawie [13] oraz [135] . .	21
2.4.	Metastruktura diagramu klas UML 2.5.1. Źródło: [114] . . . . .	24
2.5.	Trójkąt semiotyczno-semantyczny. Opracowano na podstawie [145] oraz [113]	27
2.6.	Składnia abstrakcyjna AOM wyrażona na diagramie klas UML. Źródło: [88] .	30
2.7.	Przykładowy diagram intensjonalny AML. Źródło: [88] . . . . .	31
2.8.	Struktura modelu realizująca wymagania 1-7 wyrażona w metamodelu EER.	37
2.9.	Struktura modelu realizująca wymagania 1-7 wyrażona w metamodelu ORM.	38
2.10.	Struktura modelu realizująca wymagania 1-7 wyrażona w metamodelu UML.	38
2.11.	Struktura modelu realizująca wymagania 1-7 wyrażona w metamodelu AOM.	40
3.1.	Diagram pojęć obrazujący podstawowe pojęcia modelowania . . . . .	45
3.2.	Diagram pojęć przedstawiający powiązania między rdzennymi pojęciami modelowania . . . . .	50
3.3.	Diagram pojęć przedstawiający powiązania między pojęciami modelowania związanych z modelowaniem klasyfikatorów . . . . .	54
3.4.	Diagram pojęć przedstawiający powiązania między pojęciami modelowania związanymi z modelowaniem cech . . . . .	55
3.5.	Diagram pojęć przedstawiający powiązania między pojęciami modelowania związanymi z modelowaniem zbiorowości . . . . .	56
3.6.	Diagram pojęć przedstawiający powiązania między pojęciami modelowania związanymi z modelowaniem bytów . . . . .	58
3.7.	Diagram pojęć przedstawiający powiązania między pojęciami modelowania związanych z modelowaniem kategoryzacji . . . . .	61
3.8.	Diagram pojęć przedstawiający powiązania między pojęciami modelowania związanych z modelowaniem asocjacji . . . . .	65
3.9.	Pojęcia związane z modelowaniem wzorców . . . . .	70
4.1.	Diagram pojęć prezentujący konieczne atomy semantyczne koncepcji wzorca lista . . . . .	76

4.2. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne . . . . .	78
4.3. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne oraz własność możliwą: unikalność listy . . . . .	79
4.4. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne oraz własność możliwą: przypisanie do listy . . . . .	79
4.5. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne oraz własność możliwą: zapełnienie listy . . . . .	80
4.6. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne oraz negację własności możliwej: binawigowalność . . . . .	81
4.7. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne oraz własność możliwą: zależność istnienia elementu od grupy . . . . .	81
4.8. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne oraz własność możliwą: zależność istnienia grupy od elementu . . . . .	82
4.9. Diagram modelu asocjacyjnego wzorca lista implementującego konieczne atomy semantyczne oraz własność możliwą: $n$ -ograniczona przynależność do listy dla $n = 1$ . . . . .	83
4.10. Przykład wykorzystania wzorca modelowania listy . . . . .	84
4.11. Diagram pojęć prezentujący konieczne atomy semantyczne koncepcji wzorca słownik . . . . .	85
4.12. Diagram modelu asocjacyjnego wzorca słownik implementującego konieczne atomy semantyczne . . . . .	87
4.13. Diagram AML wzorca słownik implementującego konieczne atomy semantyczne wraz z zanegowaniem własności możliwej binawigowalności słownika . . . . .	88
4.14. Diagram AML wzorca słownik implementującego konieczne atomy semantyczne oraz własność możliwą: $n$ -ograniczone użycia dla $n = 10$ . . . . .	89
4.15. Diagram modelu asocjacyjnego wzorca słownik implementującego konieczne atomy semantyczne oraz własność możliwą: $n$ -minimalna opisywalność dla $n = 1$ . . . . .	89
4.16. Diagram modelu asocjacyjnego wzorca słownik implementującego konieczne atomy semantyczne oraz własność możliwą: obligatoryjność opisywania . . . . .	90
4.17. Diagram modelu asocjacyjnego wzorca słownik implementującego konieczne atomy semantyczne oraz własność możliwą: zależność istnienia elementu opisywanego od opisyującego . . . . .	91
4.18. Przykład zastosowania wzorca modelowania słownik . . . . .	92
4.19. Diagram pojęć prezentujący konieczne atomy semantyczne koncepcji wzorca autoreferencja . . . . .	94

4.20. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne . . . . .	95
4.21. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne oraz własność możliwą $n$ -arność autoreferencji dla $n = 4$ . . . . .	97
4.22. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne oraz zanegowaną własność możliwą binawigowalność roli autoreferencji dla $r = \diamond ZwiązekAutoreferencji \xrightarrow{+Cel}$ $\square ElementAutoreferencyjny$ . . . . .	98
4.23. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne oraz własność możliwą $n$ -minimalność roli autoreferencji dla $n = 5$ oraz $r = \diamond ZwiązekAutoreferencji \xrightarrow{+Cel}$ $\square ElementAutoreferencyjny$ . . . . .	98
4.24. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne oraz własność możliwą $n$ -maksymalność roli autoreferencji dla $n = 1$ oraz $r = \diamond ZwiązekAutoreferencji \xrightarrow{+Zrodlo}$ $\square ElementAutoreferencyjny$ . . . . .	99
4.25. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne oraz własność możliwą $n$ -powtarzalność roli autoreferencji dla $n = 1$ oraz $r = \diamond ZwiązekAutoreferencji \xrightarrow{\{1\}+Zrodlo}$ $\square ElementAutoreferencyjny$ . . . . .	100
4.26. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne oraz własność możliwą zależność istnienia elementu od autoreferencji dla $r = \diamond ZwiązekAutoreferencji \xrightarrow{+Zrodlo}$ $\square ElementAutoreferencyjny$ . . . . .	100
4.27. Diagram modelu asocjacyjnego wzorca autoreferencji implementującego konieczne atomy semantyczne oraz własność możliwą zależność istnienia autoreferencji od elementu dla $r = \diamond ZwiązekAutoreferencji \xrightarrow{+Zrodlo}$ $\square ElementAutoreferencyjny$ . . . . .	101
4.28. Diagram modelu asocjacyjnego wzorca autoreferencji opartego o autoreferencyjną rolę . . . . .	102
4.29. Przykład zastosowania wzorca autoreferencji . . . . .	104
4.30. Diagram pojęć prezentujący konieczne atomy semantyczne koncepcji wzorca pęku ról . . . . .	106
4.31. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne . . . . .	108
4.32. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz własność możliwą: opisanie wiązki . . . . .	109

4.33. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz własność możliwą: $m$ -powtarzalność elementu pęku dla $m = 3$ oraz $p = \diamond Wiazka \xrightarrow{\{3\}+ElementPeku_1} \square ElementZwiazany_1$ . . . . .	110
4.34. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz własność możliwą: $m$ -minimalność elementu pęku dla $m = 3$ oraz $p = \diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$ . . . . .	110
4.35. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz własność możliwą: $m$ -maksymalność elementu pęku dla $m = 10$ oraz $p = \diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$ . . . . .	111
4.36. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz własność możliwą: $m$ -ograniczoną przynależność do wiązki elementu pęku dla $m = 1$ oraz $p = \diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$ . . . . .	112
4.37. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz własność możliwą: zależność istnienia wiązki od elementu pęku dla $p = \diamond Wiazka \xrightarrow{+ElementPeku_1} \blacktriangleright \square ElementZwiazany_1$ . . . . .	112
4.38. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz własność możliwą: zależność istnienia od wiązki elementu pęku dla $p = \diamond Wiazka \xrightarrow{+ElementPeku_1} \blacktriangleleft \square ElementZwiazany_1$ . . . . .	113
4.39. Diagram modelu asocjacyjnego wzorca pęku ról implementującego konieczne atomy semantyczne oraz zanegowaną własność możliwą binawigowalność elementu pęku dla roli $\diamond Wiazka \xrightarrow{+ElementPeku_1} \square ElementZwiazany_1$ . . . . .	114
4.40. Przykład użycia wzorca pęku ról . . . . .	116
4.41. Diagram pojęć prezentujący konieczne atomy semantyczne koncepcji wzorca sieci uogólnionej . . . . .	118
4.42. Diagram modelu asocjacyjnego wzorca sieci uogólnionej implementującego konieczne atomy semantyczne – wersja uwzględniająca stałą, maksymalną arność powiązań . . . . .	120
4.43. Diagram modelu asocjacyjnego wzorca sieci uogólnionej implementującego konieczne atomy semantyczne – wersja uwzględniająca dowolną arność powiązań	121
4.44. Diagram modelu asocjacyjnego wzorca sieci uogólnionej implementującego konieczne atomy semantyczne oraz własność możliwą: binarność powiązań sieci uogólnionej . . . . .	123
4.45. Diagram modelu asocjacyjnego wzorca sieci uogólnionej implementującego konieczne atomy semantyczne oraz własność możliwą: węzłowe ograniczenie złącza . . . . .	124
4.46. Diagram modelu asocjacyjnego wzorca sieci uogólnionej implementującego konieczne atomy semantyczne oraz własność możliwą: etykietowalność węzłów sieci uogólnionej . . . . .	125

4.47. Diagram modelu asocjacyjnego wzorca sieci uogólnionej implementującego konieczne atomy semantyczne oraz własność możliwą: etykietowalność krawędzi sieci uogólnionej . . . . .	126
4.48. Diagram modelu asocjacyjnego wzorca sieci uogólnionej implementującego konieczne atomy semantyczne oraz własność możliwą: pierwszoklasowość sieci uogólnionej . . . . .	128
4.49. Przykład zastosowania wzorca modelowania sieci uogólnionej . . . . .	129
4.50. Diagram pojęć prezentujący konieczne atomy semantyczne wzorca BACT . . . . .	131
4.51. Diagram AML reprezentujący implementację wzorca BACT . . . . .	132
4.52. Przykład zastosowania wzorca BACT . . . . .	133
4.53. Diagram pojęć prezentujący konieczne atomy semantyczne koncepcji wzorca abstrakcyjnego celu roli . . . . .	134
4.54. Diagram AML modelu implementującego konieczne atomy semantyczne wzorca abstrakcyjnego celu roli. Uczestnicy są asocjacjami. . . . .	135
4.55. Diagram AML modelu implementującego konieczne atomy semantyczne wzorca abstrakcyjnego celu roli. Uczestnicy są kolekcjami. . . . .	136
4.56. Przykład wykorzystania wzorca modelowania Abstrakcyjnego Celu Roli . . . . .	138
4.57. Diagram pojęć prezentujący konieczne atomy semantyczne koncepcji wzorca abstrakcyjnego właściciela roli . . . . .	139
4.58. Diagram AML modelu implementującego konieczne atomy semantyczne wzorca abstrakcyjnego właściciela roli. . . . .	140
4.59. Diagram AML modelu implementującego konieczne atomy semantyczne wzorca abstrakcyjnego właściciela roli. Na diagramie pokazano odziedziczone role. . . . .	141
4.60. Przykład wykorzystania wzorca modelowania Abstrakcyjnego Właściciela Roli	143
4.61. Diagram pojęć przedstawiający strukturę pojęciową wzorca kaskady asocjacji	144
4.62. Diagram AML reprezentujący implementację wzorca kaskady asocjacji . . . . .	146
4.63. Diagram AML obrazujący przykład zastosowania wzorca kaskady asocjacji . . . . .	147
5.1. Schematyczna transformacja artefaktów translacyjnych . . . . .	152
5.2. Diagram pojęć przedstawiający uproszczoną konceptualizacją metamodelu asocjacyjnego . . . . .	160
5.3. Diagram obrazujący odwzorowanie $\begin{matrix} *_{AOM}^{[kolekcja]} \\ \mapsto \\ *_{EER_{AOM}}^{[kolekcja]} \end{matrix}$ . . . . .	174
5.4. Diagram obrazujący odwzorowanie $\begin{matrix} *_{AOM}^{[atrybut]} \\ \mapsto \\ *_{EER_{AOM}}^{[atrybut]} \end{matrix}$ . . . . .	176
5.5. Diagram obrazujący odwzorowanie $\begin{matrix} *_{AOM}^{[asocjacja]} \\ \mapsto \\ *_{EER_{AOM}}^{[asocjacja]} \end{matrix}$ . . . . .	177
5.6. Diagram obrazujący odwzorowanie $\begin{matrix} *_{AOM}^{[rola]} \\ \mapsto \\ *_{EER_{AOM}}^{[rola]} \end{matrix}$ . . . . .	179
5.7. Diagram obrazujący odwzorowanie $\begin{matrix} *_{AOM}^{[opis asocjacji]} \\ \mapsto \\ *_{EER_{AOM}}^{[opis asocjacji]} \end{matrix}$ . . . . .	180
5.8. Diagram obrazujący odwzorowanie $\begin{matrix} *_{AOM}^{[dziedziczenie]} \\ \mapsto \\ *_{EER_{AOM}}^{[dziedziczenie]} \end{matrix}$ . . . . .	183
5.9. Diagram obrazujący odwzorowanie $\begin{matrix} *_{AOM}^{[kolekcja]} \\ \mapsto \\ *_{ORM_{AOM}}^{[kolekcja]} \end{matrix}$ . . . . .	189
5.10. Diagram obrazujący odwzorowanie $\begin{matrix} *_{AOM}^{[atrybut]} \\ \mapsto \\ *_{ORM_{AOM}}^{[atrybut]} \end{matrix}$ . . . . .	191

5.11. Diagram obrazujący odwzorowanie	$\ast_{AOM}^{[asocjacja]} \mapsto \ast_{ORM_{AOM}}^{[asocjacja]}$	192
5.12. Diagram obrazujący odwzorowanie	$\ast_{AOM}^{[rola]} \mapsto \ast_{ORM_{AOM}}^{[rola]}$	194
5.13. Diagram obrazujący odwzorowanie	$\ast_{AOM}^{[opis\ asocjacji]} \mapsto \ast_{ORM_{AOM}}^{[opis\ asocjacji]}$	195
5.14. Diagram obrazujący odwzorowanie	$\ast_{AOM}^{[dziedziczenie]} \mapsto \ast_{ORM_{AOM}}^{[dziedziczenie]}$	197
5.15. Diagram obrazujący odwzorowanie	$\ast_{AOM}^{[kolekcja]} \mapsto \ast_{UML_{AOM}}^{[kolekcja]}$	204
5.16. Diagram obrazujący odwzorowanie	$\ast_{AOM}^{[atrybut]} \mapsto \ast_{UML_{AOM}}^{[atrybut]}$	206
5.17. Diagram obrazujący odwzorowanie	$\ast_{AOM}^{[asocjacja]} \mapsto \ast_{UML_{AOM}}^{[asocjacja]}$	207
5.18. Diagram obrazujący odwzorowanie	$\ast_{AOM}^{[rola]} \mapsto \ast_{UML_{AOM}}^{[rola]} (1)$	209
5.19. Diagram obrazujący odwzorowanie	$\ast_{AOM}^{[rola]} \mapsto \ast_{UML_{AOM}}^{[rola]} (2)$	210
5.20. Diagram obrazujący odwzorowanie	$\ast_{AOM}^{[opis\ asocjacji]} \mapsto \ast_{UML_{AOM}}^{[opis\ asocjacji]}$	211
5.21. Diagram obrazujący odwzorowanie	$\ast_{AOM}^{[dziedziczenie]} \mapsto \ast_{UML_{AOM}}^{[dziedziczenie]}$	213
7.1. Wyniki ewaluacji miarą $TSA_p$		230
7.2. Wyniki ewaluacji miarą $ASA_p$		231
7.3. Wyniki ewaluacji miarą $FSA_p$		232
7.4. Wyniki ewaluacji miarą $LSA_p$		233
7.5. Wyniki ewaluacji miarą $TC_p$		234
7.6. Wyniki ewaluacji miarą $AC_p$		235
7.7. Wyniki ewaluacji miarą $FC_p$		236
7.8. Wyniki ewaluacji miarą $LC_p$		237
7.9. Wyniki ewaluacji miarami zagregowanymi		238
7.10. Wyniki ewaluacji miarami semantyki wynikowej: $RSSA_{s,d}$ , $RSC_{s,d}$ , $RSC_{s,d}$		239
8.1. Diagram AML zawierający model danych $ESNM^{SKB}$ . Źródło: [77]		242
8.2. Model $ESNM^{SKB}$ po translacji na EER		249
8.3. Model $ESNM^{SKB}$ po translacji na ORM		259
8.4. Model $ESNM^{SKB}$ po translacji na UML		264
8.5. Wyniki ewaluacji miarami $TSA_{m^s,m^d}$ , $ASA_{m^s,m^d}$ , $FSA_{m^s,m^d}$ oraz $LSA_{m^s,m^d}$		269
8.6. Wyniki ewaluacji miarami $TC_{m^s,m^d}$ , $AC_{m^s,m^d}$ , $FC_{m^s,m^d}$ oraz $LC_{m^s,m^d}$		270
8.7. Wyniki ewaluacji miarami $RSSA$ , $RC$ oraz $RS$		271