

Krzysztof GOCZYŁA, Aleksander WALOSZEK, Wojciech WALOSZEK¹
Politechnika Gdańska, Wydział Elektroniki, Telekomunikacji i Informatyki

SYSTEM OBSŁUGI KONTEKSTOWYCH BAZ WIEDZY CONGLOS

Streszczenie. W artykule przedstawiono założenia, architekturę i niektóre aspekty implementacyjne systemu CongloS – systemu obsługi kontekstowych baz wiedzy opartych na modelu SIM. W swoim działaniu CongloS wykorzystuje wiele specyficznych elementów koncepcji organizacji bazy wiedzy: model tarsetowy, model SIM, metaontologię opisu schematu, własne sprzęgi międzymodularne i możliwość zadawania zapytań. Koncepcje te również są opisane w artykule, wraz z zarysem sposobu ich realizacji.

Słowa kluczowe: ontologie, bazy wiedzy, konteksty, kontekstualizacja, CongloS

CONGLOS – MANAGER FOR CONTEXTUAL KNOWLEDGE BASES

Summary. In the paper we present CongloS, a manager for contextual knowledge bases, its architecture and selected implementation issues. In the CongloS we exploit a specific approach combining together ideas of tarset and SIM knowledge bases, metaontological encoding of the base, intermodular couplers and *ad hoc* queries.

Keywords: ontologies, knowledge bases, contexts, contextualization, CongloS

1. Wprowadzenie

Niniejszy artykuł opisuje system CongloS – system informatyczny służący do wspierania procesu projektowania, tworzenia i wykorzystywania baz wiedzy zbudowanych według podejścia o tej samej nazwie.

W ramach systemu CongloS wykorzystanych zostało wiele opracowanych wcześniej koncepcji. Projektant i użytkownik postrzegają utworzoną bazę wiedzy jako podzieloną na konteksty. Konteksty zorganizowane są hierarchicznie (niższe opisują mniejszy fragment dzie-

¹ Praca częściowo finansowana z funduszy NCBiR, grant nr SP/I/1/77065/10, projekt SYNAT.

dziny zainteresowań, ale w większych szczegółach) zgodnie z modelem SIM. Sama budowa bazy oparta jest jednak na koncepcji tarsetów (wcześniej nazywanych konglomeratami). Wykorzystanie algebry tarsetów pozwala m.in. na zadawanie zapytań *ad hoc* oraz tworzenie niestandardowych powiązań między modułami (sprzęgów). Te i inne idee połączono w ramach CongloS w jedno spójne podejście, które dokładniej przybliżone jest w rozdziale 2.

Implementacja systemu CongloS została wykonana w ramach prac nad projektem SyNaT. SyNaT, czyli System dla Nauki i Techniki, jest projektem NCBiR ukierunkowanym na stworzenie otwartej platformy dla zasobów nauki i wiedzy. Projekt dostarczył tym samym interesującego środowiska wdrożeniowego dla systemu CongloS, co wymagało zastosowania specyficznej architektury, której szczegóły opisano w rozdziale 3.

Opracowana architektura pozwala na wykorzystanie utworzonego systemu do różnych zadań w różnych tzw. konfiguracjach docelowych. Pierwsza opracowana konfiguracja docelowa obejmuje narzędzia projektanta bazy wiedzy i posłużyła do stworzenia edytora baz SIM zintegrowanego (w formie wtyczek) ze znanym edytorem ontologii Protégé [1]. Konfiguracja ta opisana jest dokładniej w rozdziale 4.

Podsystemy systemu służą do realizacji koncepcji algorytmów i metod wnioskowania i przetwarzania baz wiedzy wykorzystujących podejście CongloS. Najciekawsze mechanizmy opisane zostały w rozdziale 5 niniejszego artykułu.

2. Koncepcja budowy baz wiedzy typu CongloS

W ciągu ostatnich lat intensywnie pracujemy nad metodami opisu i organizacji baz wiedzy, które udoskonaliby procesy ich projektowania, budowy i wykorzystywania do procesów znanych dla relacyjnych baz danych. Owocem tych prac jest kilka mechanizmów i idei, które udało się powiązać w ramach podejścia nazwanego CongloS.

Modelowanie koncepcyjne (czy też logiczne; odpowiednik etapu tworzenia modelu E-R dla relacyjnych baz danych) jest kluczowym elementem procesu tworzenia bazy wiedzy. Dla tego etapu proponujemy metodę projektowania kontekstowych baz SIM. Stoimy przy tym na stanowisku, że kontekstualizacja jest integralną częścią procesu konceptualizacji, jako że wszystkie zdania w dowolnej bazie wiedzy są wypowiedane w pewnym kontekście (myśl ta rozwinięta jest m.in. w [2]). Jednocześnie traktujemy tu kontekstualizację jako drogę do poprawnej modularyzacji bazy wiedzy, co daje dużo korzyści w procesie jej budowy (np. możliwość rozdzielenia pracy między zespoły) i w przyszłym jej utrzymaniu i użytkowaniu (m.in. zwiększenie wydajności wnioskowania, ułatwienie modyfikowania itd.).

W metodzie SIM konteksty zorganizowane są hierarchicznie: niższe opisują mniejszy fragment dziedziny zainteresowań, ale w większych szczegółach. Zadaniem projektanta jest

utworzenie struktury terminologii (TBox) i szkieletowej struktury opisów świata (ABox) wraz ze wskazaniem sposobu ich rozszerzania. W ten sposób powstaje schemat bazy SIM (więcej o metodzie SIM piszemy m.in. w [3]).

Sama implementacja bazy odbywa się za pomocą tarsetów, wcześniej nazywanych konglomeratami² (patrz też [4]). Tarsety to moduły semantyczne, niezależne od zdaniowego sposobu ich opisu, zdefiniowane jako zbiory interpretacji Tarskiego. Dla tarsetów zdefiniowano algebrę obejmującą ważne operacje integracji wiedzy z kilku modułów. Podejście tarsetowe wraz z algebrą jest bardzo elastyczne i pozwala na wyrażenie modularnych baz wiedzy skonstruowanych za pomocą innych metod, jak DDL i \mathcal{E} -Connections [5].

Baza tarsetowa składa się ze zmiennych tarsetowych i sprzęgów, czyli wyrażeń algebry tarsetów definiujących powiązania między modułami. Schemat bazy określa sposób wprowadzania do niej nowych modułów (zmiennych tarsetowych). Bazę można odpytywać za pomocą wyrażeń algebry w taki sam sposób, jak w przypadku języka SQL.

Można zatem powiedzieć, że tarsety stanowią odpowiednik relacji dla baz relacyjnych. Podobnie jak w przypadku E-R i schematu relacyjnej bazy danych, istnieje kilka sposobów przełożenia modelu SIM na schemat bazy tarsetowej, jednak zadanie to w ogólności nie jest trudne i może być przeprowadzone automatycznie. W takim ujęciu istnieje też możliwość definiowania dodatkowych sprzęgów (poza modelem SIM) specyfikujących niestandardowe powiązania między modułami (pełniących rolę podobną do procedur składowanych w bazach danych).

Zestaw metod składających się na podejście CongloS dopełnia sposób zapisu schematów baz wiedzy. Schemat zapisywany jest za pomocą specjalnej ontologii (zwanej metaontologią), możliwej do wyrażenia np. w języku OWL. W zasadzie zapis dotyczy schematu tarsetowej bazy wiedzy, jednak dzięki wykorzystaniu zapisu ontologicznego można było zdefiniować specjalne aksjomaty, dzięki którym wiedza o schemacie tarsetowym może być przełożona na wiedzę o schemacie SIM i odwrotnie. Więcej informacji na ten temat można znaleźć w [6].

3. Architektura systemu CongloS

W niniejszym rozdziale zaprezentowano założenia architektoniczne systemu CongloS. System ten ma budowę komponentową, a zamysłem autorów było zachowanie możliwości integrowania komponentów dla różnych konfiguracji docelowych. Aby to osiągnąć, wykorzystano architekturę warstwową – ogólne założenia architektoniczne opisane są w podrozdziale 3.1.

² Nową nazwę wprowadzono m.in. ze względu na chęć ujednoczenia terminologii polskiej i anglojęzycznej.

Każda z wyróżnionych warstw jest podzielona na komponenty, głównie z myślą o oddzieleniu od siebie usług obsługi baz czysto tarsetowych od usług przeznaczonych specjalnie dla baz SIM (patrz rozdział 2). Dokładniej zawartość poszczególnych warstw prezentuje podrozdział 3.2.

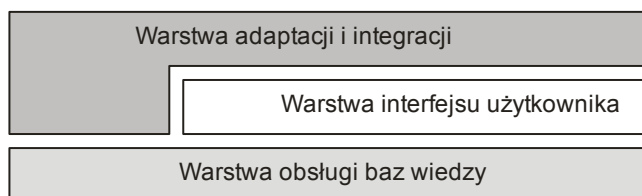
Opisywanego podziału architektonicznego udało się dokonać również dzięki wykorzystanym standardom i narzędziom. Bardzo dużą rolę odegrało tu użycie OSGi [7]. Te aspekty podziału architektonicznego przedstawiono w podrozdziale 3.3.

Dobór architektury oraz wykorzystanych podejść i narzędzi był motywowany przez konfiguracje docelowe planowane do opracowania w ramach projektu SyNaT. Wyróżniono tutaj dwie najważniejsze konfiguracje: edytora i demonstratora, współpracujące odpowiednio z systemami Protégé (edytor ontologii opracowany na uniwersytecie w Stanford, obecnie rozwijany przez uniwersytet w Manchesterze [1]) oraz Infona (platforma Yadda). Opis tych docelowych konfiguracji zawarty jest w następnym rozdziale.

3.1. Warstwowa budowa systemu CongloS

System CongloS opracowano z myślą o możliwości zastosowania jego komponentów w różnych konfiguracjach docelowych. Podstawowe założenia architektoniczne obejmują wykorzystanie podejścia warstwowego. Warstwy systemu, w różnych docelowych konfiguracjach, przedstawione są na rys 1.

Warstwa obsługi baz wiedzy zawiera mechanizmy systemu przeznaczone do obsługi tarsetowych baz wiedzy i baz wiedzy zorganizowanych według modelu SIM. Bazy te obsługiwane są w sposób kompatybilny z OWL API (patrz podrozdział 3.3). Metadane samej bazy (opis jej struktury, słownika, wykorzystywanych tarsetów) obsługiwane są w formie ontologii. Główne zadania tej warstwy to wnioskowanie z wykorzystaniem założeń algebry tarsetów i modelu SIM, obsługa zapytań do bazy oraz synchronizacja wewnętrznego stanu bazy z opisem jej metadanych. Warstwa ta zawiera komponenty kluczowe z punktu widzenia wykorzystania kontekstowych baz wiedzy.



Rys. 1. Warstwy systemu CongloS

Fig. 1. Architectural layers of CongloS system

Warstwa interfejsu użytkownika to zestaw komponentów graficznych (paneli) oferujących dostęp do najważniejszych funkcji projektowania (definiowanie struktury, tworzenie schematów baz wiedzy) oraz wykorzystywania (zadawanie zapytań) kontekstowych i tarsetowych

baz wiedzy. Komponenty graficzne w niej zawarte skonstruowane są w taki sposób, by można było wykorzystać je w jak najszerszym zakresie konfiguracji docelowych. Warstwa ta jest jednak opcjonalna i może nie być wykorzystywana w konfiguracjach docelowych, które oferują użytkownikowi inne środki dostępu do funkcji zarządzania bazami wiedzy.

Warstwa adaptacji i integracji musi zostać opracowana osobno dla każdej konfiguracji docelowej. Zawiera ona komponenty odpowiedzialne za skoordynowanie pracy komponentów niższych warstw w sposób właściwy dla danej konfiguracji. Niższe warstwy zawierają mechanizmy ułatwiające przeprowadzenie takiej integracji.

3.2. Warstwy systemu CongloS

W niniejszym podrozdziale opisano dokładniej zawartość warstw obsługi baz wiedzy (podrozdział 3.2.1) i interfejsu użytkownika (podrozdział 3.2.2) systemu CongloS.

3.2.1. Warstwa obsługi baz wiedzy

W warstwie obsługi baz wiedzy znajdują się komponenty służące do wykonywania kluczowych funkcji systemu: zarządzania zawartością oraz strukturą tarsetowych i kontekstowych baz wiedzy zorganizowanych zgodnie z modelem SIM. Komponenty zawarte w tej warstwie, a także zależności między nimi pokazane są na rys. 2. Kierunek zależności należy rozumieć zgodnie z założeniami UML (czyli np. **org.conglos.cbm** korzysta z **org.conglos.environment**).

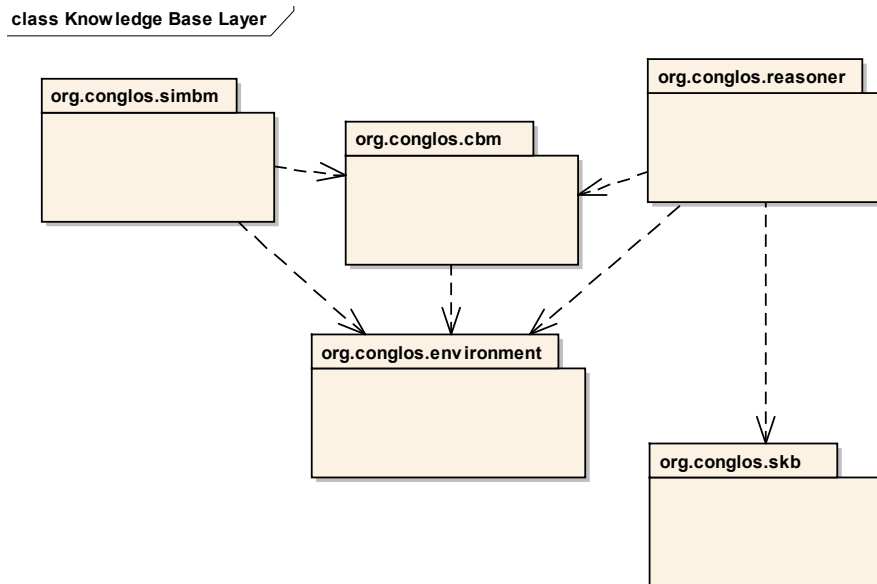
Komponenty nazwane są zgodnie z regułami obowiązującymi w języku Java (patrz podrozdział 3.3). Zadania poszczególnych komponentów opisane są poniżej.

org.conglos.skb to komponent dostarczający usług związanych z wykonywaniem operacji algebry tarsetów na poszczególnych modułach. Komponent ten nie realizuje pełnej obsługi schematów baz wiedzy; służy jedynie do obsługi instancji w zakresie wykonywania działań algebry. Oferuje także usługę przetwarzania prostych zapytań. Do poprawnego działania potrzebuje współpracy z zewnętrznym mechanizmem wnioskującym (ang. *reasoner*). Jest to związane ze specyficznym traktowaniem modułów w modelu tarsetowym, gdzie moduł może być rozumiany jako zestaw wszystkich wniosków wynikających z zawartych w nim zdań i ze sprzęgów z innymi modułami. Aby osiągnąć ten efekt, komponent **org.conglos.skb** korzysta z tzw. \mathcal{L} -(2)-reprezentacji tarsetów (patrz podrozdział 5.1).

org.conglos.cbm to komponent odpowiedzialny za obsługę schematów i instancji tarsetowych baz wiedzy. Zestaw oferowanych usług obejmuje obsługę zapisu ontologii schematu i instancji, przekazywania informacji o ich zmianie oraz koordynowania zawartości metaontologii z przechowywanymi strukturami baz wiedzy.

org.conglos.simbm odpowiada za dostarczanie usług specyficznych dla kontekstowych baz wiedzy zorganizowanych zgodnie z modelem SIM. Ponieważ zastosowane podejście

przewiduje zapis baz wiedzy SIM jako baz tarsetowych, do właściwych zadań tego komponentu należy dostarczanie usług tłumaczących (zarówno jeśli chodzi o komendy interfejsu komponentu, jak i organizację zapisu) pojęcia właściwe dla baz SIM (np. dodanie lub usunięcie typu kontekstu, czy też wystąpienia kontekstu) na odpowiednie pojęcia struktury bazy tarsetowej (dodanie lub usunięcie tarsetu bądź sprzęgu).



Rys. 2. Budowa warstwy obsługi bazy wiedzy
Fig. 2. Decomposition of Knowledge Base Layer

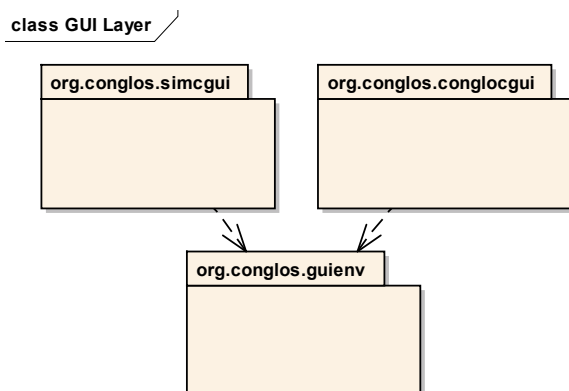
org.conglos.reasoner to komponent dostarczający usług związanych z wnioskowaniem z bazy tarsetowej. Do wnioskowania wykorzystuje usługi komponentu **org.conglos.skb**, jednak uwzględnia budowę schematu i instancji bazy odczytanych dzięki współpracy z **org.conglos.cbm**.

org.conglos.environment to komponent pełniący ważne funkcje integracyjne. Dostarcza on usług konfiguracyjnych, dostosowujących i koordynujących działanie wykorzystujących go komponentów. Do jego zadań należy m.in. dostarczanie informacji o sposobie uzyskania dostępu do zewnętrznego mechanizmu wnioskującego. Właściwe wykorzystanie tego komponentu jest kluczowe także z punktu widzenia poprawnego przygotowania warstwy adaptacji i integracji.

Wykorzystanie wszystkich komponentów warstwy obsługi baz wiedzy pozwala na przetwarzanie i modyfikowanie zapisu schematów i instancji bazy, poprawne wnioskowanie z baz SIM i baz tarsetowych oraz wykonywanie zapytań sformułowanych w algebrze tarsetów. Możliwe jest także wykorzystanie w docelowej konfiguracji jedynie niektórych, wybranych komponentów. Przykładowo, zestaw komponentów **org.conglos.cbm** i **org.conglos.environment** pozwoli na obsługę tarsetowych baz wiedzy bez wnioskowania i bez funkcji specyficznych dla kontekstowych baz wiedzy opartych na modelu SIM.

3.2.2. Warstwa interfejsu użytkownika

Warstwa interfejsu użytkownika zawiera widoki – panele – dające użytkownikowi dostęp do funkcji obsługi baz wiedzy. Panele te charakteryzują się możliwością wzajemnej współpracy oraz bardzo dużymi możliwościami konfiguracyjnymi, co sprawia, że mogą być wykorzystane w różnych środowiskach docelowych. Komponenty zawarte w warstwie interfejsu użytkownika oraz zależności między nimi zilustrowano na rys. 3.



Rys. 3. Budowa warstwy interfejsu użytkownika

Fig. 3. Decomposition of User Interface Layer

org.conglos.conglocgui to komponent dostarczający paneli służących do obsługi funkcji właściwych dla tarsetowych baz wiedzy. Przykładowe panele dostępne za pośrednictwem tego komponentu to panel definiowania sprzęgów i panel zadawania zapytań.

org.conglos.simcgui to komponent dostarczający paneli służących do obsługi kontekstowych baz wiedzy zorganizowanych zgodnie z modelem SIM. Przykładowe panele dostępne za pośrednictwem tego komponentu to panele hierarchii typów kontekstów i wystąpień kontekstów.

org.conglos.guienv jest komponentem pełniącym funkcje konfiguracyjne i integracyjne. Zawiera obiekty i interfejsy służące do wspierania przepływu informacji między panelami oraz wskazujące na cechy wizualne stylu właściwego dla odpowiedniej konfiguracji docelowej.

Podobnie jak w przypadku poprzedniej warstwy, możliwe jest wykorzystanie wybranych komponentów. Tutaj oznacza to wykorzystanie paneli właściwych jedynie dla baz tarsetowych lub SIM-owych.

3.3. Aspekty organizacyjne i technologiczne

System CongloS został zaimplementowany w języku *Java* w wersji 6.0. Wybór tego języka programowania był podyktowany wieloma czynnikami:

- duża ilość narzędzi do zarządzania ontologiami zapisanymi w logice opisowej jest zaimplementowana właśnie w Javie,
- Java zapewnia dużą przenośność wytworzonych komponentów,
- wyznaczone konfiguracje docelowe zakładają osadzenie w środowiskach wykorzystujących język Java (OWL API, OSGI, Protégé).

W ramach systemu CongloS bardzo silnie wykorzystywane jest OWL API [8], czyli biblioteka ustanawiająca standard dostępu do usług przetwarzania ontologii zapisanych w języku OWL [9]. Definiuje ona zestaw interfejsów i wzorcowych klas spełniających te interfejsy.

OWL API wykorzystywane jest w ramach systemu CongloS w trojaki sposób. Po pierwsze, jako biblioteka dostarcza usług, które potrzebne są wielu komponentom CongloS (w szczególności komponentom **org.conglos.sibm** i **org.conglos.cbm** do odczytu metaontologii i zawartości modułów). Po drugie, wybrane interfejsy OWL API zostały wykorzystane w interfejsach niektórych komponentów CongloS do przenoszenia informacji o konceptach, rolach i aksjomatach. Po trzecie, niektóre idee organizacji pracy z bazą wiedzy zostały zapożyczone z OWL API. Przykładowo, przyjęto założenie, że z bazą wiedzy można pracować bez mechanizmu wnioskującego. Użytkownik jest wówczas ograniczony do operowania wyłącznie na zapisie bazy. Mechanizm wnioskujący jest dołączany na życzenie (w ogólności można wyobrazić sobie istnienie różnych mechanizmów tego typu) i dopiero po jego dołączeniu można korzystać z kluczowych dla tarsetowych baz wiedzy przepływów wniosków zdefiniowanych za pomocą sprzęgów. Warto także wspomnieć, że organizacja zapisu tarsetowej bazy wiedzy została przeprowadzona w taki sposób, aby umożliwić częściowe wnioskowanie standardowym systemom wnioskującym (systemom „nieświadomym” sposobu organizacji bazy wiedzy).

Dla większego wsparcia komponentowej budowy systemu zdecydowano także o wykorzystaniu standardu OSGi w wersji 4.2; patrz [7]. Standard ten umożliwia większą kontrolę nad dostępem do zawartości poszczególnych komponentów. Daje także możliwość dynamicznego rozszerzania listy komponentów, co jest bardzo ważne w systemach o otwartej architekturze, w szczególności korzystających z sieciowych repozytoriów wtyczek. Obie docelowe konfiguracje są przykładem właśnie takich systemów.

4. Edytor CongloS

W niniejszym podrozdziale opisane są konfiguracje docelowe systemu CongloS, planowane do integracji w ramach projektu SyNaT. Koncentrujemy się tutaj na konfiguracji edytora zintegrowanego ze środowiskiem Protégé. Prace nad tą konfiguracją są już na ukończeniu.

Druga z planowanych konfiguracji to narzędzie dostępne przez strony WWW, służące demonstracji funkcji odpytywania bazy wiedzy; poświęcamy jej kilka słów na koniec rozdziału.

Celem opracowania konfiguracji edytora jest dostarczenie użytkownikom środków do tworzenia i edycji kontekstowych baz wiedzy opartych na modelu SIM. W konfiguracji tej komponenty systemu CongloS współpracują z popularnym edytorem ontologii Protégé (utworzonym na uniwersytecie w Stanford, obecnie rozwijanym i utrzymywanym przez uniwersytet w Manchesterze). Komponenty CongloS dostępne są w postaci dodatkowych wtyczek. Oryginalna funkcjonalność edytora Protégé nie jest więc zmieniana i może być wykorzystana do edycji zawartości poszczególnych modułów bazy wiedzy.

W ramach tej konfiguracji wyróżniamy dwa główne podsystemy: CongloS Base, obejmujący warstwę obsługi baz wiedzy, oraz CongloS Pilot, obejmujący warstwę interfejsu użytkownika oraz przygotowaną specjalnie na potrzeby tej konfiguracji docelowej warstwę adaptacji.

Warstwa adaptacji składa się zasadniczo z dwóch komponentów. Komponent **org.conglos.protege.conglodesign** służy doadaptacji widoków i usług związanych z obsługą bazy tarsetowej niezbędnych do poprawnego tworzenia baz SIM (zapytania, sprzęgi niestandardowe, wnioski), komponent **org.conglos.protege.simdesign** zaś adaptuje funkcje i widoki właściwe dla baz SIM.

Edytor CongloS daje dostęp do widoków przeznaczonych dla projektanta bazy wiedzy. Za ich pomocą można definiować hierarchie kontekstów i wystąpień kontekstów (strukturę bazy SIM). Możliwe jest też tworzenie niestandardowych sprzęgów. Utworzoną bazę można odpytywać za pomocą uproszczonego języka KQL służącego do zapisu wyrażeń algebry tarsetów.

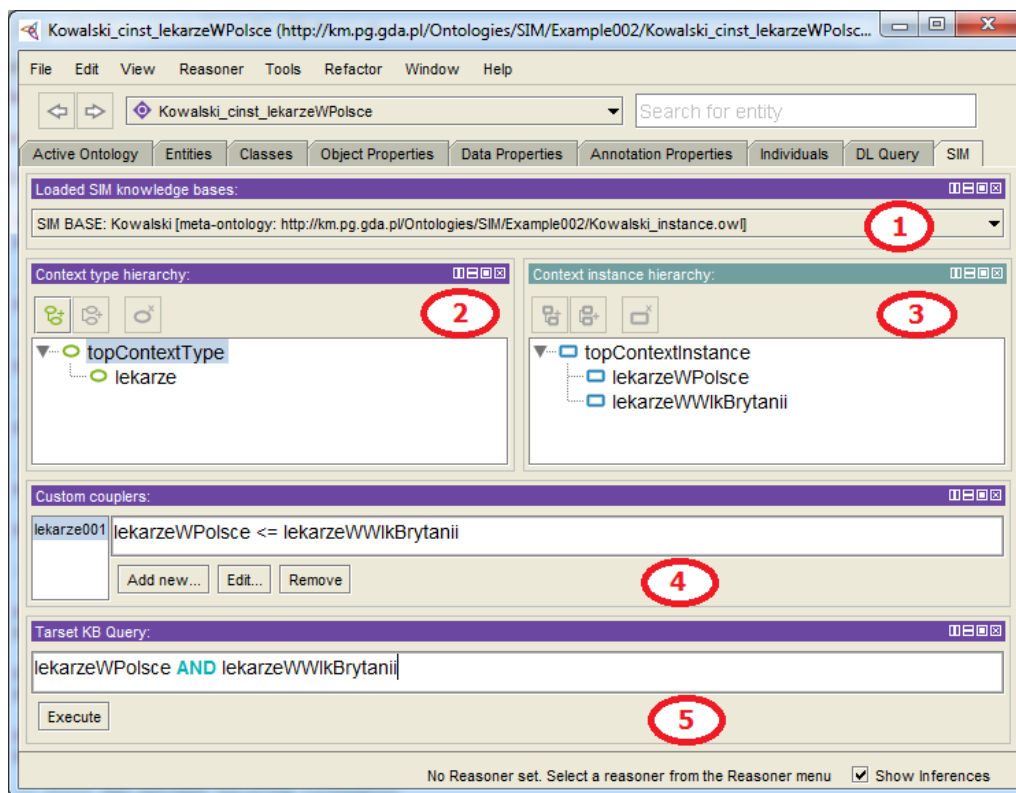
Przykładowa organizacja interfejsu użytkownika pokazana jest na rys. 4. Zgodnie z zasadami panującymi w środowisku Protégé użytkownik może ją w znacznym stopniu zmienić: widoki (panele) można przesuwac, zmieniać ich rozmiar oraz przenosić na inną zakładkę. Warstwa adaptacji zapewnia w tym zakresie współpracę widoków systemu CongloS z widokami systemu Protégé.

Pokazane na rysunku widoki służą przede wszystkim do edycji struktury bazy wiedzy. Panel (2) pokazuje hierarchię typów kontekstów, a panel (3) hierarchię wystąpień kontekstów. Panel (4) pokazuje listę ich niestandardowych sprzęgów i pozwala na definiowanie nowych.

Panel (1) pełni funkcję pomocniczą i pozwala na wybór edytowanej bazy SIM w sytuacji, gdy otwarta jest więcej niż jedna baza. Za pomocą panelu (5) możliwe jest natomiast formułowanie zapytań. Wyniki zapytania prezentowane są przez otwarcie nowego okna Protégé i pokazanie w nim wynikowego tarsetu jako ontologii złożonej z jednego modułu, zawierają-

cej wynikające z niego zdania. Warstwa adaptacji zapewnia też dostęp do mechanizmu wnioskującego właściwego dla baz SIM.

Na koniec kilka słów warto poświęcić drugiej planowanej na najbliższą przyszłość konfiguracji docelowej. Konfiguracja ta ma być dostępna przez WWW, stąd warstwa adaptacji musi zapewnić właściwe wyświetlanie widoków w tym środowisku. Planowane jest tu poszerzenie zakresu widoków o przeznaczone *stricte* do odpytywania i prezentowania zawartości bazy wiedzy w atrakcyjnej dla użytkownika formie. Widoki te po opracowaniu zostaną dołączone do konfiguracji Protégé.



Rys. 4. Przykładowa organizacja interfejsu użytkownika CongloS w edytorze Protégé
Fig. 4. Exemplary organization of CongloS user interface in Protégé

5. Rozwiązania implementacyjne

W niniejszym rozdziale krótko zaprezentowano wybrane mechanizmy wewnętrzne zaimplementowane w komponentach systemu CongloS. W podrozdziale 5.1 opisano sposób wykonywania operacji algebraicznych i wnioskowania z modułów tarsetowej bazy wiedzy, w podrozdziale 5.2 – sposób synchronizacji stanu struktur baz wiedzy ze stanem zapisu (metaontologią).

5.1. Przetwarzanie tarsetów

Mechanizmy opisane w niniejszym podrozdziale służą do przetwarzania zawartości modułów tarsetowej bazy wiedzy, w tym wnioskowania i wykonywania operacji algebraicznych, zaimplementowanych w komponencie **org.conglos.skb**.

Instancja tarsetowej bazy wiedzy składa się ze zmiennych tarsetowych (będących z grubsza odpowiednikami zmiennych relacyjnych w relacyjnej bazie danych) oraz ze sprzęgów (zapisów zależności pomiędzy zawartością poszczególnych tarsetów). Tarsety nie mają typowej zawartości; są one nieskończonymi zbiorami (lub nawet klasami w sensie matematycznym) interpretacji (rozumianych jako interpretacje Tarskiego). W czystym sensie zatem sprzęgi pełnią rolę ograniczeń zmniejszających zbiory interpretacji, zawartość tarsetu zaś może być badana jedynie przez badanie zbioru interpretacji, czyli wnioskowanie.

Teoria \mathcal{L} -2-(**D**)-reprezentowalności daje jednak narzędzia do reprezentowania zbioru interpretacji w postaci zbioru zbiorów zdań i przeprowadzania na nich odpowiednich rodzajów wnioskowania. Ze względu na brak miejsca poniżej przytoczono jedynie szkic teorii (dokładniej teoria ta jest opisana w [9]).

Dla części zastosowań wystarcza prosta reprezentacja zdaniowa tarsetów w postaci zbioru zdań: tarset jest reprezentowany przez zbiór zdań wtedy, gdy zbiór ten ma takie same interpretacje jak tarset. Jednak wykonanie niektórych operacji algebry tarsetów, w szczególności sumy, sprawia, że ten rodzaj reprezentacji przestaje wystarczać. Wówczas możliwe jest reprezentowanie tarsetu przez zbiór S zbiorów zdań – tarset zawiera sumę interpretacji wszystkich zbiorów zdań w S . Systematyczna budowa takich zbiorów pozwala na zachowanie reprezentowalności dla szerokiej gamy operacji algebraicznych ([9]).

Takie podejście pozwala też na to, by do przechowywania informacji o tarsetach (zawartości zmiennych tarsetowych) wykorzystać obiekty OWL API, a do wnioskowania odpowiednio zaadaptowane mechanizmy wnioskujące. Ich implementacja i organizacja jest podobna jak w systemie S-Pellet ([9]) – komponent **org.conglos.skb** jest jednak rozbudowany o obsługę sprzęgów oraz możliwość wykonywania zapytań (przygotowania nowego modułu na podstawie wyrażenia algebry formułowanego *ad hoc*).

5.2. Synchronizacja zapisu i struktury

Mechanizmy opisane w niniejszym podrozdziale zawarte są w komponentach **org.conglos.cbm** i **org.conglos.simbm**. Służą one do nadzoru nad zawartością metaontologii (ontologii instancji i schematu) oraz synchronizowania ich zawartości z zawartością odpowiednich obiektów opisujących strukturę bazy.

Ze względu na to, że metaontologia może zostać zmieniona przez użytkownika niezależnie od udostępnionych funkcji edycyjnych (bezpośrednio przez interfejs OWL API, np. za pomocą dodania bądź usunięcia aksjomatów), przyjęto, że wszelkie zmiany w obiektach wewnętrznie reprezentujących strukturę będą rezultatem obsługi zdarzeń zmiany ontologii schematu i instancji. W konsekwencji procedury edycyjne (np. dodania nowego sprzęgu) działają przez wprowadzenie odpowiednich aksjomatów do odpowiednich ontologii, a następnie system obsługi zdarzeń wychwytuje te zmiany i odpowiednio reaguje.

Konsekwencją tego założenia była również konieczność implementacji zaawansowanych mechanizmów odczytu metaontologii. Mechanizmy te muszą uwzględniać fakt, że użytkownik może przeprowadzić (w szczególności niedokończoną) zmianę edycyjną poza systemem CongloS (bezpośrednio w metaontologii). Aby poprawnie obsłużyć taką sytuację, wprowadzono pojęcie *maksymalnej poprawnej struktury*. Za tym pojęciem idzie następująca metoda postępowania: ze zdań zapisanych w metaontologii system odczytuje jak największy fragment, który jest w stanie zinterpretować jako poprawną strukturę bazy tarsetowej.

W celu wyznaczenia takiej struktury w komponencie **org.conglos.cbm** zaimplementowano obsługę uproszczonego algorytmu *tableaux* (najbardziej popularny algorytm wnioskowania dla ontologii zapisanych w logice opisowej). W algorytmie tym zastosowano reguły wnioskowania specyficzne dla ontologii schematu i ontologii instancji. W rezultacie algorytm ten jest zdolny do rozpoznania typowych elementów struktury bazy nawet w sytuacji braku pełnej informacji i bez zewnętrznego mechanizmu wnioskującego (którego użycie w tym przypadku okazało się nieakceptowalne ze względów wydajnościowych). Graf jest budowany na bieżąco w reakcji na zdarzenia aktualizacji ontologii.

Dodatkową kwestią wymagającą rozwiązania było udostępnienie użytkownikowi dwóch języków służących do opisanie bazy: jednego specyficznego dla baz tarsetowych, drugiego dla baz kontekstowych. Kwestia ta została rozwiązana w taki sposób, że za podstawowy został uznany zapis tarsetowy. Komponent **org.conglos.simbm** nasłuchuje zmian w metaontologii polegających na dodaniu lub usunięciu zdań opisujących strukturę SIM. Kiedy takie zmiany zostaną zidentyfikowane, są wprowadzane do komponentu **org.conglos.cbm**. Komponent **org.conglos.simbm** reaguje na nie niejako wtórnie, dostosowując się do zmian w strukturze bazy tarsetowej.

6. Podsumowanie

Stworzenie systemu CongloS to bardzo ważny krok pozwalający na przeniesienie opracowywanych teorii dotyczących baz wiedzy do praktycznych, inżynierskich środowisk informatycznych. Edytor CongloS to jedna z kluczowych konfiguracji systemu, pozwalająca użyt-

kownikowi na tworzenie własnych baz wiedzy i, częściowo, na korzystanie z nich za pomocą mechanizmów wnioskujących i zadawania zapytań.

Bardzo ważna jest jednak możliwość tworzenia nowych konfiguracji, dostosowanych do konkretnych potrzeb klas użytkowników. Zastosowana architektura daje w tym względzie znaczne możliwości, zastosowanie zaś warstwy adaptacyjnej sprawia, że elastyczność nie pociąga za sobą konieczności rezygnacji ze specyficznych udogodnień środowisk docelowych.

Zakres oferowanych funkcji będzie stopniowo powiększany wraz z przygotowywaniem kolejnych konfiguracji docelowych. Planowana na najbliższe miesiące konfiguracja jest przeznaczona dla projektu SyNaT i będzie służyć głównie prezentowaniu wyników wnioskowania w przejrzystej i atrakcyjnej formie.

BIBLIOGRAFIA

1. <http://protege.stanford.edu/>, The Protégé Ontology Editor and Knowledge Acquisition System, data dostępu: grudzień 2012.
2. Goczyła K., Waloszek A., Waloszek W.: Towards context-semantic knowledge bases, FedCSIS 2012.
3. Goczyła K., Waloszek A., Waloszek W.: Hierarchiczny podział przestrzeni ontologii na konteksty. Bazy danych. Nowe technologie, WKŁ, Warszawa 2007, s. 247÷260.
4. Goczyła K., Waloszek A., Waloszek W.: Algebra konglomeratów jako narzędzie opisu problemów przetwarzania ontologii. *Studia Informatica*, Vol. 30, No. 2A (83), 2009, s. 141÷156.
5. Goczyła K., Waloszek A., Waloszek W., Zawadzka T.: Odwzorowania międzyontologiczne w algebrze konglomeratów. *Studia Informatica*, Vol. 33, No. 2A (105), 2012, s. 329÷346.
6. Goczyła K., Waloszek A., Waloszek W., Zawadzka T.: Schemat konglomeratowej bazy wiedzy. *Studia Informatica*, Vol. 33, No. 2A (105), 2012, s. 347÷366.
7. OSGi Service Platform Core Specification, Release 4, Version 4.2, OSGi Alliance, czerwiec 2009.
8. Horridge M., Bechhofer S.: The OWL API: A Java API for OWL Ontologies. *Semantic Web Journal* 2(1), Special Issue on Semantic Web Tools and Systems, 2011, s. 11÷21.
9. Waloszek W.: S-Pellet jako implementacja zdaniowej reprezentacji modułów algebry konglomeratów. *Studia Informatica*, Vol. 33, No. 2A (105), 2012, s. 311÷328.

Wpłynęło do Redakcji 15 stycznia 2013 r.

Abstract

The article presents CongloS – a management system for designing, building and exploiting contextual knowledge bases.

The approach supported by ColgoS is based on two main methods proposed by the authors of the paper. At the conceptual level, the SIM method (proposed elsewhere) is used for design of a knowledge base: SIM organizes a knowledge base into hierarchically ordered contexts. The contextualization of a base is perceived as a part of conceptualization process, but it also influences the way of modularization of a base. At the implementation level, the taret approach is used: taretsets are semantically defined knowledge base modules that can be efficiently combined in various ways taretset algebra expressions. The same algebra is also used to define relationships (called couplers) between taretsets in a knowledge base.

Currently the main taretset configuration of CongloS is an editor integrated with the well-known Protégé ontology editor. In this configuration CongloS is packaged as a set of plug-ins and offers fully integrated views with functions for defining the structure of a base (standard: contexts, and non-standard: with additional couplers), reasoning, and for querying a base with *ad hoc* formulated taretset algebra expressions.

Though the configuration is very important for knowledge base designers and for dissemination of CongloS approach, it has to be stressed that the architectural approach to building CongloS system allows it to be adapted to various taretset configurations (with selected relevant functions). A configuration currently being developed is intended for presenting fragments of a knowledge base and query results via WWW interface.

Adresy

Krzysztof GOCZYŁA: Politechnika Gdańska, Wydział Elektroniki, Telekomunikacji i Informatyki, ul. Narutowicza 11/12, 80-233 Gdańsk, Polska, kris@eti.pg.gda.pl.

Aleksander WALOSZEK: Politechnika Gdańska, Wydział Elektroniki, Telekomunikacji i Informatyki, ul. Narutowicza 11/12, 80-233 Gdańsk, Polska, alwal@eti.pg.gda.pl.

Wojciech WALOSZEK: Politechnika Gdańska, Wydział Elektroniki, Telekomunikacji i Informatyki, ul. Narutowicza 11/12, 80-233 Gdańsk, Polska, wowal@eti.pg.gda.pl.