

DĄBROWSKI Włodzimierz, STASIAK Andrzej

ŚRODOWISKO PROJEKTOWANIA SYSTEMÓW STEROWANIA RUCHEM

Streszczenie

Poprawne projektowanie systemów sterowania ruchem wymaga, przy obecnej ich złożoności, wsparcia konceptualnego i narzędziowego. Niniejsze prace przedstawia propozycję budowy środowiska projektowania systemów kontroli i sterowania ruchem TransCAD. Środowisko to bazuje na języku TransML, który jest specjalizowanym, dziedzinowym językiem dedykowanym do budowy systemów kontroli i nadzoru ruchu. Proponowane środowisko projektowe może stanowić wsparcie dla ekspertów dziedzinowych przy projektowaniu systemów nadzoru ruchu. Artykuł przedstawia koncepcję języka dziedzinowego TransML, wprowadza definicję reguł dziedzinowych zapisanych w sposób formalny w języku OCL i przedstawia środowisko projektowania TransCAD. Autorzy ilustrują koncepcję zastosowań środowiska na prostym przykładzie. Definiowanie reguł dziedzinowych (w postaci formuł OCL) daje możliwość weryfikacji poprawności budowanego modelu oraz automatyzacji procesów projektowania i budowy tego typu systemów. W artykule autorzy proponują też dalsze kierunki rozwoju i badań języka TransML i środowiska TransCAD.

WSTĘP

Problem efektywnego sterowania ruchem ulicznym jest przedmiotem ciągłych badań. Badania te mają też bezpośrednie przełożenie na praktyczne zastosowania w regulacji ruchem aglomeracjach miejskich. Niniejsze prace stanowią kontynuację badań autorów nad tą problematyką. W pracach [1,2] zostały przedstawione wyniki badań uzyskanych w ramach projektu badawczego realizowanego przez Politechnikę Warszawską i francuską firmę Dynalogic w ramach grantu DynaSignal Eureka. Zaproponowano w nich struktury danych oraz interfejsy GUI do baz danych dla systemu wspomagającego projektowanie planów świateł z wykorzystaniem programowania urządzeń sterujących zainstalowanych na skrzyżowaniach.

Kolejnym krokiem badawczym jest rozszerzenie zaproponowanego podejścia o środowisko komputerowego wspomagania projektowania do sterowania ruchem ulicznym.

W artykule przedstawiona jest propozycja dziedzinowego języka modelowania TransML dedykowanego do budowy systemów kontroli i nadzorowania ruchem. Zaproponowany język dziedzinowy powinien ułatwić projektantom rozwiązań wspomagających sterowanie ruchem oraz inżynierom ruchu budowę i walidację systemów nadzoru i sterowania ruchem. Autorzy proponują metodę wykorzystania zaproponowanego środowiska CAD (ang. Computer Aided Design) w procesach projektowania systemów sterowania ruchem. Pod koniec pracy zostały zaproponowane obszary przyszłych badań.

1. JĘZYKI DZIEDZINOWE DSL

Jezyki dziedzinowe (DSL - Domain-Specific Language), w przeciwieństwie do tradycyjnych metod programowania, przeznaczone są do specyficznych zadań i rozwiązywania konkretnych problemów. Tym, co je wyróżnia jest możliwość definiowania reguł transformacji pomiędzy elementami różnych języków dziedzinowych [3]. Martin Fowler w pracy [4] proponuje podział języków dziedzinowych na dwa rodzaje – zewnętrzny język dziedzinowy (*External DSL*) oraz wewnętrzny język dziedzinowy (*Internal DSL*).¹ Język zewnętrzny, jest zapisany w języku innym niż główny język aplikacji. Jego zaletą jest dowolność składniowa, dzięki czemu możliwe staje się powiązanie go z dziedziną i uzyskanie formy stosunkowo prostej do odczytania i modyfikacji. Do wad należy natomiast zaliczyć konieczności skonstruowania dla niego translatora, trudności w komunikowaniu i integrowaniu się z językiem podstawowym programu oraz spowodowany nim problem kakofonii językowej. Wewnętrzny język dziedzinowy jest utworzony jest na bazie języka podstawowego. Jego składnia jest taka sama jak języka podstawowego, jednocześnie jednak zawiera on pewne charakterystyczne konstrukcje, wyróżniające go na tle języka podstawowego. Ten rodzaj DSL pozwala na komunikowanie się i integrowanie z językiem podstawowym, jednak jego możliwości dostosowania się do dziedziny są ograniczone. Język wewnętrzny jest sposobem wypracowania wspólnego rozwiązania przez specjalistów z danej dziedziny, a język zewnętrzny pozwala na komunikację między różnymi ich grupami.

Główną korzyścią płynącą z zastosowania języków dziedzinowych, jest ich użyteczność w identyfikowaniu istotnych cech i wymagań występujących na różnych poziomach abstrakcji. Dzięki precyzyjnemu wyrażaniu ograniczeń nałożonych na konstrukcje językowe, możliwa jest także kontrola poprawności elementów składowych oraz walidacja tworzonych modeli. W przypadku zastosowania języków dziedzinowych do reprezentacji modeli, możliwe jest utworzenie odrębnych języków dla każdego z nich – konceptualnego, logicznego i fizycznego. Dobór odpowiedniego języka dziedzinowego, wpływa natomiast na proces walidacji modelu konceptualnego – jego czytelność oraz operowanie pojęciami bezpośrednio związanymi z obszarem zastosowania, ułatwia ekspertowi dziedzinowemu dokonanie przeglądu. Niewątpliwą zaletą języków dziedzinowych jest również automatyzacja procesu transformacji modeli.

Warto pamiętać, że dwa typy domen stają się tematem DSL. Pionowe domeny skupić się na kluczowych cechach i abstrakcji z danej firmy lub branży. Poziome domeny skupić się na technologii i szkieletach wytwórczych – ich ramach.

Modelowanie specyficznej dziedziny (DSM - Domain-Specific Modeling) - wymaga użycia języka dziedzinowego (DSL) do: specyfikacji, budowy i dokumentowania - budowanych systemów.

2. DEFINIOWANIE JĘZYKA TRANSML

Przedstawiona przez nas w niniejszym artykule propozycja języka transML, wg Martina Fowlera [4], stanowi (internal DSL) – będący dialektem języka UML, który został przez nas zaimplementowany w postaci profilu języka UML. Ponieważ transML jest adresowany dla inżynierów sterowania ruchem odwołuje się on do koncepcji dziedziny pionowej (Vertical domain [3]), tzn. skupia się na kluczowych cechach i abstrakcjach z obszaru inżynierii ruchu drogowego.

W naszej propozycji zasadniczymi użytkownikami modeli specyficznej dziedziny, której dotyczy artykuł, obok zespołu IT, będą inżynierowie sterowania ruchem ulicznym, jako

¹) Naszym zdaniem w miejsce bezpośredniego tłumaczenia terminów: external i internal DSL, lepszym będzie stosowanie wobec nich określeń: autonomiczny DSL i pochodny DSL.

interesariusze biznesowi (business stakeholders), którzy muszą wyrazić i zrozumieć problem, który należy rozwiązać.

Warto tu zauważyć, że poszukiwane rozwiązanie ogranicza swój zakres jedynie do pojęć i zasad specyficznej dziedziny, opisanej w języku DSL, który z założenia ogranicza sposób budowy rozwiązania tak, aby unikać błędów projektowych.

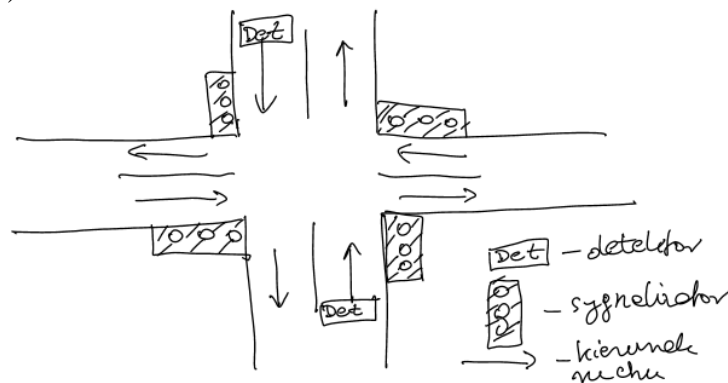
Model dziedzinowy pozwala także na bardziej efektywną - od modeli ogólnego przeznaczenia komunikację, zapewniając lepsze zrozumienie dla budowanego rozwiązania przez wszystkich jego uczestników. Może więc on zapewnić optymalny poziom szczegółowości komunikacji między tymi, którzy pracują w tej dziedzinie.

2.1. Szkic języka

Opracowanie nowego języka dziedzinowego zawsze wymaga przeprowadzenia szczegółowej analizy dziedziny, którą pozwoli on opisać. Proces tej analizy najczęściej jest inicjowany przez opracowanie słownika i wykonanie wstępnego szkicu istotnych fragmentów dziedziny.

Proces szkicowania języka [5,6] polega na wyszukaniu w modelowanym wycinku rzeczywistości (dziedzinie) pojęć i zasad określających poprawne w nim zachowanie.

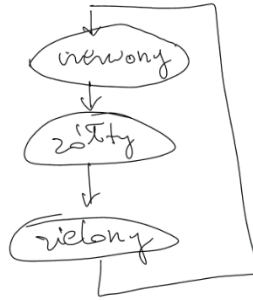
Zwykle taki szkic wykonuje się odręcznie. Ilustruje on pojęcia występujące w dziedzinie, które język wynikowy będzie w stanie przedstawić oraz pozwoli na weryfikację poprawności budowanych zdań przez definicję jego semantyki i syntaktyki. Dwa szkice poniżej przedstawiają podstawowe elementy odcinka drogi, na którym znajdują się skrzyżowania oraz wymagana do sterowania ruchem na nich sygnalizacja świetlna (rys. 1) i zachowanie tej sygnalizacji (rys. 2).



Rys. 1. Szkic modelu ruchu drogowego na jednopoziomowym skrzyżowaniu zwykłym

Proces szkicowania rozpoczynamy od określenia zbioru pojęć modelowanej dziedziny, które odnoszą się do jej struktury (tu stanowią je elementy dróg – w tym skrzyżowania) (rys.1). Jak wynika z przedstawionego szkicu, aby poprawnie opisać skrzyżowanie (a właściwie reguły ruchu drogowego na nim), niezbędne jest użycie pojęć, takich jak: droga, skrzyżowanie, wylot ze skrzyżowania, jezdnia, pas ruchu, kierunek ruchu, detektor, sygnalizator.

Następnie niezbędne jest określenie pojęć, które odnoszą się do specyfikacji zachowania zdefiniowanej struktury. Przykładowo, zdefiniowany element struktury, jakim jest sygnalizator świetlny ma wiele swoich postaci, tj. sygnalizator (jako generalizacja), może być zastąpiona dowolną specjalizacją: sygnalizator z czterema światłami, sygnalizator z trzema światłami (rys.2), sygnalizator z dwoma światłami, czy sygnalizator z jednym światłem.



Rys. 2. Szkic sekwencji zmian trzech świateł w ruchu drogowym z segregacją czasową przez sygnalizację świetlną





Problem sterowania sygnalizacją świetlną oczywiście jest bardziej złożony, ale na etapie szkicowania języka ograniczamy go do reprezentatywnego przypadku zakładając, że pozostałe modele będą mogły być opisane za pomocą pojęć określonych dla tego modelu.







Podsumowując, należy pamiętać, że szkicowanie pozwala określić „zasadnicze ramy” dziedziny, ponieważ ma na celu zobrazowanie wszystkich istotnych elementów języka, zawierających pewne wspólne cechy i właściwości, które muszą zostać włączone do składni budowanego języka dziedzinowego.

2.2. Pojęcia języka i jego metamodel

Na podstawie analizy utworzonych szkiców (rys. 1) i (rys. 2), stanowiących nieformalny opis dziedziny, opracowywany jest słownik (tab. 1), w którym każde zidentyfikowane pojęcie ma przypisywane znaczenie w dziedzinie problemu oraz nazwę.

Tab. 2. Semantyka pojęć języka transML

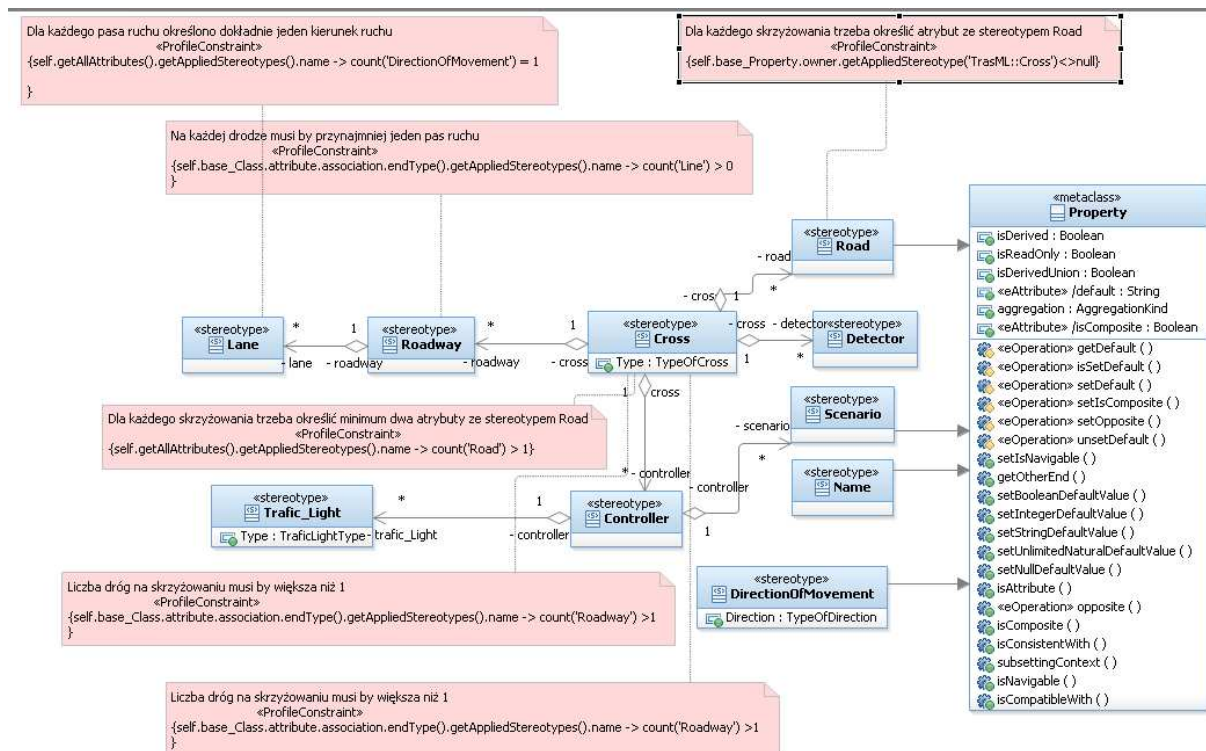
Lp.	Nazwa pojęcia	Piktogram	Znaczenie pojęcia
1	Distance (dystans)		odległość między dwoma wskazanymi punktami na drodze (np.: między skrzyżowaniami)
2	Cross (skrzyżowanie)		zasadnicze pojęcie języka - określa odcinek drogi na którym możliwa jest zmiana kierunku ruchu pojazdu, węzeł drogowy
	Scenario (scenariusz)		opis zachowania, który określa przebieg sterowania światłami w ruchu ulicznym dla danego skrzyżowania, a obsługiwany jest przez konkretny kontroler
4	Controller (kontroler)		urządzenie, które realizuje ustalony scenariusz sterowania światłami w ruchu ulicznym

5	Trafic light (światła uliczne)		obiekt sterowania, który przedstawia uczestnikom ruchu decyzje określone w scenariuszu za pomocą sygnałów świetlnych
6	Lane (pas ruchu)		określa odcinek drogi po którym mogą poruszać się uczestnicy ruchu w jednym kierunku.
7	Trafic light type (typ świateł ulicznych)		światła z jedną, dwoma, trzema, ..., lampami
8	Detector (detektor)		inteligentny czujnik umożliwiający zarówno wykrycie specyficznych obiektów w obszarze skrzyżowania jak i np: ich zaliczenie.
9	State of lane (stan pasa)		Zakładamy, że pas ruchu może przebywać w jednym z trzech stanów: sprawny odcinek drogi, częściowo sprawny odcinek drogi, wyłączony odcinek drogi.
10	incydent na drodze		zdarzenie na drodze, które prowadzi do zmiany jej stanu np.: wypadek drogowy, roboty drogowe, ... Zdarzenie zawsze posiada min. dwa atrybuty, tj: czas i miejsce – określone przez dystans
	ConsistsOut	Asocjacja	Relacja określająca składniki typu wyjście (np.: do skrzyżowania)
	ConsistsIn	Asocjacja	Relacja określająca składniki typu wejście (np.: do skrzyżowania)
	KindOf (rodzaj)	Generalizacja	Relacja, która odwołuje się do postulatu pełnej zastępowalności uogólnienia przez uszczegółowienie

Ponieważ budowany język dziedzinowy stanowi język graficzny, z każdym pojęciem wiążemy piktogram – który stanowi jego metaforę, tzn. powinien przez eksperta dziedzinowego być bezpośrednio zrozumiały.

Następnie budowany jest metamodel języka dziedzinowego, określane często mianem modelu koncepcyjnego. Podobnie jak słownik opisuje on podmioty (byty), ale ich reprezentację stanowią obiekty (konceptje) i ich atrybuty, relacje oraz reguły ich łączenia i ograniczenia. Metamodel opisywany jest na diagramie klas języka UML, który zobrazuje słownictwo i podstawowe pojęcia z dziedziny problemu. Metamodel stanowi zarówno wzór jak i weryfikator budowanych modeli (opis gramatyki języka).

Reprezentacją metamodelu, budowanego przez nas języka dziedzinowego TransML, jest profil UML (rys. 3), którego stereotypy jako nowe pojęcia języka są mapowane na metaklasy języka UML, a ograniczenia definiowane są w postaci typów enumeratywnych i wyrażeń w języku OCL.



Rys. 3. Profil języka TransML (bez pojęć opisujących relacje)

Ponieważ opracowywane modele dziedziny, wyrażone w sposób obiektowy za pomocą pojęć z profilu, określają sformalizowany model pojęciowy, to mogą być wykorzystane do automatycznej weryfikacji i walidacji modeli dziedziny przez budowane środowisko CAD.

3. KOMPUTEROWE ŚRODOWISKO PROJEKTOWANIA SYSTEMÓW STEROWANIA RUCHEM ULICZNYM

Budowane przez nas środowisko stanowi narzędzie typu CAD (ang. Computer Aided Design). Posiada ono własne repozytorium projektu oraz zbiór edytorów graficznych, które poprzez piktograficzne metafory, reprezentowane jako paleta pozwoli na budowanie modeli w zaproponowanym przez nas języku dziedziny - TransML.

Pojęcia tego języka zostały zdefiniowane w postaci słownika i metamodelu (rozdział 2) dla wycinka rzeczywistości dotyczącego opisu elementów składających się na topologię drogi, w tym strukturę skrzyżowań dróg oraz możliwości odwzorowania zagadnień dotyczących inżynierii ruchu drogowego (w tym użytych znaków drogowych i sygnalizacji świetlnej).

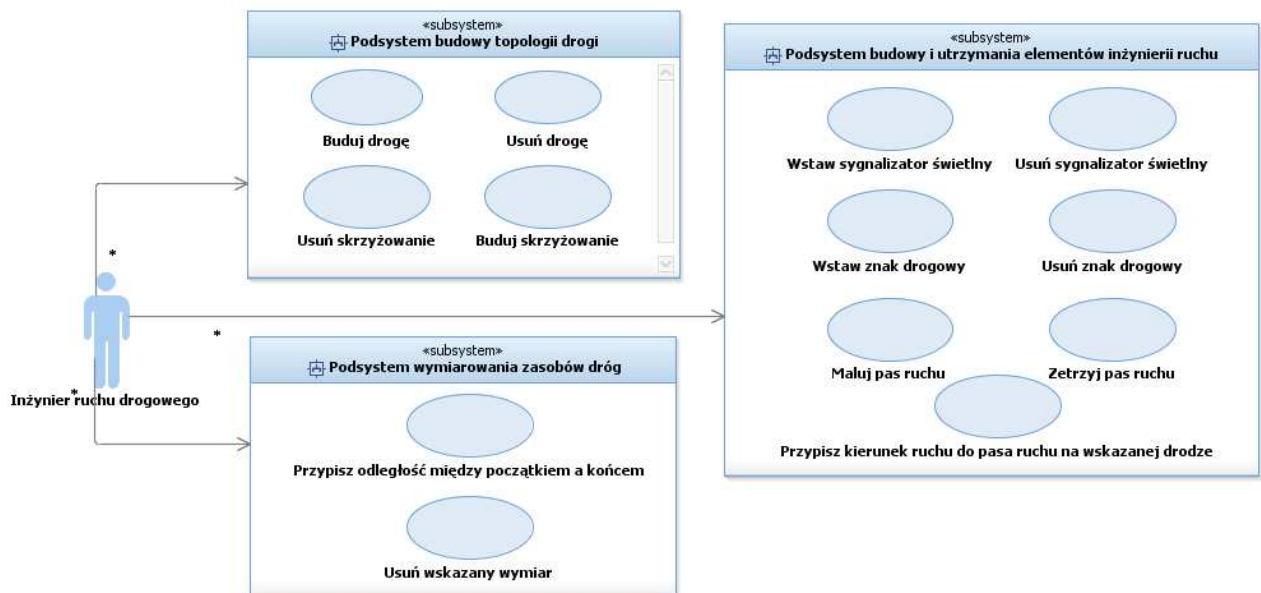
Najciekawszym elementem budowanego środowiska TransCAD, w naszej ocenie, jest możliwość definiowania reguł - w postaci wyrażeń napisanych w języku OCL, jako ograniczeń modelu.

3.1. Wymagania na system TransCAD

W artykule zaprezentujemy jedynie kluczowe wymagania, które pozwolą zbudować obraz obszarów budowanego środowiska, a nie szczegółowe wymagania (ze względu na ograniczenia publikacyjne).

Zakres zaprezentowanych wymagań ograniczymy jedynie do specyfikacji funkcjonalności systemu, poprzez określenie: użytkowników systemu (aktorów) i jego usług (przypadków użycia). Wstępnie, usługi budowanego środowiska CAD zostały zgrupowane w trzy podsystemy:

1. Podsystem budowy topologii drogi:
 - Buduj drogę;
 - Usuń drogę;
 - Buduj skrzyżowanie;
 - Usuń skrzyżowanie;
2. Podsystem budowy i utrzymania elementów inżynierii ruchu:
 - Maluj pas ruchu;
 - Zetrzyj pas ruchu;
 - Przypisz kierunek ruchu do pasa ruchu na wskazanej drodze;
 - Wstaw znak drogowy;
 - Usuń znak drogowy;
 - Wstaw sygnalizator świetlny;
 - Usuń sygnalizator świetlny;
3. Podsystem wymiarowania zasobów dróg:
 - Przypisz odległość między początkiem a końcem;
 - Usuń wskazany wymiar;



Rys. 4. Usługi systemu TransComp

3.2. Model ograniczeń

W rozdziale tym przedstawiamy wybrane ograniczenia i ich implementację w języku OCL. Poprawność ograniczeń została zweryfikowana w procesie testowania środowiska TransCAD.

-- Na każdej drodze musi być przynajmniej jeden pas ruchu

context Roadway **inv:**

```
self.base_Class.attribute.association.endType().getAppliedStereotypes().name -> count('Line') > 0
```

-- Dla każdego skrzyżowania trzeba określić atrybut ze stereotypem Road

context Road **inv:**

```
self.base_Property.owner.getAppliedStereotype('TrasML::Cross') <> null
```

-- Dla każdego pasa ruchu określono dokładnie jeden kierunek ruchu

context Lane **inv:**

```
self.getAllAttributes().getAppliedStereotypes().name -> count('DirectionOfMovement') = 1
```

-- Liczba dróg na skrzyżowaniu musi być większa niż 1

context Cross **inv**:

```
self.base_Class.attribute.association.endType().getAppliedStereotypes().name -> count('Roadway') > 1
```

-- Dla każdego skrzyżowania trzeba określić minimum dwa atrybuty ze stereotypem Road

context Cross **inv**:

```
self.getAllAttributes().getAppliedStereotypes().name -> count('Road') > 1
```

-- Kontroler musi mieć zdefiniowany przynajmniej jeden atrybut ze scenariuszem

context Controller **inv**:

```
self.getAllAttributes().getAppliedStereotypes().name -> count('Scenario') > 0
```

3.3. Model środowiska TransCAD

Środowisko TransCAD w naszym projekcie stanowi kompleksowe rozwiązanie w zakresie komputerowo wspomaganego projektowania modeli ruchu drogowego.

Środowisko to wspiera następujące procesy:

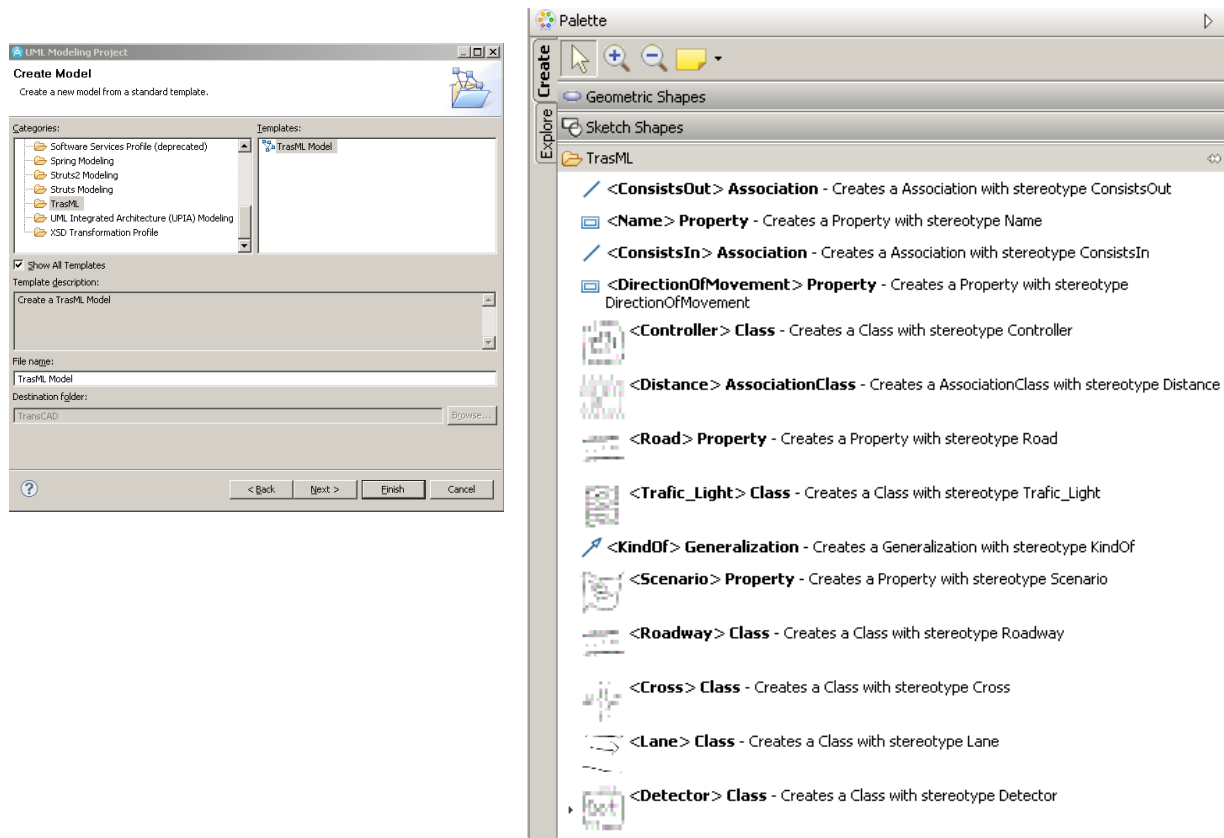
- 1) Budowy środowiska CAD – szkielet tego środowiska stanowi produkt, IBM Rational Software Architect (IBM RSA), który rozszerzono o autorski dodatek (plugin) - w postaci profilu języka TransML (rys. 5) oraz pakiet IBM RSA Simulation Toolkit. Zbudowane przez nas środowisko jest otwarte na nowe pojęcia języka DSL – TransML, a profile dla kolejnych jego wydań będą budowane w procesach automatycznej generacji kodu.
- 2) Budowy modeli – Inżynier ruchu drogowego buduje swoje modele w środowisku TransCAD (rys. 6) i zapisuje je w repozytorium IBM RSA;
- 3) Przeprowadzania eksperymentów (w tym budowy różnych wariantów rozwiązań). Dzięki pakietowi IBM RSA Simulation Toolkit opracowane modele zachowania mogą być symulowane (tak jak modele UML);
- 4) Oceny uzyskanych wyników.

W zaproponowanym narzędziu symulacja komputerowa została wykorzystana do badania opracowanych modeli topologii dróg i modeli ruchu drogowego na tych drogach. Modele te zostały opracowane w obiektowym języku modelowania TransML, bazującym na języku UML i standardach z nim związanych, a wspieranych przez grupę OMG, tj: fUML, Alf, OCL [7].

Podczas symulacji TransCAD odwzorowuje powiązane z topologią drogi zachowanie jej użytkowników (opisane przez scenariusz), kolejno przechodząc przez elementy topologii drogi wg. kolejności ustalonej w scenariuszu. Sterowanie procesem symulacji odbywa się poprzez zdarzenia, z poziomu kodu napisanego w języku Alf bądź ręcznie (np. przy blokach decyzyjnych).

W zbudowanym środowisku TranCAD (podobnie jak w IBM RSA) symulacja może być przeprowadzana w trybie wykonawczym (ang. execute), bądź w trybie debugowania (ang. debug). W pierwszym trybie inżynier ruchu drogowego jest tylko obserwatorem symulacji, natomiast w drugim trybie ma możliwość ingerowania w wykonanie (np. poprzez zmianę w modelach), podgląd i zmianę stanu zmiennych, wybór ścieżek decyzyjnych, itp.

Dodatkowo środowisko TransCAD umożliwia zebranie następujących wyników symulacji: historii komunikatów przesyłanych między obiektami, śladów przepływu komunikatów, historii zapisów na konsoli operatora symulacji. Możliwości środowiska można znacznie rozszerzyć stosując język Alf lub UAL do specyfikacji zachowania konkretnych elementów modelu, w celu np. symulacji interakcyjnej pracy, wykorzystania plików, czy stosowania złożonych struktur danych.



Rys. 5. Paleta zbudowanego języka TrasML wraz z opracowanym szablonem projektu

4. PROPONOWANA METODA I PRZYKŁAD JEJ ZASTOSOWANIA

Proponowana metoda bazuje bezpośrednio na możliwościach badania (weryfikacji i walidacji) modeli konceptualnych wynikających z zasad obowiązujących w danej dziedzinie, dla której opracowano język domenowy – DSL.

Zasadniczo badanie modelu konceptualnego polega na jego przeglądzie i analizie przez eksperta dziedzinowego, który określa czy model:

- w procesach weryfikacji - został zbudowany zgodnie ze sztuką (składnia modelu jest poprawna);
- w procesach walidacji - przedstawia rozwiązanie, które było przedmiotem modelu, czy nie. Inaczej rzecz ujmując: czy odpowiada zdefiniowanym wymaganiom, czy nie.

Czytelność opracowanego modelu, a zwłaszcza to, czy model ten operuje pojęciami bezpośrednio związanymi z dziedziną zastosowania, decydują o trudnościach w procesach jego badania i analizy.

Ponieważ zakres proponowanej metody ogranicza się do projektowania dróg (w tym skrzyżowań drogowych²) należało w nim uwzględnić dwa etapy prac:

1. badanie i analizę ruchu oraz przedstawienie koncepcji wariantów rozwiązania,
2. projekt techniczny wybranego wariantu drogi.

Obecnie zakładamy, że proponowana metoda będzie wykorzystywana w pierwszym z powyższych zakresów (pomijamy procesy wspomaganie projektów technicznych).

W metodzie tej inżynier ruchu, wyposażony w komputerowe środowisko projektowania sterowania ruchem ulicznym TransCAD, opracuje a następnie dokona badania wariantów rozwiązania dla drogi będącej przedmiotem projektu. Wynikiem analizy wariantów

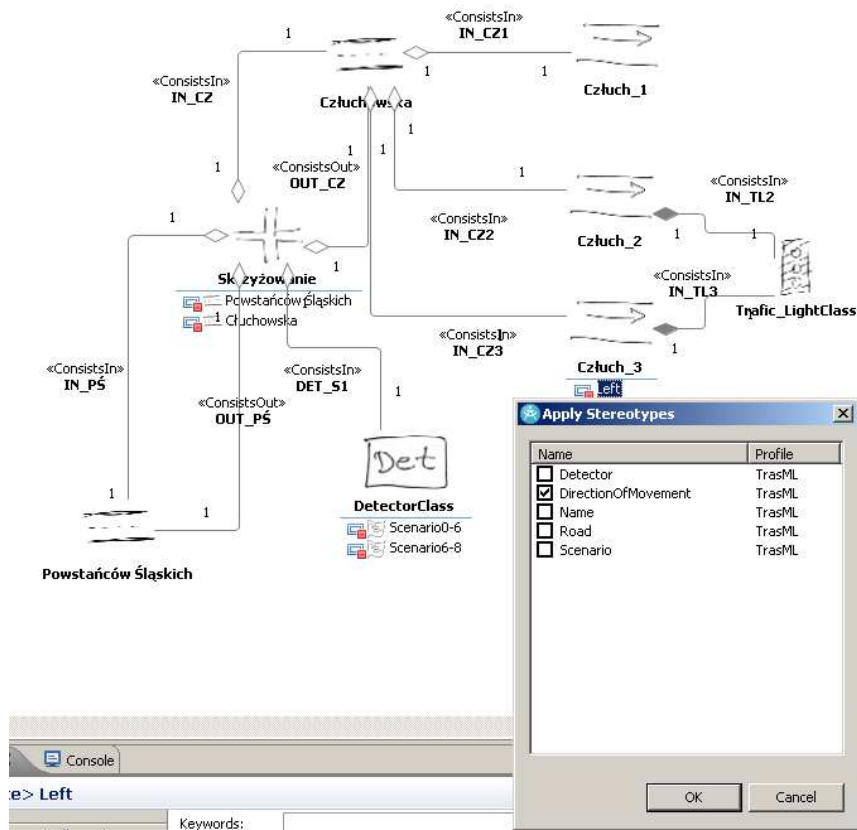
²) Prawidłowo zaprojektowane skrzyżowanie powinno być: łatwo rozpoznawalne w ciągu drogi, przejrzyste, zrozumiałe i przejezdne.

rozwiązania będzie jego decyzja, którą stanowił będzie symulacyjnie zweryfikowany i zwalidowany model drogi o wysokiej przepustowości ruchu (z uwzględnieniem segregacji czasowej ruchu za pomocą sygnalizacji świetlnej).

1. Uproszczony scenariusz metody sprowadza się do wykonania następujących zadań:
2. buduj topologię drogi
 - a. buduj skrzyżowania dróg jednopoziomowych
 - b. (skrzyżowania zwykłe, skanalizowane, z wyspą centralną, o ruchu okrężnym, z koleją);
 - c. buduj ronda
 - d. (ronda duże, ronda małe, mini ronda, skrzyżowania z wyspą centralną (kołową, eliptyczną, rombowa);
 - e. buduj odcinek drogi;
 - f. ...
3. dokonaj weryfikacji i walidacji projektu topologii drogi;
4. przypisz odległości (i pozostałe wymagane atrybuty modelu);
5. rozmieść elementy inżynierii ruchu drogowego:
 - a. wprowadź segregację czasową ruchu za pomocą sygnalizacji świetlnej;
 - b. znaki drogowe;
6. dokonaj symulacyjnego badania ruchu drogowego:
 - a. określ punkty kolizji;
 - b. wyznacz powierzchnię kolizji;
 - c. określ przepustowości;
 - d. ...
7. raportuj o przeprowadzonych badaniach;

....

Obecnie środowisko TransCAD jedynie w ograniczonym zakresie wspiera proponowany scenariusz, tj. budowę topologii drogi i przypisywanie atrybutów elementów modelu oraz ograniczoną weryfikację i walidację modelu topologii drogi. W wyniku zastosowania zaproponowanej metody zbudowano uproszczony model dla jednego z Warszawskich skrzyżowań (rys. 6).



Rys. 6. Przykład modelu w języku TrasML opisu skrzyżowania zbudowany w środowisku TransCAD

PODSUMOWANIE

Artykuł prezentuje koncepcję budowy dedykowanego języka dziedzinowego TransML, który może być wykorzystany do modelowania i projektowania systemów kontroli ruchu. Proponowany język ze względu na swoją prostotę może być wykorzystywany przez ekspertów dziedzinowych bez konieczności znajomości rozbudowanych języków projektowania uniwersalnego. Wprowadzenie do języka definicji reguł dziedzinowych umożliwia kontrolę poprawności budowanych modeli i otwiera drogę do automatycznych transformacji modelu dziedzinowego.

W artykule zaprezentowano również propozycję środowiska projektowego TransCAD dla systemów sterowania ruchem ulicznym, a szerzej rzecz ujmując - zagadnień związanych z inżynierią ruchu drogowego.

Ramę technologiczną dla budowanego narzędzia CAD, w naszej propozycji, stanowią: CASE - IBM RSA ver. 9.0 z pakietem IBM RSA Simulation Toolkit. Rama ta wymagała od autorów budowy jej rozszerzeń w postaci plugin'ów, z których na obecnym etapie prac najistotniejszym jest dodatek w postaci oprogramowania palety narzędziowej dla opracowanego języka dziedzinowego TransML.

Opisany język może zostać rozszerzony o cechy języka i pojęcie perspektywy projektowej, które pozwoli na nanoszenie na zdefiniowaną sieć dróg, informacji o zagrożeniach dla różnych służb i obywateli, którzy mogliby wykorzystywać te informacje poprzez serwisy web i aplikacje typu gov.

Dodatkowo zakładamy, że dzięki zaproponowanemu środowisku TransCAD można by zredukować zadania centrów kryzysowych, ponieważ możliwe byłoby jednoczesne informowanie o każdym incydencie wszystkich służb, po wprowadzeniu incydentu przez jedną z nich. Wówczas prostym rozszerzeniem opracowanych modeli byłaby informacja o incydentach np: w sieci gazowniczej, czy zagrożeniach powodziowych.

Warto również zauważyć, że ciekawą własnością zaproponowanego podejścia jest możliwość zbudowania modelu growego, który pozwoli na zbudowanie gry edukacyjnej dla kadr sterowania ruchem ulicznym i wykorzystania proponowanego środowiska jako platformy szkoleniowej dla pracowników zajmujących się sterowaniem i nadzorem ruchu.

W modelu tym ocena gracza zależałaby od odległości pomiędzy decyzją gracza, a decyzją wyznaczoną przez system (dla zweryfikowanych scenariuszy, optymalną - określoną przez eksperta dla wcześniej określonego scenariusza, lub ustaloną przez system dla scenariuszy przez niego generowanych).

Opisana koncepcja języka TransML oraz środowiska TransCAD może być rozwijana w wielu kierunkach i według nas stanowi interesujący kierunek badań nad konceptualnym i narzędziowym wsparciem procesów zarządzania i sterowania ruchem w złożonych systemach drogowych.

BIBLIOGRAFIA

1. Amborski K., Dąbrowski W., Kowalczyk P., Markowski K., *Logistyka* 3/2011, *Zastosowanie baz danych w sterowaniu ruchem ulicznym*, ISSN 1231-5478
2. W. Dąbrowski, P. Kowalczyk, K. Amborski, P. Kruczkowski, *Struktury GUI do baz danych w zastosowaniu do sterowania ruchem ulicznym w: Computer Systems aided science industry and transport*, Transport Committee of the Polish Academy of Science, 2011.
3. Dubilewicz I., Hnatkowska B., Huzar Z., Tuzinkiewicz L. (2008). *Języki dziedzinowe w modelowaniu danych w kontekście MDA*, [w:] S. Kozielski, B. Małysiak, P. Kasprowski, D. Mrozek (red.), *Bazy danych. Rozwój metod i technologii. Architektura, metody formalne i zaawansowana analiza danych*, tom 1. Warszawa: Wydawnictwa Komunikacji i Łączności.
4. Fowler M, *Domain Specific Languages*, Pearson Education Inc, 2012
5. A.Kosior, A.Stasiak, W.Dąbrowski "Wytwarzanie serwisów informacyjnych z wykorzystaniem koncepcji modelowania dziedzin. Budowa transformacji", *Biuletyn Instytutu Automatyki i Robotyki*, 32/201.
6. Student Manual, „Domain-Specific Modeling with IBM Rational Software Architect v7.5”, RD420 July 2009;
7. Concrete Syntax for a UML Action Language: Action Language for Foundational UML (ALF), OMG, marzec 2013

DESIGN ENVIRONMENT FOR TRAFFIC CONTROL SYSTEMS

Abstract

Proper design of complex control systems needs conceptual tools. This paper presents a proposal for the construction of design environment for control of traffic systems TransCAD . This environment is based on the language TransML, which is a specialized , domain centric language for experts in the field of traffic control systems. This paper presents the concept of TransML language , provides a definition of domain rules written in a formal language OCL and presents design environment TransCAD. The authors illustrate the concept uses a simple example. Defining the rules of domain (in the form of formulas OCL) provides the opportunity to verify the correctness of the model under construction and automation of the design and construction of such systems. In this paper the authors discuss future development of the language TransML and the environment TransCAD.

Autorzy:

dr inż. **Włodzimierz Dąbrowski** – Politechnika Warszawska, Wydział Elektryczny, Instytut Sterowania i Elektroniki Przemysłowej, 1. Koszykowa 75, Warszawa, e-mail: w.dabrowski@ee.pw.edu.pl

dr inż. **Andrzej Stasiak** – Wojskowa Akademia Techniczna, Wydział Cybernetyki, Instytut Teleinformatyki i Automatyki, ul. Kaliskiego 2, Warszawa, e-mail: astasiak@wat.edu.pl