

Prosecco: Procesowo-semantyczne zarządzanie przedsiębiorstwem, metody i narzędzia

Raport końcowy z zadania 2

Raport końcowy z zadania 2: *Opracowanie modelu taksonomii pojęć* opracowany w ramach projektu

Procesowo-semantyczne zarządzanie przedsiębiorstwem, metody i narzędzia
nr PBS1/B3/14/2012 realizowanego w ramach 1. Konkursu Programu Badań Stosowanych,
finansowanego przez Narodowe Centrum Badań i Rozwoju



Tytuł zadania	Opracowanie modelu taksonomii pojęć
Numer zadania	2
Okres realizacji zadania	od Grudzień 2012 do Wrzesień 2013
Wersja dokumentu	1.0

Spis treści

1	Cel zadania	5
2	Metodologia badawcza	6
2.1	Przegląd oraz porównanie otwartych systemów CRM i DMS pod kątem uchwycenia wspólnych schematów pojęciowych	6
2.2	Analiza dostępnych metodologii pozyskiwania wiedzy do budowy modelu taksonomicznego	15
2.2.1	Podstawowe metodyki tworzenia ontologii	16
2.2.2	Pozyskiwanie wiedzy i kierunki budowy taksonomii	18
2.3	Ewaluacja istniejących ontologii przedsiębiorstw	19
2.3.1	Ontologia organizacji ORG	20
2.3.2	Ontologia organizacji IntelLEO	21
2.3.3	Ontologia organizacji GZ	26
2.3.4	Ontologia PROTON	27
2.3.5	Inne ontologie	27
2.4	Analiza znanych ontologii biznesowych BPMN oraz CRM	29
3	Rezultaty	34
3.1	Specyfikacja wymagań dla modelu taksonomii pojęć	35
3.2	Proces pozyskiwania wiedzy	43
3.2.1	Spotkania analityczne	45
3.2.2	Analiza otrzymanych materiałów i procesów biznesowych pod kątem taksonomii	45
3.2.3	Prototypy taksonomii firm	46
3.2.4	Narzędzia reprezentacji wiedzy	47
3.3	Model taksonomii pojęć	51

3.3.1	Analiza opracowanych szkieletów taksonomii	51
3.3.2	Założenia dotyczące semantycznego modelu ontologicznego	52
3.3.3	Semantyczny model ontologiczny – ontologia Prosecco	54
3.3.4	Taksonomiczny model pojęć	56
3.3.5	Przykład zastosowania taksonomii w procesach	72
4	Efekty zadania	77
5	Załączniki	78
5.1	Prototyp taksonomii PromoAgency	78
5.2	Prototyp taksonomii IQON Consulting	85
5.3	Prototyp taksonomii SimplyUser	93

Streszczenie

Niniejszy raport podsumowuje prace przeprowadzone w ramach zadania 2 projektu Prosecco. Raport ilustruje zastosowaną metodologię badawczą oraz jej wyniki, jak również rezultaty i efekty zadania. W ramach prac dokonano analizy istniejących rozwiązań w zakresie systemów CRM i DMS; przeprowadzono wywiady z pracownikami i klientami małych i średnich przedsiębiorstw; przeanalizowano dostępne metodologie pozyskiwania wiedzy oraz poddano ewaluacji istniejące ontologie przeznaczone dla przedsiębiorstw, jak również systemów CRM i BPMN. W wyniku przeprowadzonych analiz powstała specyfikacja wymagań dla opracowanego modelu taksonomicznego pojęć; autorski proces pozyskiwania wiedzy; oraz model taksonomii pojęć.

1. Cel zadania

Celem zadania 2 było wypracowanie całościowego zhierarchizowanego modelu ontologicznego dla małych i średnich przedsiębiorstw pozwalającego na wprowadzenie taksonomii pojęć występujących w procesach. Realizacja zadania wymagała podjęcia następujących działań:

- zidentyfikowanie realnych potrzeb małych i średnich przedsiębiorstw przeprowadzając ustrukturalizowane wywiady,
- wyspecyfikowanie wymagań dla modelu taksonomii pojęć na podstawie wcześniejszych wywiadów,
- przeanalizowanie aktualnego stanu wiedzy w zakresie modelowania ontologii przeznaczonych dla przedsiębiorstw,
- wypracowanie metodologii procesu pozyskiwania wiedzy bazującego na ustrukturalizowanym opisie tekstowym,
- opracowanie modelu taksonomii pojęć na podstawie analizy wcześniej uzyskanych wyników uwzględniając rzeczywiste potrzeby małych i średnich przedsiębiorstw.

Wynikiem końcowym zadania jest opracowany taksonomiczny model pojęć, który pozwala na ujednoczenie i ujednoznaczenie pojęć używanych w systemie zarządzania małymi i średnimi przedsiębiorstwami. Przyczyni się to do uproszczenia procesu zarządzania a tym samym obniży jego koszt. Semantyczny model ontologiczny będzie rozwijany i wykorzystywany w kolejnych zadaniach projektu Prosecco.

2. Metodologia badawcza

W zadaniu 2 analitycy PP i AGH przeprowadzili: ustrukturalizowane wywiady z pracownikami STH i ich klientami z wybranych MŚP, a następnie pogłębili analizę, której celem było pozyskanie wiedzy o hierarchii istotnych pojęć wykorzystywanych w praktykach biznesowych.

Zidentyfikowano i uporządkowano pojęcia pod kątem ich zgodności z wcześniej zidentyfikowanymi i zamodelowanymi procesami biznesowymi (w ramach zadania 1). Otrzymano model o charakterze hierarchicznym i zmodularyzowanym. Obejmuje on opracowanie taksonomii dziedzinowych w niższej warstwie, dla poszczególnych części funkcjonalnych systemu. W warstwie wyższej dostarcza taksonomii wysoko-poziomowej pozwalającej na wskazanie pojęć wspólnych dla wszystkich taksonomii i kluczowych dla całego zadania zarządzania w wybranych jego aspektach. Do praktycznej realizacji zadania będą wykorzystywano standardowe i swobodnie dostępne narzędzia do modelowania taksonomii, np. Protégé¹.

Kolejne podrozdziały zawierają kluczowe elementy przeprowadzonych prac badawczych, które polegały na zaznajomieniu się z aktualnym stanem wiedzy (ang. state-of-the-art) oraz na analizie dostępnych narzędzi.

2.1. Przegląd oraz porównanie otwartych systemów CRM i DMS pod kątem uchwycenia wspólnych schematów pojęciowych

Prace w ramach zadania 2 rozpoczęto od identyfikacji typowych zadań i funkcji realizowanych przez istniejące na rynku rozwiązania. Wykonano dość szczegółowy przegląd istniejących na rynku rozwiązań na podstawie których dokonano wyodrębnienia typowych dla systemów DMS i CRM funkcjonalności.

Kompleksowe rozwiązanie wspomagające zarządzanie relacjami z klientami powinno

¹<http://protege.stanford.edu/>

zapewnić wspomaganie wszystkich faz kontaktu klienta z firmą, począwszy od rozpoznania jego potrzeb i identyfikację, poprzez zawarcie transakcji, na usłudze posprzedażnej skończywszy.

Klasyczny CRM składa się z trzech części:

- interaktywnego - ma zapewnić uniezależnienia standardu obsługi klienta od kanału komunikacji. Obejmuje zagadnienie koordynacji i zarządzania dostępem do kanałów kontaktu z klientem, dostawcami i partnerami biznesowymi. Stosowane środki komunikacji to telefon, SMS, WEB, e-mail, faks, aplikacje głosowe czy nawet poczta tradycyjna.
- operacyjnego – odpowiada za gromadzenie i udostępnianie informacji o kliencie i produktach.
- analitycznego - zajmującego się dokonywaniem wszechstronnych analiz danych o klientach, takich jak np.: wielowymiarowa segmentacja klientów, analiza wartości klientów, analiza lojalności.

Podstawowe obszary działania firmy wspierane przez większość istniejących na rynku aplikacji klasy CRM to:

1. Obsługa kontaktu z klientami: integracja różnych kanałów kontaktu z klientami, automatyzacja procedur obsługi klienta, współpraca z Call Center.
2. Analizy biznesowej i raportowania: analizy zyskowności klientów, kanałów dystrybucji, analizy struktury i dynamiki bazy klientów.
3. Marketingu: automatyzacja obsługi kampanii marketingowych.
4. Sprzedaży: integracja i zarządzanie kanałami sprzedaży.
5. Obsługi posprzedażowej.

6. Zarządzanie bazą klientów i dostawców

W ramach zidentyfikowanych obszarów wyróżnić można następujące grupy wymagań i zadań stawianych przed tego typu aplikacjami:

- Zarządzanie strukturą organizacyjną sieci sprzedaży:
 - Możliwość zdefiniowania struktury organizacyjnej sprzedaży
 - Możliwość zdefiniowania struktury stanowisk dla komórek organizacyjnych występujących w sieci sprzedaży
 - Definiowanie ról w strukturze oraz w procesach sprzedaży
 - Możliwość definiowania rutynowych zadań i obowiązków dla poszczególnych ról i stanowisk
 - Umożliwienie dostępu do danych gromadzonych w systemie dla podmiotów zewnętrznych poprzez zdefiniowanie odpowiednich uprawnień
- Zarządzanie bazą klientów - Kompletna wiedza o kontrahentach:
 - agregacja danych o Kliencie - tzw. widok 360
 - Rejestracja oraz przechowywanie danych o klientach (dane podstawowe i kontaktowe, typy: klient indywidualny, pracodawca klienta, klient instytucjonalny, przedstawiciele reprezentujący klienta, ...).
 - Segmentacja klientów
 - Ewidencja, historia i zakres kontaktów z klientem, podejmowane działania sprzedażowe oraz w okresie obsługi posprzedażowej
 - Dane o produktach klienta (produkty rzeczywiste, produkty potencjalne)
 - Portfel moich klientów
 - Grupowanie klientów w teczki

- Statusy klientów (stali klienci, nowi klienci, kontakty kwalifikowane, kontakty niekwalifikowane, utraceni klienci)
- Gromadzenie informacji o klientach potencjalnych
- Ankiety dla klienta (wyniki pojedynczej ankiety klienta, wyniki zbiorcze)
- Zarządzanie procesem sprzedaży i reklamacji:
 - Możliwość definiowania i zmiany procesów predefiniowanych
 - Możliwość stworzenia nowych procesów sprzedaży i obsługi klienta
 - Możliwość definiowania reguł obsługi zdarzeń sprzedażowych przez wprowadzanie np. priorytetów
 - Śledzenie okazji sprzedażowych, poprzez pełen cykl sprzedaży
 - Użycie mierzalnego procesu sprzedaży
 - Zarządzanie budżetami sprzedaży
 - Kontrola jakości zamówienia przy użyciu list kontrolnych
 - Obsługa posprzedażna, zgłoszenia serwisowe, reklamacje
 - Produkty skojarzone, cross-selling
 - Wsparcie zarządzania procesami
 - Automatyczne powiadomienia
 - Definiowanie procedur postępowania
- Zarządzanie bazą produktów:
 - Definiowanie dowolnej kategorii produktów
 - Analiza szczegółów zmian w zakresie oferty i sprzedaży na przestrzeni czasu
 - Możliwość generowania ofert i umów na podstawie bazy produktów

- Porównywarka produktów umożliwiająca zestawienie najważniejszych cech produktów
- Wydruk karty informacyjnej produktu
- Operacyjne zarządzanie pracą własną i współpracowników w procesie sprzedaży:
 - Delegowanie zadań i kontrola ich realizacji
 - Automatyczne informowanie o zadaniach przydzielonych do użytkownika
 - Mechanizm przypomnień o zadaniach do wykonania
 - Użycie kalendarza współdzielonego z innymi osobami
 - Zarządzanie kalendarzami współdzielonych zasobów
- Przeprowadzenie akcji marketingowych:
 - Przeprowadzanie akcji marketingowych z określonym workflow wykonywanych działań
 - Możliwość importu danych z zewnętrznych źródeł danych
 - Analiza grupy odbiorców kampanii – liczebności grupy, obszaru, z którego grupa się wywodzi itp.
 - Przetworzenie danych klientów według określonych algorytmów w celu wyselekcjonowania segmentów lub grup klientów celem zaoferowania dedykowanego produktu/usługi
 - Automatyczne rozdzielanie zadań w ramach akcji
 - Kontrola wykonania zadań zleconych w ramach akcji marketingowych
 - Statusy prowadzonych akcji marketingowych i aktywności sprzedażowych
 - Korespondencja seryjna do klientów objętych wybraną kampanią

- Współdzielenie wiedzy firmowej
- Sprawozdawczość operacyjna, analityczna i zarządcza:
 - Możliwość monitorowania pracy sprzedawców
 - Szczegółowe zestawienie z działań handlowych i komunikacji
 - Możliwość tworzenia własnych raportów
- Integracja z innymi aplikacjami
 - Integracja z programem pocztowym
 - Synchronizacja z telefonami
 - Integracja z innymi systemami
 - Wymiana danych ze stroną www lub aplikacjami webowymi
 - Integracja z centralą telefoniczną lub systemem call center

Wynikiem przeglądu i analizy istniejących rozwiązań i wymagań dla tego typu systemów było przygotowanie szczegółowej Listy Kontrolnej pozwalającej na precyzyjne określenie oczekiwanych wymagań. Lista została umieszczona w systemie DokuWiki firmy Softhis.

W analogiczny sposób dokonano przeglądu istniejących rozwiązań dla systemów klasy DMS. Wynikiem przeglądu było powstanie listy wymagań funkcjonalnych typowego systemu klasy DMS. Obejmuje ona następujące zagadnienia:

1. Zarządzanie procesami biznesowymi

System powinien zapewniać intuicyjne definiowanie procesów workflow(np. za pośrednictwem diagramów) z wykorzystaniem wcześniej zdefiniowanej struktury przedsiębiorstwa, pozwalając na ich bieżące dostosowywanie przez pracowników (lub innego użytkownika) odpowiednio do zmieniających się potrzeb

przedsiębiorstwa. Wizualizacja przebiegu procesu powinna umożliwiać intuicyjne zorientowanie się, na którym etapie procesu aktualnie znajduje się zadanie i jaka ścieżka jeszcze przed nim. Rozszerzeniem funkcjonalności jest wprowadzenie możliwości powiązania dokumentów z procesami biznesowymi zdefiniowanymi w module zarządzania procesami biznesowymi.

2. Zarządzanie obiegiem dokumentów

System powinien umożliwiać określenie odpowiedniej ścieżki obiegu każdego typu i rodzaju dokumentu. Ścieżka ta może być dowolnie modelowana przez firmę, w zależności od potrzeb i wypracowanych procedur zakładając wiele pozycji akceptacyjnych. Powinna istnieć możliwość wielopoziomowej kontroli stanu realizacji zadań (spraw), raportowania stanu poszczególnych zadań, w każdej chwili można zlokalizować nie tylko sam dokument, ale i określić status prac z nim związanych. Typowe klasy dokumentów stosowanych w firmach do których zarządzanie obiegiem dokumentu znajduje zastosowanie:

- faktury,
- zaliczki,
- delegacje,
- zamówienia,
- wnioski urlopowe,
- wnioski premiowe,
- intranet pracowniczy,
- korespondencja zewnętrzna i wewnętrzna,
- zlecenia oraz proces ofertowania,
- umowy i aneksy,
- zgłoszenia serwisowe i reklamacyjne,

- protokoły,

3. Archiwizacja dokumentów

System powinien umożliwiać przechowywanie i rejestracja e-dokumentacji w nieograniczonych ilościach. System DMS powinien tworzyć bezpieczne archiwum dokumentów, posiadające szereg zabezpieczeń umożliwiających dostęp do dokumentów tylko osobom upoważnionym. Łatwość i szybkość odszukania archiwizowanego dokumentu oraz wgląd w jego wszystkie wersje ułatwia i znacznie skraca czas pracy. Dostarczane archiwum dokumentów powinno pozwalać przechowywać pliki w każdej formie, a jego zawartość może być udostępniana również bezpośrednio na potrzeby aplikacji trzecich np.: systemów klasy ERP, finansowo-księgowych, kontrolingowych, magazynowych, czy typowych dla danej branży systemów dziedzinowych.

4. Rozpoznawanie treści dokumentów OCR

System DMS powinien umożliwiać automatyczne uzupełnianie wartości indeksów odczytanych z obrazu dokumentu. Funkcjonalność przyspiesza proces wprowadzania danych o dokumencie analizując jego zeskanowany obraz i uzupełniając odczytanymi wartościami pola opisujące ten dokument np. NIP, nr rachunku bankowego, data wystawienia faktury, itp. Rozwiązanie pobiera dane z dokumentu i automatycznie zaczytuje je do systemu sprawdzając poprawność sczytanych danych w trakcie procesu walidacji. Automatyczna detekcja pól nagłówkowych

5. Zarządzanie dokumentacją projektową

System DMS powinien wspomagać zarządzanie dokumentacją projektową. Funkcjonalność ta jest dedykowana firmom, dla których niezbędnym narzędziem w codziennej pracy jest rozwiązanie umożliwiające zarządzanie dokumentami oraz śledzenie zmian w dokumentach (projekty, umowy, dokumentacja prze-

targowa). Narzędzie poprawia i efektywnie usprawnia komunikację w obrębie złożonych projektów. Dzięki takiemu rozwiązaniu, wszyscy uczestnicy przedsięwzięcia mogą z dowolnego miejsca zatwierdzać dokumenty, mają dostęp do najnowszych wersji oraz mogą informować o postępie prac swoich współpracowników. Zarządzanie dokumentacją projektową powinno umożliwiać między innymi:

- wsparcie dla wersjonowania plików (historia zmian /każda wersja pliku posiada informacje o autorze, dacie modyfikacji oraz komentarz dot. zmiany/, możliwość podglądu i/lub przywrócenia dowolnej poprzedniej wersji),
- obsługę różnorodnych formatów plików (m.in. Word, Excel, PDF, CAD i inne),
- automatyczne powiadamianie wszystkich osób zaangażowanych w projekt, o dokonanych zmianach w plikach,
- zapewnienie kontroli nad jednoczesną pracą kilku osób na tym samym pliku,(zapewnienie, że praca żadnej osoby nie zostanie nadpisana i utraczona),
- definiowanie uprawnień do wybranych katalogów (branż projektowych) dla grup roboczych (prawo do odczytu, prawo do wprowadzania zmian, prawo do akceptacji zmian),
- akceptację zmian oraz tworzenie procesu akceptacyjnego.

6. Generowanie szablonów dokumentów

System powinien posiadać wbudowane narzędzia umożliwiające tworzenie w prosty sposób szablonów dla dowolnego typu dokumentów np. wniosek urlopowy, zamówienie, zapotrzebowanie. Generowanie dokumentów z wcześniej zdefiniowanych szablonów może wykorzystywać, poza informacjami wprowadzonymi przez użytkownika, różne źródła dodatkowych danych. Sporządzając

zamówienie użytkownik może mieć do dyspozycji zaimplementowaną w systemie bazę artykułów. Funkcjonalność generowania szablonów powinna pozwalać również na kreowanie wyglądu graficznego dokumentu celem jego prezentacji w systemie

2.2. Analiza dostępnych metodologii pozyskiwania wiedzy do budowy modelu taksonomicznego

Taksonomia będąca elementem docelowego systemu zarządzania przedsiębiorstwem, musi uwzględniać kluczowe pojęcia i relacje występujące w firmie, z drugiej zaś strony integrować się z opisem dynamiki przedsiębiorstwa, wyrażonej za pomocą procesów i reguł biznesowych. Na podstawie ustrukturalizowanych wywiadów z firmami należy zidentyfikować pojęcia kluczowe dla funkcjonowania firmy, potrzebne do stworzenia modelu taksonomicznego.

W ramach obszaru inżynierii ontologii (ang. *ontology engineering*) nie wypracowano jednego obowiązującego modelu [2]. Istnieje kilka podejść, w szczególności opracowano szereg metodyk. Jednak przegląd [1] z 2010 roku pokazuje, że:

- Istnieje niewiele wyszczególnionych, zdefiniowanych metodologii tworzenia ontologii a te, które istnieją są stosunkowo stare.
- Istniejące metodyki można podzielić na iteracyjne i przyrostowe oraz te zorientowane na kompleksowy proces tworzenia systemu opartego na wiedzy.
- W większości metodyk wyróżnia się te same podstawowe etapy, od oceny do wdrożenia, testowania i poprawek.
- Aktualne metodyki są dość ograniczone i z wyjątkiem tych opartych na UML w większości wydają się nie spełniać standardów korporacyjnych.
- W większości opracowań brak dyskusji na temat wsparcia narzędziowego.

- Brak nowych rozwiązań wskazuje na zmniejszenie aktywności naukowej w tej dziedzinie.

Należy zatem przeanalizować istniejące metody i dobre praktyki dotyczących tworzenia taksonomii i pozyskiwania wiedzy i zaadaptować je do projektu.

2.2.1. Podstawowe metodyki tworzenia ontologii

W ramach zadania dokonano przeglądu najważniejszych i najlepiej ugruntowanych metodologii tworzenia i zarządzania ontologiami [1, 2]. Przedstawione poniżej metodyki obejmują całość procesu tworzenia i zarządzania ontologią (w szczególności taksonomią). Podstawowe etapy w tych metodykach to:

TOVE

1. *motivating scenarios*: punkt wyjścia, przykłady problemów, które ma rozwiązać ontologia/taksonomia,
2. *informal competency questions*: wymagania stworzone na podstawie ww. scenariuszy definiujące na jakie pytania powinna odpowiadać ontologia (jakie zadania powinny być możliwe do wykonania),
3. *terminology specification*: formalna specyfikacja pojęć, obiektów i relacji w logice formalnej,
4. *formal competency questions*: formalna specyfikacja pytań, na które powinna odpowiedzieć ontologia,
5. *axiom specification*: zapis aksjomatów ontologii/taksonomii uwzględniający wcześniejsze etapy,
6. *completeness theorems*: ewaluacja ontologii

Enterprise

1. *identify purpose*: określenie poziomu formalizacji ontologii,
2. *identify scope*: specyfikacja obejmująca zakres informacji, które ontologia ma zamodelować, które podyktowany jest celem, w jakim ma być wykorzystana,
3. *formalisation*: zapis formalny wg specyfikacji,
4. *formal evaluation*: ewaluacja ontologii.

METHONTOLOGY

1. *specification*: identyfikacja celu, poziomu formalizacji, zakresu, użytkowników. Efetem tej fazy jest słowny opis-specyfikacja,
2. *knowledge acquisition*: pozyskanie wiedzy: nie określono dokładnych wytycznych, aczkolwiek sugerowane metody to rozmowy z ekspertami i analiza tekstów z dziedziny,
3. *conceptualisation*: konceptualizacja pojęć, relacji i obiektów dziedziny z użyciem nieformalnej reprezentacji,
4. *integration*: wykorzystanie definicji z istniejących ontologii,
5. *implementation*: zapis formalny ontologii,
6. *evaluation*: walidacja i weryfikacja, wykrywanie niespójności, redundancji i niekompletności,
7. *documentation*: stworzenie dokumentacji.

Oprócz kompleksowych metodyk z dziedziny *ontology engineering* istnieją również metody i wypracowane dobre praktyki w zakresie poszczególnych etapów zarządzania ontologią, m.in. Ontolingua, CommonKADS, KACTUS, PLINIUS, ONIONS,

Mikrokosmos, MENELASPHYSSYS, SENSUS (por. [2]). Można w nich znaleźć np. wskazówki jak tworzyć ontologie/taksonomie na podstawie już istniejących, jak integrować wiedzę z różnych źródeł, jakie heurystyki pomagają uzyskać odpowiedni poziom szczegółowości, jakie postulaty powinna spełniać taksonomia.

2.2.2. Pozyskiwanie wiedzy i kierunki budowy taksonomii

Pozyskiwanie wiedzy do ontologii może się odbywać poprzez metody automatyczne, rozmowy z ekspertami, a nawet w procesie tzw. burzy mózgów [2]. W przeanalizowanych metodykach i dobrych praktykach tworzenia ontologii powtarza się postulat oddzielenia części nieformalnej i formalnej tworzenia ontologii. Niektóre metodologie proponują wprowadzenie *reprezentacji pośredniej* będącej pomostem między nieustrukturalizowanym opisem słownym dziedziny, a zbiorem formalnych aksjomatów. Taka pośrednia reprezentacja może używać ustrukturalizowanego (ograniczonego) języka naturalnego lub być prostym językiem graficznym (jak w metodyce IDEF5).

Podczas tworzenia taksonomii można realizować podejście *top-down* polegające na zidentyfikowaniu i zdefiniowaniu najpierw pojęć najbardziej ogólnych, a następnie ich stopniowemu uszczegółowieniu, *bottom-up*, w którym stopniowo uogólnia się zidentyfikowane na niskim poziomie abstrakcji pojęcia i relacje, a także podejście *middle-out*, w którym najpierw identyfikuje się pojęcia najistotniejsze i zgodnie z potrzebami uogólnia się je lub uszczegółowia [3]. Wadą podejścia bottom-up jest ryzyko wystąpienia niespójności, zbyt dużego narzutu pracy oraz trudności w dostrzeżeniu wspólnych części zidentyfikowanych obiektów. Z kolei podejście top-down obarczone jest niebezpieczeństwem, że arbitralnie narzucone kategorie wysokiego poziomu nie będą przystawać do faktycznych warunków i w konsekwencji konieczne będzie poprawianie taksonomii.

Podsumowując przegląd metodologii można stwierdzić że:

1. Nie istnieje gotowe rozwiązanie uznane za najlepsze, dobór metody i narzędzi

zależy od przypadku użycia.

2. Podstawowe etapy i zagadnienia powtarzają się w różnych metodykach.
3. Tworzenie ontologii jest procesem przyrostowym, większość metodyk proponuje iteracyjne ulepszanie modelu.
4. Pojęcia w ontologii powinny być stosunkowo bliskie realnym obiektom (fizycznym lub logicznym) oraz relacjom w modelowanej dziedzinie. W praktyce przekłada się to na utworzenie zbioru rzeczowników i czasowników. Nie ma jednak jasnych wytycznych w jaki sposób identyfikować istotne pojęcia.
5. W zależności od sytuacji można do budowy taksonomii (ontologii) użyć podejścia top-down, bottom-up lub middle-out. Każde z podejść ma swoje zalety oraz wady, warto świadomie dobrać metodę do przypadku użycia.

2.3. Ewaluacja istniejących ontologii przedsiębiorstw

Zgodnie z koncepcją realizacji projektu Prosecco, wymaga się semantycznego opisu nie tylko dotyczącego modelowania procesów, ale również przedsiębiorstw. W ramach prac projektowych w zadaniu 2 przeanalizowano następujące ontologie dla przedsiębiorstw:

1. An organization ontology [4] – ontologia rozwijana przez konsorcjum W3C oraz firmę Epimorphics Ltd².
2. IntelLEO Organization Ontology [5] – ontologia powstała w ramach projektu Intelligent Learning Extended Organisations (IntelLEO).
3. Ontology for organizations [6] – ontologia jest częścią większej Gazette [7] ontologii służącej do oznaczania treści The London Gazette [8].

²<http://www.epimorphics.com/>

4. PROTON (PROTo ONtology) [9] – ontologia wyższego rzędu (upper-level ontology) służąca do modelowania zarządzania dokumentami w organizacji.

Ontologie 1, 2, 3 oraz 4 są publicznie dostępne. Zostały szczegółowo przeanalizowane i porównane z dotychczasowym szkieletem ontologii *onto-orb* oraz *onto-sth*.

2.3.1. Ontologia organizacji ORG

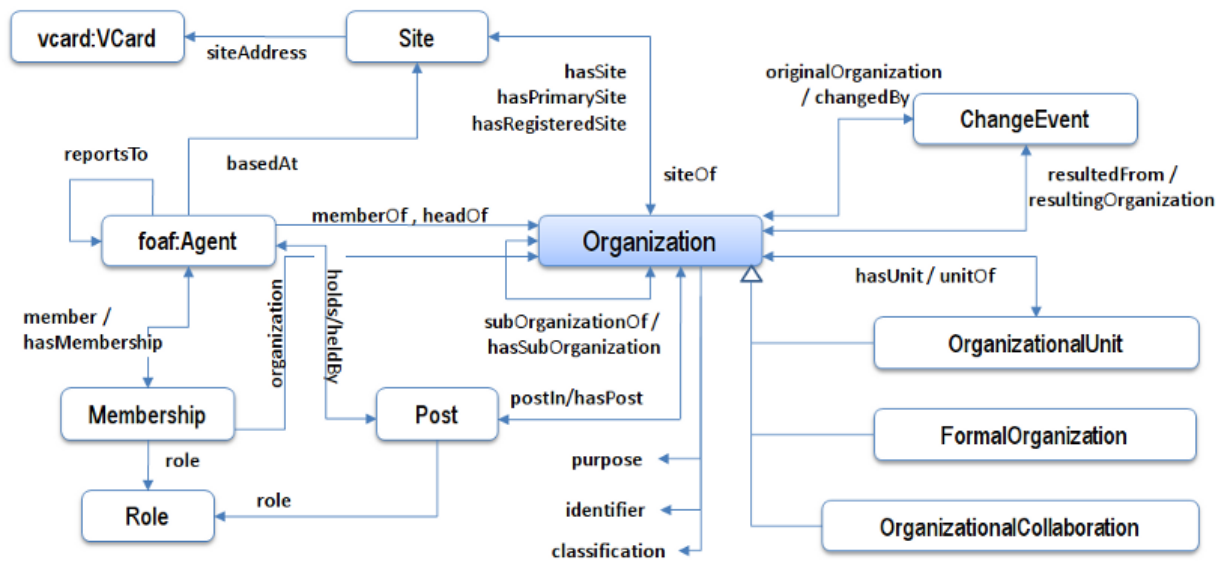
Ontologia organizacji ORG zdefiniowana została w języku *SH(D)*. Zawiera pojęcia oraz role typu *ObjectProperties* i *DatatypeProperties*. Celem tej ontologii jest zdefiniowanie podstawowych struktur organizacyjnych, umożliwiających publikowanie danych dotyczących organizacji działających w różnych dziedzinach. Dużym plusem ontologii jest jej prostota oraz możliwość rozszerzenia o pojęcia i relacje specyficzne dla konkretnej domeny. Użyte narzędzia wnioskujące (Pellet, Fact++ i Hermit) nie zgłaszają żadnych zastrzeżeń co do spójności ontologii. Za pomocą ontologii ORG można opisać następujące elementy:

- a) strukturę organizacji (firmy) składającą się z wielu jednostek zależnych (oznaczane relacją *hasSubOrganization/subOrganizationOf* i *hasUnit/unitOf*),
- b) powiązanie ludzie z organizacją (relacja *memberOf*, *headOf* itd.),
- c) informacje o lokalizacji danej jednostki (relacje *hasSite*, *location*),
- d) informacje historyczne opisujące zmiany strukturalne w organizacji (np. zmiana nazwy wyrażona za pomocą relacji *ChangeEvent*).

Idea ontologii ORG obrazująca najważniejsze pojęcia i role została przedstawiona na Rysunku 1.

Rysunki 2-4 pochodzące z narzędzia Protégé przedstawiają ontologię ORG.

Główne zalety ontologii ORG są następujące:



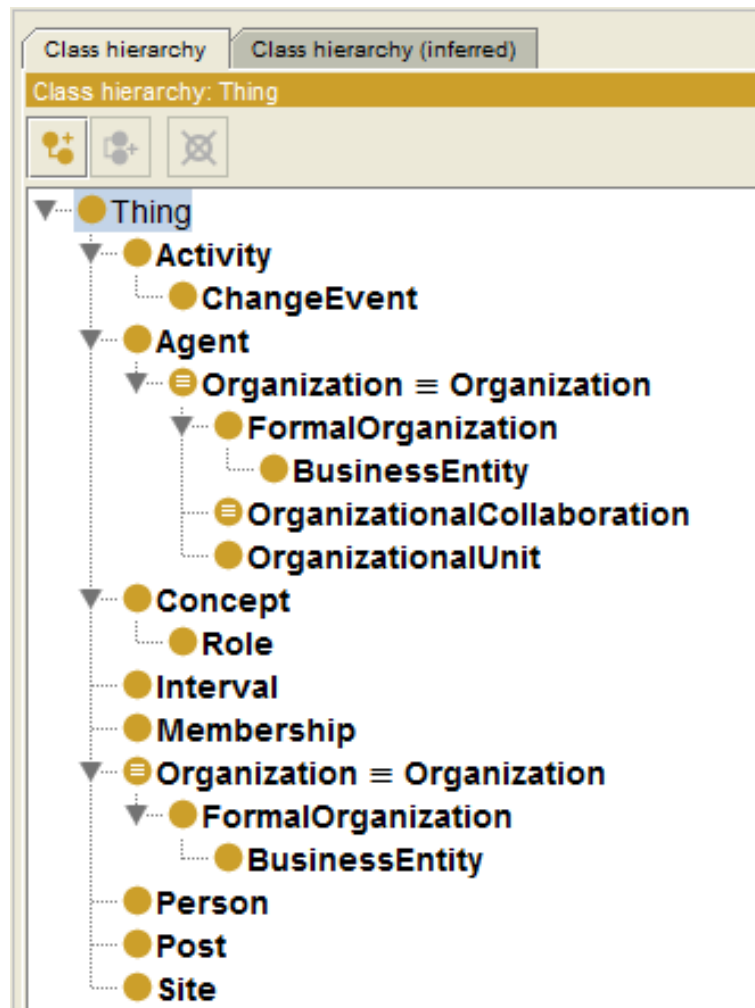
Rysunek 1: Główne pojęcia i role w ontologii ORG. Źródło [4].

- została zaimplementowana w prostym języku logiki deskrypcyjnej,
- jest dobrze opisana i udokumentowana; posiada opis (prawie) każdego elementu w językach angielskim oraz francuskim,
- łatwo ją rozszerzać ze względu na użyty język i dokładny opis,
- ontologia w zamierzeniu ma być jeszcze rozwijana (ostatnia wersja z 23 października 2012).

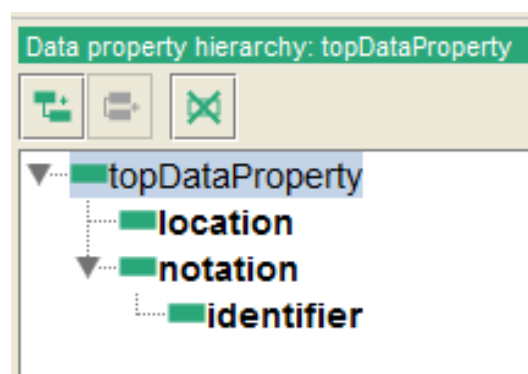
W porównaniu do szkieletu taksonomii onto-sth czy też onto-orb ontologia ORG jest mniej dokładna i bardziej ogólna. Tym samym, mogłaby być użyta jako warstwa wyższa, którą należałoby odpowiednio rozszerzyć.

2.3.2. Ontologia organizacji IntelLEO

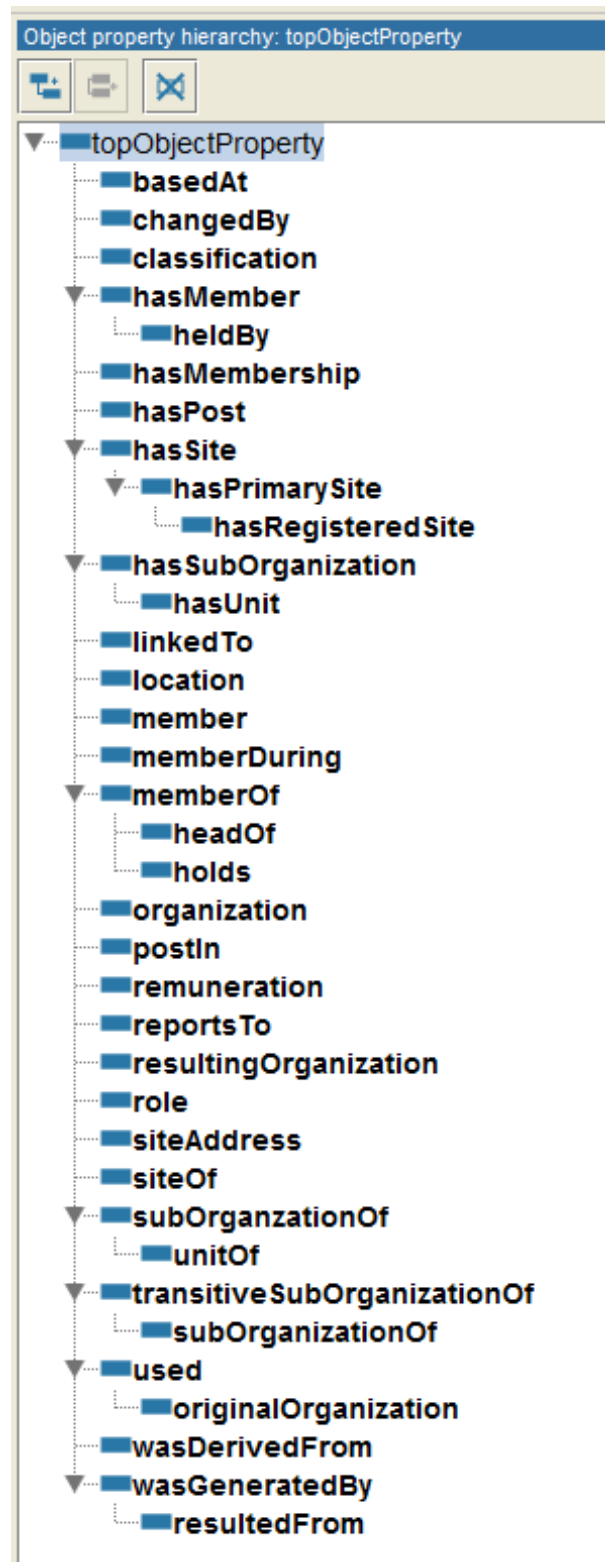
Ontologia organizacji IntelLEO powstała w ramach projektu Intelligent Learning Extended Organisations. Celem ontologii jest zdefiniowanie podstawowych pojęć i ról służących do opisu struktur organizacji, przypisania do niej pracowników oraz wyrażenia



Rysunek 2: Hierarchia klas w ontologii ORG.



Rysunek 3: Hierarchia relacji/ról typu DatatypeProperties.



Rysunek 4: Hierarchia relacji/ról pomiędzy obiektami (ObjectProperties) w ontologii ORG.

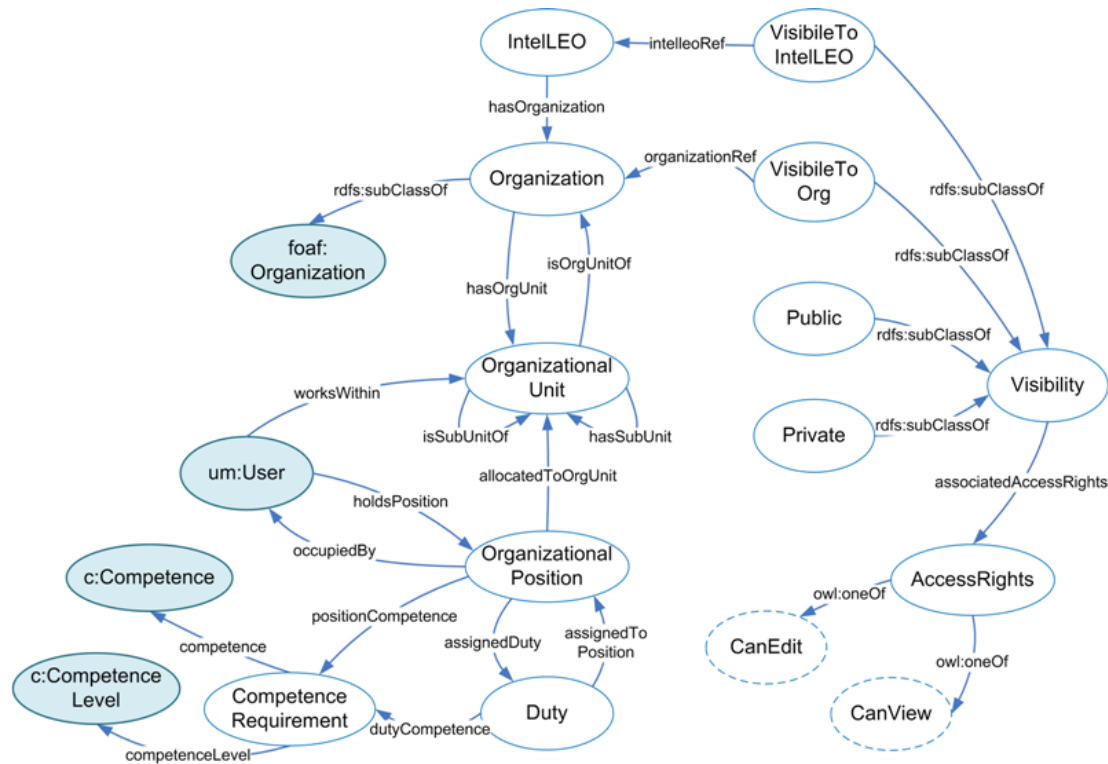
obowiązków pracownika przypisanych do danego stanowiska. Za pomocą ontologii IntelLEO można opisać następujące elementy:

- a) organizację i jej jednostki podrzędne (wykorzystując relacje `hasSubUnit/isSubUnitOf`),
- b) stanowiska przypisane do danej jednostki organizacyjnej (np. `allocatedToOrgUnit`),
- c) obowiązki związane z danym stanowiskiem w firmie (jednostce organizacyjnej),
- d) przypisanie kompetencji i uprawnień (np. `requiresCompetence`, `positionCompetence`, `AccessRights`)
- e) grupę osób pracujących w zespole,
- f) poziomy „widzialności” dla innych jednostek organizacji (nie tylko podrzędnych) wraz z określeniem uprawnień (np. `visibility`, `VisibleToOrganization`, `associatedAccessRights`).

Ontologia IntelLEO zdefiniowana została w języku *ALUOIF(D)*. Niestety, żadne z dostępnych narzędzi wnioskujących (Pellet, Fact++, Hermit) nie potrafi poprawnie wyliczyć hierarchii pojęć i ról. Każde narzędzie zwraca inny błąd (dotyczy relacji `hasSubCompetence` oraz typu danych „XMLSchema#date”. Ontologia wymaga poprawek, żeby prawidłowo przeprowadzić proces wnioskowania i tym samym zapewnić jej spójność. Główne pojęcia i role tej ontologii zostały przedstawione na Rysunku 5.

Główne cechy ontologii IntelLEO są następujące:

- a) ontologia nie jest już rozwijana,
- b) jest dobrze udokumentowana,
- c) w ciekawy sposób modeluje struktury organizacyjne bazując na kompetencjach poszczególnych osób (stanowisk pracowniczych) oraz ich wzajemnych powiązań (definiując również prawa dostępu między nimi),



Rysunek 5: Główne pojęcia i role w ontologii IntelLEO. Źródło [5]

d) ontologia sprawia problemy związane z wyliczeniem hierarchii oraz sprawdzeniem spójności.

W porównaniu do szkieletu ontologii `onto-sth` czy też `onto-orb` ontologia IntelLEO pozwala modelować strukturę organizacyjną firmy/przedsiębiorstwa oraz przypisać poszczególne zakresy kompetencji. Umożliwia również podział jednostek organizacyjnych celem wymiany informacji pomiędzy nimi w zależności od ustalonych praw dostępu. Dodatkowo, co pokrywa się z ontologią `onto-orb` mamy możliwość nadawania priorytetów kompetencjom (po nieznacznej modyfikacji również zadaniom). Ontologia IntelLEO nie pozwala jednak na modelowanie procesów zachodzących w organizacjach ani na definiowanie wymagań dotyczących osób, jednostek czy też projektów.

2.3.3. Ontologia organizacji GZ

Ontologia organizacji GZ jest modułem ontologii Gazette służącej do opisu treści dziennika The London Gazette. W niniejszym opracowaniu skupiono się tylko na ontologii GZ. Ontologia jest udokumentowana w bardzo ograniczony sposób. Dostępne są tylko proste opisy pojęć i relacji. Niestety spora część pojęć oraz relacji nie jest wyjaśniona. Część jest opisana poprzez rozbitcie połączonych wyrazów definiujących pojęcie bądź relację, np. rola `hasName` ma opis `has Name`. Hierarchia klas dotyczących organizacji (firm) z ontologii GZ została przedstawiona na Rysunku 6.



Rysunek 6: Taksonomia pojęć dotyczących organizacji w ontologii GZ.

Ontologia GZ zawiera więcej pojęć niż przedstawione na rysunku. Część z nich pochodzi z innych modułów ontologii. Pełna ontologia Gazette zawiera 24 moduły (importowane ontologie), do których należą chociażby znane ontologie takie jak: FOAF [10], czy Dublin Core [11]. Ontologia GZ została napisana w języku *SHIN(D)*. Narzędzia wnioskujące nie sygnalizują wykrycia niespójności i poprawnie wyliczają hierarchie pojęć oraz relacji.

W porównaniu do szkieletu ontologii onto-sth czy też onto-orb ontologia GZ dostarcza mechanizmów opisywania organizacji w kontekście jej działalności, np. czy firma jest organizacją zarobkową, charytatywną czy też rządową. Nie ma możliwości wyrażenia struktury samej organizacji. Ontologię można by było wykorzystać jako element opisujący organizację/przedsiębiorstwo z punktu widzenia jego działalności a nie samej struktury czy też procesów w nim zachodzących.

2.3.4. Ontologia PROTON

Ontologia jest przeznaczona jako warstwa wyższa służąca do opisu zadań, dokumentów, organizacji, obiektów, wydarzeń, czasu i lokalizacji. Składa się z 4 modułów: System, Top, Upper, and KM (Knowledge Management). Ontologia została zaimplementowana w języku *ALCHI(D)* i swoim zakresem nie wykracza poza OWL Lite [26]. Ontologia PROTON jest stosunkowo prosta i dobrze udokumentowana. Jest najbardziej rozbudowaną ontologią spośród wszystkich przeanalizowanych. Ontologia była wykorzystana w projekcie SEKT – Semantic Knowledge Technologies³. Ontologia zawiera ponad 250 pojęć i ponad 100 ról obu dopuszczalnych typów (Object i Datatype). Fragment części dotyczącej organizacji został przedstawiony na Rysunku 7.

W porównaniu do szkieletu ontologii onto-sth czy też onto-orb ontologia PROTON pokrywa następujące obszary: tworzenie projektów, opis dokumentów i konkretnych stanowisk w firmie (Manager, Kierownik itd.). Brakuje możliwości opisanie procesów.

2.3.5. Inne ontologie

W ramach zadania 2 zwrócono również uwagę na inne, istotne dla projektu Prosecco, ontologie:

³<http://www.sekt-project.com/>



Rysunek 7: Taksonomia pojęć dotyczących organizacji w ontologii PROTON.

1. Unified Enterprise Modelling Ontology (UEMO) [12] – przedstawia ontologię przedsiębiorstwa i modelowania system informatycznego opartą o UEMML, Unified Enterprise Modeling Language. Jest sprzężona z BPM.
2. O-CREAM-v2, a core reference ontology for the CRM domain [13] – przedstawia szczegółową ontologię CRM (Customer Relationship Management). Jedną z jej wad jest brak nacisku na service (usługę).
3. WeCoTin [14] – ontologia służąca do opisu dopasowywania oferty do wymagań

klienta.

4. Z ontologii bardzo szczegółowych na uwagę zasługuje ontologia e-business (w szczególności e-procurement) – opracowana w ramach projektu DIP [15]. Jest to połączenie BULO Upper ontology (wcześniejszy PROTON) ze sformalizowanym UBL (Universal Business Language). UBL jest biblioteką standardowych elektronicznych dokumentów biznesowych w formacie XML, takich jak zamówienie i faktura. UBL został opracowany przez OASIS Technical Committee.

Niestety, ontologie UEMO, O-CREAM-v2 i WeCoTin nie są one dostępne publicznie i tym samym nie ma możliwości ich wnikliwej analizy czy też porównania ze szkieletem taksonomii onto-sth i onto-orb. Ontologia O-CREAM-v2 została opisana w rozdziale 2.4.

2.4. Analiza znanych ontologii biznesowych BPMN oraz CRM

Ontologie BPMN

The Business Process Model and Notation⁴ (Model procesów biznesowych i notacja), (BPMN) jest powszechnie stosowanym standardem w modelowaniu procesów biznesowych, zarządzanym przez Object Management Group (OMG).

Specyfikacja BPMN 2.0 jest bardzo wszechstronna i obejmuje ponad 500 stron. Już dla BPMN 1.0 podkreślano w literaturze niekonsystencje i niejasności [20, 21]. Podobnie jest z wersją 2.0 [22]. Definicje elementów są rozproszone w różnych sekcjach, a czasem sprzeczne. Ponadto, struktura elementów i ich relacji są opisane w meta modelu, jednak dalej reguły składniowe są określone w języku naturalnym w tekście. W pracy [22] zdefiniowano ontologię BPMN 2.0, która może formalnie reprezentować specyfikację BPMN. Elementy są połączone w odpowiednie klasy i dalsze wyjaśnienia znajdują się w adnotacjach. To pozwala na znacznie szybsze

⁴<http://www.omg.org/spec/BPMN/2.0>

zrozumienie BPMN. Ponadto, ontologia ta służy do sprawdzania składni w celu weryfikacji konkretnych modeli BPMN. Niestety nie wszystkie elementy są opisane w wyczerpujący sposób. Ontologia bpmn20base jest oparta na specyfikacji metamodelu BPMN, w tym wszystkich diagramów klas, tabel określających atrybuty i relacje oraz schematy XML.

Ontologia bpmn20⁵ pochodzi od ontologii bpmn20base i zapewnia jej rozszerzenie. Zawiera wymagania składniowe zaczerpnięte z tekstu specyfikacji BPMN. Dlatego ontologia bpmn20 dodaje nowe lub poprawia istniejące klasy i ograniczenia. Ogólnym celem jest to, żeby Ontologia BPMN 2.0 służyła jako baza wiedzy dla prawie wszystkich zasad składniowych w specyfikacji BPMN. Spójność Ontologii BPMN 2.0 została zweryfikowana 3 silnikami wnioskującymi: FaCT++, Pellet i HermiT 1.2.4. Wadą ontologii BPMN jest jednak to, że jej modalność jest niejawnie wyrażona przez strukturę przebiegu procesu. Wszystkie działania są niejawnie obowiązkowe i gdy coś powinno być opcjonalne, używa się bramy czy zdarzenia do dzielenia przebiegu procesu i oferuje się możliwość realizacji zadania lub nie robić nic. To wymaga kompleksowego zrozumienia całego procesu w celu identyfikacji wymaganych, opcjonalnych lub alternatywnych działań. Ontologia bpmn20 została zaimplementowana w języku *ALCHIQ(D)*. Wadą ontologii bpmn20 jest jej licencja, która uniemożliwia jej zastosowanie w komercyjnych aplikacjach - jest to licencja Creative Commons BY-NC-SA 3.0⁶.

Wcześniejsza wersja standardu BPMN również została poddana próbie jej wyrażenia w ontologii [23]. Ontologia BPMN⁷ bazuje na specyfikacji BPMN 1.1. Niestety, również w tym przypadku głównym mankamentem ontologii jest jej licencja CC BY-NC-SA 3.0.

⁵[http://www.scch.at/Files/bpmn20\(1\).owl](http://www.scch.at/Files/bpmn20(1).owl)

⁶<http://creativecommons.org/licenses/by-nc-sa/3.0/>

⁷https://dkm.fbk.eu/index.php/BPMN_Ontology

Ontologie CRM

Zarządzanie konfiguracją jest ważnym problemem w rozwoju dużych programów i systemów. Gdy mamy do czynienia z setkami komponentów, śledzenie wersji zmian i różne ograniczenia zależnościami nałożone na system, w całym jego cyklu życia rozwój jest bardzo trudne. Oprócz rozwiązań firmowych (ad hoc) i zastrzeżonych istnieje standard określający poprawną konfigurację oprogramowania - ANSI EIA-649-B Configuration Management Standard⁸ (dostępny, ale płatne są Załączniki). Definiuje pięć funkcji CM wraz z ich zasadami ich użycia:

1. Configuration Management Planning and Management.
2. Configuration Identification.
3. Configuration Change Management.
4. Configuration Status Accounting.
5. Configuration Verification and Audit.

Wymagania są zaprojektowane indywidualnie oraz określają istotę funkcji związanej z zarządzaniem treścią, i mogą być stosowane wspólnie aby utworzyć listę kryteriów oceny program CM. W opisie każdej funkcji CM i jej szczegółów, standard ten wykorzystuje neutralną terminologię zarządzania konfiguracją, a także zapewniających równoważne warunki, które w przeszłości były używane w różnych środowiskach produktów. Nie ma to na celu faworyzować konkretnego zestawu terminologii. Podobnie, standard ten wykorzystuje neutralny zestaw nazw dla faz cyklu życia produktu, które są na tyle ogólna łatwo odwzorowywane na mnóstwo różnych modeli cyklu życia w użyciu. Standard ten jest bardzo szczegółowy, ale niesemantyczny. Istnieją sformalizowane ontologiczne reprezentacje Software Configuration Ontology [24, 25].

⁸daphne.meccahosting.com/~a00070f5/EIA-649-B\%20CM\%20Std.pdf

W pracy [25] ograniczenia komponentów i ograniczenia wersji są zakodowane w ontologii przy wykorzystaniu standardowego języka OWL DL [26], co ułatwia dzielenie się wiedzą na temat konfiguracji, w różnych systemach. Wykrywanie i poszukiwanie elementów nieścisłości przez człowieka, jest żmudnym i czasochłonnym procesem. Podstawowe założenia to:

- Wersje pakietów są traktowane jak obiekty klas. Obiekty te muszą być zadeklarowane jako parami różne.
- Pakiety są traktowane jako klasy w ontologii. Każda klasa jest określona przez bezpośrednie wyliczenie jej obiektów. Odbywa się to przy użyciu owl:oneOf.
- Ograniczenia wersji pakietu są reprezentowane przez klasy. Dla każdego ograniczenia, musi zostać utworzona nowa klasa.
- Relacje są dodawane do ontologii dla prawidłowego modelowania ograniczeń.
- ValidConfiguration jest klasą, która określa ograniczenia i może być każdy dowolnie skomplikowane wyrażenie logiczne.
- Testem jest indywiduum, która określa konfigurację. Te konfiguracje mogą być *valid* lub *invalid*. Test jest określony jako owl: differentFrom wszystkie innych obiektów w ontologii. Wszystkie wersje pakietów w teście konfiguracji dodaje się do tego obiektu, przy użyciu obiektowych asercji relacji P.

O-CREAM-v2, a core reference ontology for the CRM domain [13] – przedstawia szczegółową ontologię CRM opartą o szczytową ontologię Dolce [19]. O-CREAM-v2 składa się z pięciu modułów: Relacje (Relationship), Wiedza (Knowledge), Działania (Activities), trzy pierwsze obejmujące zarówno górną i dolną warstwę rdzenia ontologii), oprogramowanie (Software) and i Różne Miscellaneous (zarówno ograniczone do górnego rdzenia). Pięć modułów O-CREAM są szczegółowo opisane w pracy. Obecnie O-CREAM-v2 zawiera kilkaset klas, relacji i reguł. Zastosowanie ontologii

zostało przewidziane dla dwóch możliwych perspektyw zastosowań (a) repozytoriów opartych na sieci Web wsparcia mediacji pomiędzy potrzebami firmy a funkcjonalnościami istniejących narzędzi związanych z CRM (b) narzędzia wspierające użytkowników w budowaniu formalnych reprezentacji zasobów w celu zastosowania semantycznych systemów ekstrakcji informacji.

Ontologia O-CREAM-v2 rozwinięta przez grupę z *Dipartimento di Informatica, Università di Torino, Italy (Unito)*⁹ jest formalnie zdefiniowana. Jej część dotycząca systemów informatycznych korzysta z pracy [16]. Ontologia ta jest wykorzystana w systemie ARNEIS (Advanced Repository for Needs of Enterprises and Innovative Software) [17], prototypowej implementacji webowego repozytorium opisów rozwiązań dla CRM, które wykorzystuje Semantic Web Services. Użytkownik jest w stanie dopasować istniejące rozwiązania CRM do swoich potrzeb. Opublikowane artykuły grupy Unito nie dają pełnego obrazu szczegółowości ontologii. Wydaje się jednak, że nie jest ona wystarczająco głęboka, jeśli sędzić z przykładu zamieszczonego w [17] – order management (gestione ordini). Ontologia O-CREAM-v2 jest używana raczej do procesów operacyjnych niż do CRM analitycznego. Choć więc cele ARNEIS i Prosecco są zbliżone, wg nas wyniki przy użyciu O-CREAM-v2 są skromne. Pewne elementy ontologii mogącej znaleźć zastosowanie w CRM analitycznym znajdują się w [18].

⁹http://www.unito.it/unitoWAR/appmanager/dipartimenti1/D004_en?_nfpb=true

3. Rezultaty

W wyniku prac przeprowadzonych w ramach zadania 2 uzyskano następujące rezultaty:

1. Przygotowano propozycję wymagań funkcjonalnych, które pozwoliłyby na przygotowanie modelu taksonomii dedykowanego dla danej grupy firm (uwzględniając ich rzeczywiste potrzeby w danym obszarze). Źródłem wiedzy na temat potrzeb oraz funkcjonowania poszczególnych firm były opisy procesów biznesowych wraz z zestawem reguł biznesowych znajdujące się w w systemie Do-kuWiki firmy Softhis.
2. Na podstawie przeanalizowanych metodyk i dobrych praktyk dostępnych w literaturze opracowano metodykę pozyskiwania wiedzy do modelu taksonomicznego projektu Prosecco. Zakłada ona podejście *middle-out* w procesie pozyskiwania wiedzy oraz wykorzystanie pośredniej, semi-formalnej reprezentacji wiedzy. Opracowany proces pozyskiwania wiedzy został opisany w rozdziale 3.2.
3. Opracowano semantyczny model ontologiczny, na którym ma bazować zarządzanie przedsiębiorstwami o małej i średniej wielkości, bazuje na pojęciach, rolach oraz atrybutach zidentyfikowanych w wyniku analizy działalności następujących firm: Softhis, PromoAgency, NOLABEL, IQON Consulting i Simply User.

Semantyczny model ontologiczny (SMO) ma za zadanie wspomagać elementy procesów i ich wzajemne relacje, wspierać wyszukiwanie danych (zapytania o charakterze semantycznym) oraz wspierać opracowywane narzędzie rekomendacyjne.

Model SMO będzie uwzględniał metodykę Business Motivation Model¹⁰ (BMM) wykorzystywaną do opisu organizacji oraz logiki biznesowej, np. środki, dyrektywy, reguły. Dodatkowo, zgodnie z założeniami BMM, pojęcia oraz role występujące w modelu SMO będą mogły zostać wykorzystane w budowie reguł takich jak SBVR¹¹ czy reguł produkcji¹². Semantyczny model ontologiczny, zgodnie z Ontology Definition MetaModel¹³, jest tworzony uwzględniając podejście *model centric* (konceptualizacja, struktura, język formalny), natomiast implementacja będzie uwzględniała podejście *application centric* (zastosowanie, integracja, wersjonowanie). Model SMO został przedstawiony w rozdziale 3.3.

3.1. Specyfikacja wymagań dla modelu taksonomii pojęć

Celem całościowym projektu Prosecco jest opracowanie narzędzi informatycznych, które będą wspomagać i ułatwiać zarządzanie przedsiębiorstwa z sektora małych i średnich firm, ze szczególnym uwzględnieniem firm świadczące usługi IT. Zaprezentowany w rozdziale 2.1 przegląd funkcjonalności otwartych systemów klasy DMS oraz CRM zdecydowanie przekracza swoim zakresem potrzeby docelowych beneficjentów projektu Prosecco.

Opracowana propozycja specyfikacji wymagań, wraz ze wstępną identyfikacją podstawowych elementów proponowanego rozwiązania, łączy w sobie funkcjonalności CRM, DMS oraz wspomaganie zarządzania projektami.

Projektowane oprogramowanie powinno posiadać następujące cechy:

- możliwość definiowania nowych profili kontrahentów zawierających niezbędne dane kontaktowe oraz ich późniejszą modyfikację

¹⁰OMG Business Motivation Model Specification v1.1, <http://www.omg.org/spec/BMM/1.1>

¹¹OMG Semantics of Business Vocabulary and Business Rules, <http://www.omg.org/spec/SBVR/1.1/>

¹²OMG Production Rule Representation, <http://www.omg.org/spec/PRR/1.0/>

¹³OMG Ontology Definition Metamodel, <http://www.omg.org/spec/ODM/>

- przypisywanie osób dedykowanych do obsługi kontrahenta oraz wyspecyfikowanie zadań im przydzielonych
- przydzielanie do klientów usług i produktów oraz możliwość ich edycji i zmian w konfiguracjach
- definiowanie profili użytkowników systemu wraz z przyziałem odpowiednich praw do systemu
- podgląd wszelkich zdarzeń wykonanych w systemie z rozróżnieniem na użytkownika wykonującego daną operację
- funkcja wysłania wiadomości do wszystkich klientów z ważną informacją dotyczącą działań podjętych we firmie
- Zarządzanie kalendarzem zdarzeń dla pracownika
- Dodawanie/usuwanie/edycja przyszłych spotkań i telefonów do kalendarza
- Historia kontaktów z klientami
- Zarządzanie listą oferowanych produktów i usług
- Lista sprzedanych produktów klientom
- Możliwość projektowania i generowania podstawowych raportów
- Historia kontaktów klienta z firmą
- Dostęp do archiwum dokumentów związanych z danym projektem
- Możliwość wysyłania maili do klientów z systemu
- Definiowanie szablonów maili wraz z tworzonymi w systemie umowami/ofertami jako załącznikami

- Lista wysyłanych do klienta dokumentów
- Dodawanie notatek do spotkań
- Lista notatek w ramach obsługi klienta
- Wygenerowanie historii spotkań/telefonów, notatek, złożonych ofert i zamówień do postaci pdf

Proponowana architektura rozwiązania wyróżnia następujące obiekty główne:

Kontrahenci

Celem wprowadzenia pojęcia Kontrahent jest umożliwienie zarządzania firmami i bezpośrednimi kontaktami personalnymi w tych firmach. Pojęcie to umożliwia dodawanie, usuwanie, edycję firm i kontaktów. Firma zawierać będzie cechy typu: nazwa, numery: NIP, REGON, KRS, itp., branżę, logo, nr telefonu, faksu, adres email, itp., ale również dane adresowe, kontaktowe, lokalizację. Firma będzie mogła posiadać wiele bezpośrednich kontaktów (osób), z którymi będzie można współpracować. Osoby takie będą posiadały cechy typu: imiona, nazwisko, nr telefonów, adres email, zdjęcie, stanowisko, itp. Pojęcie ten ma zapewnić powiązanie firmy z osobami kontaktowymi oraz przyjazny sposób zarządzania tymi danymi i kontaktowania się z wybranymi osobami.

Do informacji o kontrahentach przechowywanych w systemie powinny należeć także między innymi:

- dane teleadresowe
- status klienta
- lista osób do kontaktu z firmy klienta
- lista faktur związanych z klientem oraz informacje o ich statusie
- lista plików związanych z klientem wraz z automatycznym numerowaniem wersji

- lista projektów związanych z klientem
- informacja o aktualnym statusie oraz historia zmian statusu
- historia zdarzeń związanych z klientem (telefony, wizyty, itd.)
- planowane zadania
- informacje o dacie i sposobie pozyskania klienta
- lista pracowników do których został przypisany klient oraz informacje o branży z którą jest związany
- opiekunowie klienta czyli użytkownicy systemu odpowiedzialni za kontakty z danym klientem

Funkcjonalności:

- Przechowywanie oraz odpowiednie przetwarzanie informacji o kontrahentach
- Zaawansowane wyszukiwanie i sortowanie klientów według różnorodnych kryteriów
- Nadawanie klientom statusów (np. "potencjalny", aktywny")
- Mailing – możliwość wysyłania wiadomości do wybranych klientów i grup klientów
- Aktualizować informacje o klientach oraz monitorować i planować działania, które powinny zostać podjęte w celu zwiększenia jakości obsługi klienta

Zadania

Istotą CRM jest praca na wielu zadaniach z wieloma klientami, celem utrzymania stałego kontaktu z klientami i utrzymywaniu relacji klienta z firmą. Terminowe

wywiązywanie się z podjętych przed Klientem zadań świadczy również o profesjonalizmie firmy. CRM powinien wspierać efektywne zarządzanie zadaniami (kontaktami z klientem, spotkaniami, zadaniami innych typów) poprzez:

- Łączne spojrzenie na wszelkie zadania w systemie (wynikające z wielu spraw, wielu klientów, wielu typów zadań) w 1 miejscu,
- Możliwość sortowania i filtrowania łącznej kolejki zadań po ich statusie, nadchodzącym terminie realizacji, priorytecie etc.
- Możliwość delegowania zadań innym osobom, możliwość śledzenia ich statusu
- Możliwość notowania zadań dodatkowych, nie wynikających wprost z ustalonego procesu obsługi klienta
- Możliwość określania priorytetu terminu podanego dla zadania
- Powiadamianie o nowych / zaległych zadaniach w systemie
- Wizualizacja zadań na kalendarzu

Każdemu z pracowników po zalogowaniu do systemu powinna być prezentowana lista zadań do wykonania, terminy ich realizacji, priorytety oraz najważniejsze cele. Przykładowe kategorie zadań, które mogą zostać przydzielone to: telefon lub wizyta u klienta, dostarczenie lub odbiór towaru, przygotowanie i wysłanie pisma, itp.

Wymagana funkcjonalność:

- definiowanie zadań do wykonania (również zadań cyklicznych)
- przydzielanie zadań do realizacji wybranym osobom
- śledzenie przebiegu wykonania zadań
- tworzenie szczegółowych raportów działań wg dowolnych kryteriów

- możliwość zdefiniowania przebiegu realizacji zadania. Dzięki temu osoba nadzorująca może błyskawicznie pozyskać informacje takie jak: rzeczywisty termin realizacji, osiągnięte efekty, informacje pozyskane w trakcie realizacji (np. informacje od klienta), ewentualne problemy przy wykonaniu zadania i uwagi do niego
- przypominanie o zadaniach priorytetowych, wyróżnianie zadań zaległych.
- możliwość zdefiniowania powiadomień na e-mail lub SMS.
- raporty z zadań.

Produkty

Celem produktu jest umożliwienie zarządzania produktami, które znajdują się w ofercie Spółki. Produkty zawierają cechy typu: nazwa, producent, wielkość, cena, gatunek, rodzaj, itp. Mogą występować wzajemne relacje pomiędzy poszczególnymi produktami, np. komputer HP zawiera 2 kości pamięci Kingston, laptop Dell jest połączony z systemem operacyjnym Windows. Wprowadzone produkty w systemie będą brały udział w dalszym procesie sprzedażowym, w tym będą mogły być dołączane do ofert, itp.

Funkcjonalności:

- Dodawanie/usuwanie/edycja produktów
- Lista produktów
- Wyszukiwanie produktów
- Filtrowanie produktów
- Klasyfikacja produktów: rodzaj, typ, produkt

Faktura

Celem wprowadzenia pojęcia faktura jest umożliwienie zarządzania fakturami, wystawianymi przez firmę.

Funkcjonalności:

- Lista faktur
- Wyświetlanie danych faktury
- Wyszukiwanie faktur
- Filtrowanie faktur
- Generowanie faktury do postaci pdf i html
- Wyświetlanie korekt do faktur
- Wyświetlanie korekt do pozycji na fakturze

Usługa

Celem usługi jest umożliwienie zarządzania modelem usług świadczonych przez firmę. Usługi zawierać będą cechy typu: nazwa, dział świadczący usługi, koszt usługi, sugerowana cena, itp. Wprowadzone usługi będą ponadto brały udział w dalszym procesie sprzedażowym, np. tworzeniu ofert, zapytań, itp.

Funkcjonalności:

- Dodawanie/usuwanie/edycja usług wewnętrznych/zewnętrznych/dodatkowych
- Dodawanie/usuwanie/edycja usług zewnętrznych/dodatkowych
- Lista usług wewnętrznych/zewnętrznych/dodatkowych
- Wyszukiwanie usług
- Filtrowanie usług

- Klasyfikacja usług: wewnętrzna, zewnętrzna, dodatkowa

Dokumenty

Celem wprowadzenia pojęcia dokumentu jest umożliwienie automatycznego tworzenia dokumentów typu np. .rtf, .pdf na podstawie utworzonych szablonów danych (umowy, oferty, itp.) oraz danych przechowywanych w bazie danych. Użytkownicy powinni mieć możliwość tworzenia w aplikacji własnych szablonów dokumentów na podstawie posiadanych danych i generowania dokumentów w różnych miejscach w aplikacji w zależności od potrzeb.

Wymagana funkcjonalności

- Generowanie dokumentów do postaci pdf i html (ewentualnie rtf)
- Definiowanie szablonów umów/zamówień/ofert/faktur
- System powinien pozwalać na szybkie i wygodne wgrywanie plików, grupowanie ich w katalogi oraz edytowanie połączone z automatycznym numerowaniem kolejnych wersji
- Możliwość łatwego powiązania plików(dokumentów) z kontaktami, zdarzeniami lub projektami
- Zarządzanie dostępem (uprawnieniami) dla poszczególnych katalogów i plików

Projekty

Projektowany system CRM powinien pozwalać na łatwe i intuicyjne zarządzanie projektami realizowanymi przez firmę. Powinna istnieć możliwość stworzenia hierarchicznej struktury projektu oraz łatwe kontrolowanie harmonogramu projektu z wykorzystaniem wybranych notacji graficznych np. diagramu Gantta.

Dla każdego z projektów powinna istnieć możliwość zdefiniowania:

- statusów

- dat rozpoczęcia i zakończenia
- uczestników projektu wraz z indywidualnymi uprawnieniami
- praw dostępu uczestników do projektu (prawa do edycji, do odczytu, itd.)
- zadań realizowanych w ramach projektu
- klientów związanych z projektem
- plików związanych z projektem
- pozycji budżetowych, planowanych kosztów i przychodów
- faktur związanych z projektem
- podprojektów, które można traktować zarówno jako kolejne etapy projektu, jak również jako niezależne jednostki realizowane w ramach projektu głównego

Zróżnicowany poziom uprawnień do projektów dla poszczególnych uczestników pozwala na pełną kontrolę dostępu do danych oraz plików powiązanych z projektami. W zależności od roli danego użytkownika jaką pełni on w projekcie można przykładowo ograniczyć mu możliwość edycji głównych danych projektowych oraz widoczność sekcji definiującej kontrahentów powiązanych z projektem. W przypadku tworzenia hierarchicznej struktury możliwe jest niezależne zdefiniowanie różnych uprawnień tego użytkownika dla każdego z podprojektów. Inną istotną cechą jaką powinien mieć projektowany system jest możliwość śledzenia historii zmian w projekcie przez osoby o pełnych uprawnieniach. Funkcja ta pozwoliłaby na szybkie wychwycenie, kto odpowiada za konkretne zmiany podczas planowania oraz realizacji projektu.

3.2. Proces pozyskiwania wiedzy

Podstawową metodą pozyskiwania wiedzy były rozmowy z pracownikami analizowanych firm. W trakcie spotkań analitycznych notowane były podstawowe pojęcia

i relacje występujące między nimi. Jednak bazą dla taksonomii były zidentyfikowane procesy biznesowe, jako zawierające **najważniejsze** pojęcia, relacje, atrybuty. Na ich podstawie wyszczególniono podstawowe klasy, relacje i atrybuty istotne dla działania firmy. Graficzno-tekstowe notatki ze spotkań stanowiły ewentualne uzupełnienie dla powyższej konceptualizacji.

Projekt Prosecco ma na celu wypracowanie technologii informatycznych, które posłużą do usprawnienia zarządzania firmami z sektora małych i średnich przedsiębiorstw. Jednym z podstawowych artefaktów zidentyfikowanych do stworzenia w projekcie jest ontologia. W założeniu powinna ona pomóc użytkownikowi na lepsze dopasowanie systemu do potrzeb jego przedsiębiorstwa poprzez możliwość modelowania pojęć i relacji występujących w firmie, a także pozwolić na bardziej zaawansowane wyszukiwania informacji. W celu stworzenia formalnego modelu ontologicznego projektu Prosecco przyjęto metodologię polegającą na analizie taksonomicznej wybranych firm, a także istniejących systemów do zarządzania przedsiębiorstwami. Proces budowania ontologii składa się z etapu analizy dziedzinowej wybranych firm reprezentujących pewną klasę przedsiębiorstw, modelowanie wiedzy taksonomicznej w *semi-formalnej* postaci, a następnie analizę otrzymanych wyników. Aby tego dokonać, przeprowadzono szereg ustrukturalizowanych wywiadów z przedstawicielami wybranych firm. W trakcie wywiadów, zorientowanych przede wszystkim na opis procesów biznesowych realizowanych w firmach zidentyfikowano kluczowe role, pojęcia, obiekty biorące udział w procesach oraz relacje występujące między nimi. Prototypy modeli taksonomicznych muszą być spójne ze zidentyfikowanymi procesami biznesowymi oraz uwzględniać zidentyfikowane reguły biznesowe.

Dodatkowo, dobrano metody i narzędzia reprezentacji wiedzy, aby były z jednej strony intuicyjne i odpowiednie do szybkiego prototypowania, a z drugiej wystarczająco sformalizowane aby umożliwić automatyczne przetwarzanie (np. wyszukiwanie klas, relacji, atrybutów, a także wizualizację).

3.2.1. Spotkania analityczne

W ramach prac przeprowadzono szereg spotkań i ustrukturalizowanych wywiadów z pracownikami następujących firm:

- PromoAgency (<http://www.promoagency.pl/>),
- NOLABEL (<http://www.nolabel.pl/>),
- IQON Consulting (<http://www.iqon.pl/>),
- Simply User (<http://www.simplyuser.pl/>).

Celem analizy było pozyskanie wiedzy związanej z pojęciami występującymi w firmie oraz relacjami między nimi. Pojęcia te dotyczą zarówno struktury firmy (role, stanowiska, działy), jak i procesów biznesowych realizowanych w przedsiębiorstwie (głównie artefakty wytwarzane w ramach procesów).

3.2.2. Analiza otrzymanych materiałów i procesów biznesowych pod kątem taksonomii

Na podstawie spotkań, otrzymanych materiałów oraz notatek z wywiadów dokonano analizy zidentyfikowanych procesów biznesowych pod kątem spójności pojęciowej. Materiały od firmy PromoAgency znajdują się na stronie: http://wiki.softhis.com/prosecco:company_repo:promo:wprowadzenie. Zidentyfikowane w wywiadzie procesy firmy PromoAgency umieszczono na stronie: http://wiki.softhis.com/prosecco:company_repo:promo:procesy. Uwagi taksonomiczne dotyczyły przede wszystkim nazewnictwa stanowisk (programista, webdesigner, webdeveloper) oraz podziału obowiązków między stanowiska: Projekt Manager (PM), Account Executive (AE), Account Manager (AM). Uwagi miały charakter drobnych poprawek i były związane ze stosunkowo skomplikowanymi zasadami przydzielania odpowiedzialności w firmie.

Materiały od firmy IQON Consulting umieszczono na stronie: http://wiki.softhis.com/prosecco:company_repo:simplyuser:wprowadzenie, a zidentyfikowane procesy na stronie: http://wiki.softhis.com/prosecco:company_repo:iqon:procesy. Uwagi do opisu procesów firmy IQON Consulting polegały na wyróżnieniu projektów "twardych" (inwestycyjnych), w ramach których przygotowuje się: 1) wniosek, 2) załączniki i 3) biznesplan lub studium wykonalności oraz projektów "miękkich" (nieinwestycyjnych), w ramach których przygotowuje się 1) wniosek i 2) załączniki. Dodatkowo zidentyfikowano drobną niespójność pojęciową dotyczącą podpisania umowy.

Materiały od firmy SimplyUser umieszczono na stronie: http://wiki.softhis.com/prosecco:company_repo:simplyuser:wprowadzenie a zidentyfikowane procesy na stronie: http://wiki.softhis.com/prosecco:company_repo:simplyuser:procesy. W tym przypadku uwagi miały na celu uogólnienie pojęć pod kątem uogólnienia procesów (typowa nazwa procesu w pozostałych firmach to "Sprzedaż" – nazwy tej nie można zatem użyć do określenia uczestnika procesu). Ponadto wykryto nieścisłości polegające na pojawianiu się jednej aktywności w dwóch procesach oraz niedoprecyzowaniu czy opisywany podproces zawiera się w odpowiednim procesie nadrzędnym.

3.2.3. Prototypy taksonomii firm

Na podstawie analizy dostępnych metodyk pozyskiwania wiedzy i tworzenia ontologii, zdecydowano się na podejście *middle-out*, które zakłada identyfikację kluczowych pojęć, a następnie uogólnianie oraz uszczegóławianie ich. Za pojęcia kluczowe przyjęto osoby i artefakty występujące w zidentyfikowanych procesach biznesowych realizowanych w firmach. Dla każdej firmy listę pojęć (klas), relacje między nimi oraz atrybuty zapisano w plikach tekstowych w formacie odpowiednim do automatycznego przetwarzania i wizualizacji. Wszystkie pliki, zarówno tekstowe jak i graficzne umieszczono na serwerze firmy Softhis:

- PromoAgency: http://wiki.softhis.com/prosecco:company_repo:promo:taksonomia

- IQON Consulting: http://wiki.softhis.com/prosecco:company_repo:iqon:taksonomia
- SimplyUser: http://wiki.softhis.com/prosecco:company_repo:simplyuser:taksonomia

Znajdują się one również w załącznikach do niniejszego raportu.

3.2.4. Narzędzia reprezentacji wiedzy

Jako reprezentację pośrednią wybrano ustrukturalizowany opis tekstowy. Wybrane narzędzia to język SWI-Prolog (do zapisu bazy wiedzy oraz predykatów umożliwiających jej przetwarzania) oraz Graphviz (do wizualizacji bazy wiedzy). Korzyści płynące z takiego podejścia obejmują:

- połączenie intuicyjności opisu w języku naturalnym z możliwością automatycznego przetwarzania wiedzy,
- możliwość różnorodnego wykorzystania wiedzy za pomocą opracowania odpowiednich predykatów,
- możliwość dynamicznej wizualizacji.

Efektom jest stworzenie następujących plików:

- plik `psc2013ontology-sth.pl` zawierający szkic taksonomii firmy Softhis będącej słownikiem dla zidentyfikowanych procesów,
- plik `onto-sth.pdf` zawierający wizualizację ww. taksonomii,
- Pliki pomocnicze:
 - plik szkieletowy `psc2013ontology-skel.pl` do tworzenia kolejnych taksonomii w tym formacie,

- plik `psc2013onto-utils.pl` zawierający predykaty prologowe obsługujące format zapisu taksonomii (np. predykat `klasy/0` wypisujący wszystkie klasy, czy `relacje/0`, `atrybuty/0` itp.),
- plik `psc2013onto-graphviz.pl` zawierający predykaty służące do wizualizacji taksonomii z użyciem narzędzia `graphviz`.

Pliki te dostępne są na stronie: http://wiki.softhis.com/prosecco:company_repo:softhis:taksonomie.

Tworzenie taksonomii

W celu opisanie klas, relacji i atrybutów użyto następujących zapisów (predykatów):

- klasy: `c(?Id, ?Name)`,
- relacje: `r(?Subject, ?Predicate, ?Object)`,
- atrybuty: `a(?Class, ?AttributeName, ?ListOfPossibleValues)`.

W celu stworzenia nowej taksonomii należy utworzyć plik (lub zmodyfikować plik szkieletowy) wg następujących zasad:

1. Aby dodać nową relację, należy umieścić jej nazwę w sekcji Metadefinicje predykatów wg wzoru: `:- op(502, xfx, nazwa_relacji).`, a następnie wpisać daną relację w postaci: `r(argument1, nazwa_relacji, argument2)..`
2. Aby dodać nową klasę należy wpisać ją w postaci: `c(id_klasy, 'Nazwa klasy')..`
Przyjęto konwencję, że `id_klasy` jest uproszczoną nazwą ułatwiającą identyfikację klasy na wizualizacji taksonomii.
3. Aby dodać nowy atrybut, należy wpisać trójkę w postaci:
`a(id_klasy, nazwa_atrybutu, [lista,możliwych,wartości,atrybutu])..`

UWAGA: W celu poprawnej wizualizacji w taksonomii muszą pojawić się klauzule dotyczące każdej z grup: klasy, relacje, atrybuty.

W pliku psc2013ontology-skel znajduje się szkic taksonomii oraz przykłady poszczególnych zapisów:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%                               POCZĄTEK TAKSONOMII                               %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Metadefinicje predykatów          %
%% Przykład:                          %
%:- op(502, xfx, jest).                %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Definicje (aksjomaty)              %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Klasy                              %
%% Przykłady:                          %
%% Osoby/Role                          %
%c(analitikIT, 'Analityk IT'). % [rp]   %
%% Działy, jednostki, miejsca         %
%c(biuro, 'Biuro'). % [rp]             %
%% Artefakty                           %
%c(aneks_do_umowy, 'Aneks do umowy'). % [rp] %
%% Aktywności                          %
%c(aktywnosc, 'Aktywność').           %
%% Procesy                              %
%c(proces, 'Proces biznesowy').       %
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Atrybuty %
%% Atrybuty klas %
%a(zadanie, status, [nowe, zaplanowane, w_realizacji, zrealizowane]).%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Relacje %
%% Przykłady: %
%% Hierarchia klas %
%r(ewaluacja_prac jest aktywnosc). % [rk-e] %
% Odpowiedzialności %
%r(analitikIT konsultuje backlog). % [rp] %
% Inne zaleznosci %
%r(teczka_pracownika zawiera dok_pracownika). % [rk] %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% KONIEC TAKSONOMII %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Obsługa taksonomii

W pliku `psc2013onto-utils.pl` znajdują się predykaty do obsługi zapisanej taksonomii. Stosuje się je wywołując odpowiednie cele w języku Prolog.

1. Aby wyświetlić wszystkie klasy należy użyć predykatu `klasy/0` (`:- klasy.`).
2. Aby wyświetlić wszystkie relacje należy użyć predykatu `relacje/0` (`:- relacje.`).
3. Aby wyświetlić wszystkie atrybuty należy użyć predykatu `atrybuty/0` (`:- atrybuty.`).

Wizualizacja

Wizualizacja jest możliwa dzięki predykatom z pliku `psc2013onto-graphviz.pl` i wymaga zainstalowania narzędzia `graphviz`.

1. Aby zwizualizować taksonomię należy wywołać predykat `rysuj_ontologie/0` (`:- rysuj_ontologie.`).
2. Aby zapisać taksonomię do pliku (w formacie `*.dot`) należy wywołać predykat `zapisz_ontologie/0` (`:- zapisz_ontologie.`). Taksonomia zapisywana jest do pliku `graph-tmp.dot` w bieżącym katalogu. (**UWAGA:** Jeżeli wywołamy zapis dla kolejnej firmy, plik zostanie nadpisany.)

3.3. Model taksonomii pojęć

3.3.1. Analiza opracowanych szkieletów taksonomii

Dokładna analiza opracowanych siedmiu taksonomii (Orbizer, Softhis, IQON, Simply User, Simply User Extra, PromoAgency oraz NOLABEL) prowadzi do następujących wniosków o charakterze problemowym:

- Każda z taksonomii reprezentuje inne spojrzenie na działalność danej firmy, co powoduje dużą rozbieżność pojęć, relacji i atrybutów.
- Tylko niewielki fragment terminologii występuje we wszystkich 7 taksonomiach, przy czym w niektórych przypadkach różnice są dość drastyczne (np. taksonomie IQON – Simply User, gdzie opis projektu/zlecenia są z punktu teoretycznego zbliżone, natomiast ich reprezentacja taksonomiczna jest zróżnicowana – np. *realizacja_projektu obejmuje wystawienie faktury* versus *realizacja_zlecenia zawiera aktywność wystawienie faktury*).

- W zależności od taksonomii daje się zauważyć zróżnicowane spojrzenie i potrzeby analizowanych firm, jednakże we wszystkich przypadkach zaproponowana terminologia służy do opisu procesów, projektów, aktywności, zasobów, osób i różnego rodzaju artefaktów.
- Ujednolicenie stosowanej terminologii będzie się wiązało z dużym wysiłkiem i wymaga wprowadzenia dokładnego opisu stosowanych pojęć i relacji.
- Poziom abstrakcji stosowanych pojęć i relacji jest stosunkowo niski, co jest pożądane z praktycznego punktu widzenia, jednakże w przyszłości może się okazać niewystarczający w kontekście zadań wyszukiwania.
- Niektóre elementy taksonomii wymagają skomplikowanych konstrukcji (konstruktorów logik deskrypcyjnych bądź reguł), np. warunki liczbowe w IQON czy też zależności if-then-else w Promo Agency, jak również poziomy odpowiedzialności osób w firmie.
- Ontologia Orbizera (zapisana w OWL, dotycząca jednej z poprzednich wersji tego narzędzia) stanowi punkt wyjścia do rozwoju modelu SMO. Wymaga jednak udokumentowania – trudno się zorientować co oznaczają niektóre pojęcia oraz role, np. PBI czy partnerNDA.

3.3.2. Założenia dotyczące semantycznego modelu ontologicznego

Na podstawie analizy opracowanych taksonomii, wytycznych projektu Prosecco oraz uwag szczegółowych opracowano założenia metodyki tworzenia ontologii. Założenia te zgodne są z podejściem tworzenia ontologii *middle-out*.

Założenia konstruowanego semantycznego modelu ontologicznego (ontologii Prosecco, w skrócie OP) są następujące:

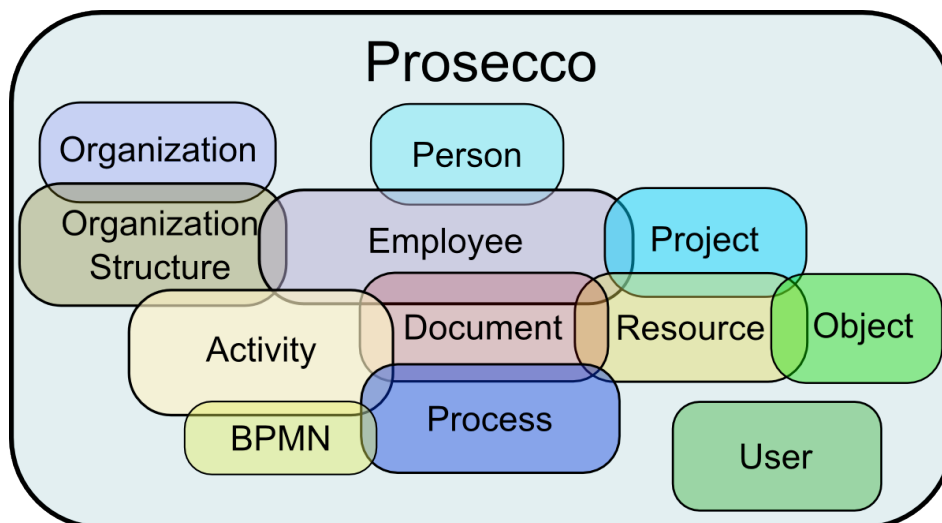
- Ontologia będzie mieć strukturę modułową – będzie podzielona na mniejsze ontologie, z których każda będzie dotyczyła innego zakresu tematycznego (innej dziedziny).
- Wykorzystane zostaną tylko i wyłącznie ontologie dziedzinowe bez uwzględniania ontologii fundacjonistycznych ze względu na ich wysoki poziom abstrakcji.
- Ontologia będzie wykorzystywana do opisu elementów występujących w procesach
- Ontologia powinna zostać zdefiniowana w możliwie najprostszym języku logiki deskrypcyjnej (OWL Lite_A, OWL 2 QL [27], maksymalnie OWL 2 RL [27]).
- Każdy element ontologii musi być dobrze udokumentowany (opis słowny co dokładnie oznacza konkretne pojęcie i rola). Każde pojęcie musi zawierać informacje o tym jakie są dopuszczalne powiązania pomiędzy instancjami tego pojęcia z innymi (na podstawie odpowiednich ról/relacji).
- Pojęcia, role oraz ich opisy będą zdefiniowane w językach: angielskim oraz polskim.
- Ontologia będzie mogła być rozszerzana poprzez użytkowników systemu. Należałoby ograniczyć dopuszczalne rozszerzenia, ze względu na możliwość naruszenia spójności ontologii, np. można dodawać tylko i wyłącznie pojęcia/role bez korzystania z konstruktorów DL – podejście „słownikowe” (osobna ontologia jako zbiór używanych terminów, bez wyraźnych zależności oraz aksjomatów). W przeciwnym wypadku wymagane byłoby użycie narzędzia wnioskującego, które sprawdza nowo utworzoną ontologię i w przypadku wystąpienia błędu wskazuje jego przyczynę lub proponuje rozwiązanie problemu.

3.3.3. Semantyczny model ontologiczny – ontologia Prosecco

Semantyczny model ontologiczny składa się z następujących elementów (ontologii dziedzinowych):

1. Organization – ontologia opisująca rodzaje organizacji/firm, np. Fundacja, Firma prywatna, Małe przedsiębiorstwo itd. Na etapie implementacji ontologii częściowo skorzystamy z ontologii GZ, ORG oraz PROTON.
2. OrganizationStructure – ontologia opisująca możliwe struktury występujące w organizacjach – podział na departamenty, oddziały itd.. Na etapie implementacji ontologii częściowo skorzystamy z ontologii ORG oraz IntelLEO.
3. Employee – ontologia opisująca stanowiska pracownicze oraz ich obowiązki i odpowiedzialności.
4. Person – ontologia opisująca osoby, informacje o ich edukacji, doświadczeniu itd.
5. Resource – ontologia opisująca zasoby firmy – powiązana z ontologią Employee (pracownik to też zasób firmy).
6. Document – ontologia opisująca dokumenty oraz ich składowe, np. Faktura, Umowa, Dokument_księgowy.
7. Project – ontologia opisująca elementy związane z realizacją projektów. Uwzględnia również podział projektów oraz ich uczestników.
8. Activity – ontologia opisująca wszelkiego rodzaju aktywności w formie ról, np. wystawienie_faktury lub w formie pojęć, np. Negocjacja_umowy.
9. Process – ontologia służąca do opisu procesów.

10. BPMN – ontologia Business Process Modeling Notation 2.0¹⁴ służąca do opisywania procesów biznesowych zgodnie ze standardem BPMN w wersji 2.0.
11. Object – ontologia opisująca elementy, które nie pasują tematycznie do żadnej z powyższych, np. burza_mózgów.
12. Prosecco – główna ontologia integrująca wszystkie powyższe. Docelowo zawierać będzie reguły zidentyfikowane w ramach zadania 3.
13. User – ontologia, która nie będzie zawierać żadnych pojęć oraz ról. Jest ona przeznaczona dla potencjalnego użytkownika systemu, który w jej ramach będzie mógł rozszerzać model ontologiczny o dodatkowe pojęcia, role a także reguły.



Rysunek 8: Struktura ontologii Prosecco.

Rysunek 8 przedstawia strukturę ontologii Prosecco oraz wzajemne zależności pomiędzy poszczególnymi elementami składowymi (ontologiami). Przecinanie się poszczególnych elementów oznacza, że będą one wykorzystywać wspólne pojęcia

¹⁴<http://www.scch.at/en/ontologydownload>

i/lub role (będą subsumowane przez odpowiedni element ontologii Prosecco). Wczytując całą ontologię Prosecco będziemy mieć dostęp do wszystkich pojęć oraz ról. Każda z ontologii będzie tematycznie ograniczona do danej dziedziny. Ontologia *User* nie przecina się z żadną inną ontologią dlatego, że nie zawiera żadnych pojęć oraz ról. Jednakże ma dostęp do wszystkich innych zdefiniowanych pojęć oraz ról (w ramach całej ontologii Prosecco).

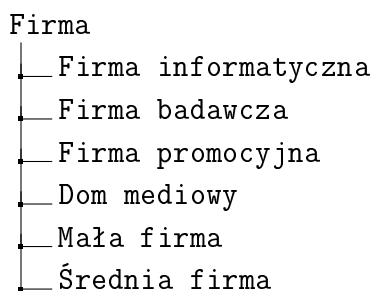
3.3.4. Taksonomiczny model pojęć

Niniejszy rozdział stanowi propozycję taksonomicznego modelu pojęć, który swoim zakresem obejmuje uogólnione pojęcia oraz relacje pochodzące z przeprowadzonych wywiadów oraz analiz istniejących ontologii. Dodatkowo, zawiera pojęcia oraz relacje, które nie występowały w ontologiach pochodzących z wywiadów (bądź innych) a wynikają z metodologii tworzenia ontologii. Następne podrozdziały obejmują kolejne taksonomie pojęć i relacji odpowiadające odpowiednim elementom ontologii Prosecco.

3.3.4.1. Struktura organizacyjna oraz hierarchia stanowisk w firmie

Taksonomie przedstawione w niniejszym rozdziale znajdują się w następujących elementach ontologii Prosecco: Organization, OrganizationStructure oraz Employee.

Ogólny podział firm:



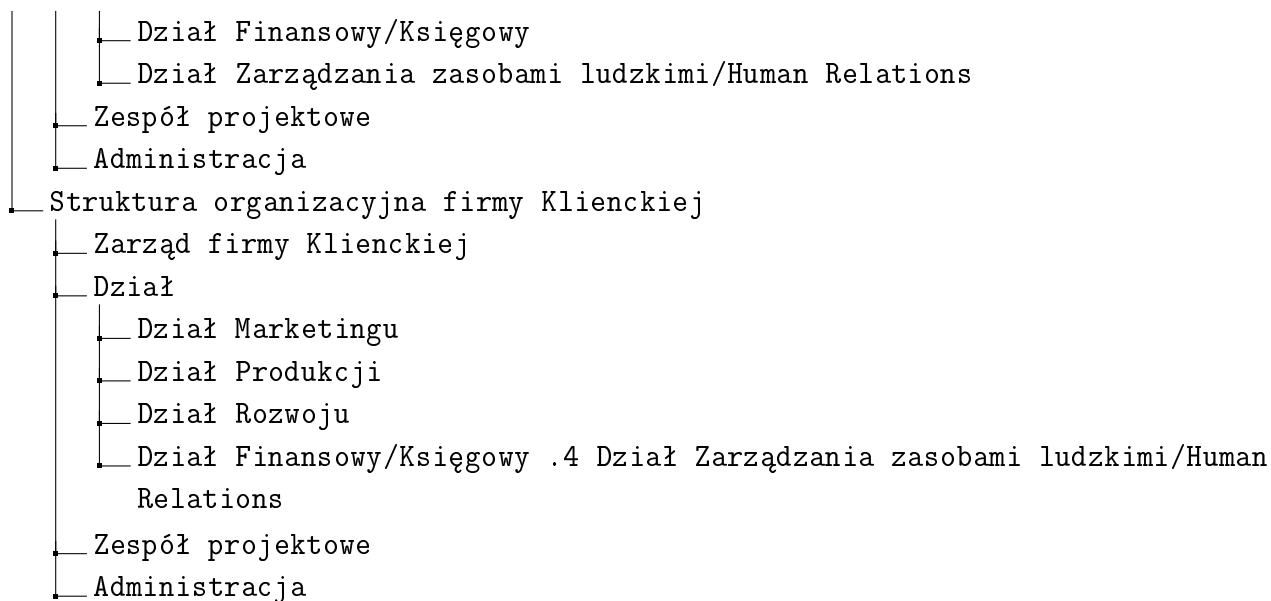
- | Duża firma
- | MSP

Atrybuty (parametry) firmy:

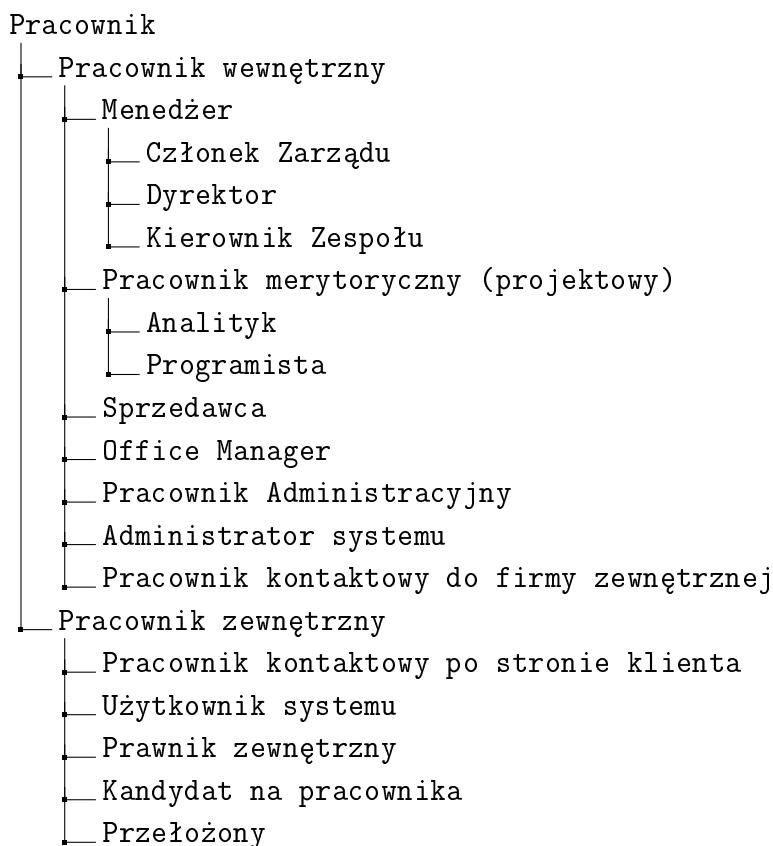
- Atrybuty Firmy
 - | Lokalizacja
 - | Adres
 - | Adres WWW
 - | Rodzaj działalności gospodarczej
 - | Logo
 - | Znak firmowy
 - | Mała firma
 - | Kapitał
 - | Przychód
 - | Sprzedaż (jako składnik przychodu)
 - | Sprzedaż bezpośrednia
 - | Dotacja
 - | Koszt
 - | Dochód
 - | Zysk
 - | Podatek
 - | CIT
 - | VAT
 - | Wielkość zatrudnienia
 - | NIP
 - | REGON
 - | Nr VAT EU

Struktura organizacyjna firmy:

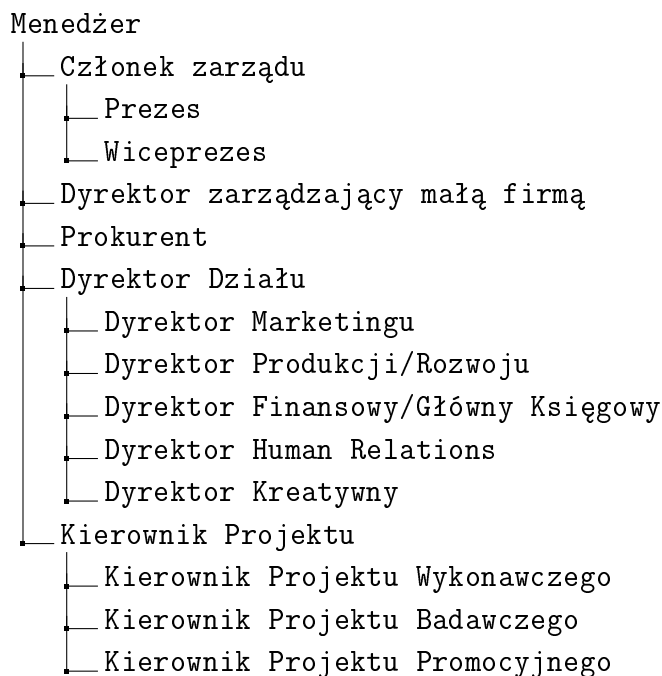
- Struktura organizacyjna
 - | Struktura organizacyjna firmy Wykonawczej
 - | Zarząd firmy wykonawczej
 - | Dział
 - | Dział Marketingu
 - | Dział Produkcji
 - | Dział Rozwoju



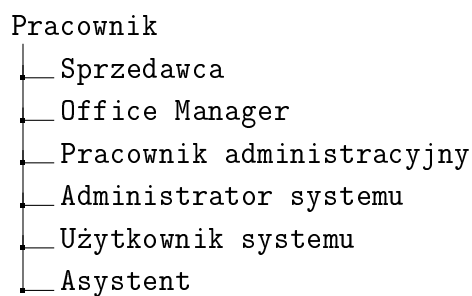
Informacje dotyczące pracownika:



Stanowiska struktury zarządczej



Pracownicy struktury marketingowej i administracyjnej:



3.3.4.2. Czynności i pojęcia dotyczące przedmiotu i kontekstu czynności

Zbiór pojęć przedstawiony w niniejszym rozdziale znajdzie się w elemencie Activity ontologii Prosecco. W kolejnych taksonomiach przyjęto założenie, że byty wydarzające mogą być pojęciami a nie relacjami.

Zarządcza decyzja projektowa:

- Zarządcza decyzja projektowa
 - |_ Zatwierdzenie specyfikacji
 - |_ Zatwierdzenie odbioru modułu
 - |_ Zatwierdzenie protokołu rozbieżności
 - |_ Zatwierdzenie odbioru systemu

Standardowa konkretna decyzja projektowa:

- Konkretna decyzja projektowa
 - |_ Stworzenie zespołu
 - |_ Rozbicie projektu na zadania
 - |_ Przydział zadań
 - |_ Przypisania pracownika do zadania

Czynności wynikające z obowiązków Kierownika Projektu:

- Czynność Kierownika Projektu
 - |_ Planowanie budżetu
 - |_ Utrzymanie kontaktu z klientem
 - |_ Utrzymanie kontaktu ze sponsorem
 - |_ Monitorowanie statusu projektu
 - |_ Odpowiednie motywowanie członków zespołu
 - |_ Zarządzanie stronami zaangażowanymi w projekt
 - |_ Zarządzanie zespołem projektowym
 - |_ Zarządzanie ryzykiem projektu
 - |_ Zarządzanie budżetem projektu
 - |_ Zarządzanie konfliktami

Czynności dotyczące zadania przedwykonawczego:

- Czynność przedwykonawcza
 - |_ Złożenie wniosku
 - |_ Rozpatrzenie zapytania ofertowego pod względem technicznym przez Account Manager/Dyrektor Rozwoju/Kierownik Projektu w sektorze X

- Rozpatrzenie zapytania ofertowego pod względem finansowo-organizacyjnym przez Dyrektora Zarządzającego/Dyrektora Produkcji/Dyrektora działu w sektorze
- X
- Wycena do zawarcia kontraktu
 - Wycena godzinowa
 - Wycena kwotowa
- Wykonanie konsultacji zewnętrznej (w przypadku PromoAgency u OrcaTransform)
- Wykonanie studium wykonalności projektu
- Napisanie briefu
- Negocjacja warunków umowy
- Akceptacja klienta
- Podpisanie umowy

Czynności dotyczące zadania końcowego:

Czynność końcowa

- Protokół częściowego odbioru
- Wykonanie zaleceń protokołu odbioru/dokonanie poprawek
- Wykonanie protokołu zakończenia projektu
- Wystawienie faktury
- Archiwizacja projektu
- Wykonanie raportu poprojektowego
- Otrzymanie płatności

3.3.4.3. Dokumenty w firmie

Taksonomie przedstawione w niniejszym rozdziale znajdują się w elemencie Document ontologii Prosecco.

Dokumenty i ich rodzaje:

Dokument

- Dokument zarządczy
 - Statut
 - Regulamin
 - Umowa
 - Aneks do umowy
 - Umowa o współpracy
 - List intencyjny

- └─ Umowa NDA
- └─ Oferta
- ─ Dokument projektowy
 - └─ Formalna specyfikacja wymagań
 - └─ Protokół odbioru
 - └─ Brief
 - └─ Notatka ze spotkania
- ─ Dokument administracyjny
 - ─ Dokument administracyjny finansowy
 - └─ Polecenie dokonania płatności
 - └─ Faktura
 - └─ Rachunek
 - └─ Druk-Zaliczka
 - └─ Druk-Delegacja
 - └─ Druk zamówienia
 - └─ Lista zapotrzebowania na zakupy
 - └─ Lista wydatków kartą firmową
 - ─ Dokument administracyjny Human Relations
 - └─ Umowa cywilno-prawna
 - └─ Umowa o pracę
 - └─ Umowa o dzieło
 - └─ Umowa zlecenie
 - └─ Aneks
 - └─ Instrukcja
 - └─ Kwestionariusz osobowy
 - └─ Lista płac
 - └─ Dokument ewaluacji pracownika
 - └─ Świadectwo pracy
 - └─ Lista zwolnień lekarskich
 - └─ Plan urlopów
 - └─ Wniosek urlopowy
 - └─ Wniosek premiowy
 - └─ Druk zamówienia
 - └─ Zgłoszenie rekrutacyjne
 - └─ Zgłoszenie serwisowe i reklamacyjne
 - └─ Harmonogram pracy
 - └─ Dokument podatkowy
 - └─ Korespondencja zewnętrzna
 - └─ Korespondencja wewnętrzna

- └─ Protokół
- └─ Dokument elektroniczny
 - └─ Faktura
 - └─ Lista faktur
 - └─ Lista zapotrzebowania na zakupy
 - └─ Dokumenty podatkowe
 - └─ Dokumenty intranetu pracowniczego
 - └─ Poczta elektroniczna
 - └─ Wiadomość
 - └─ Regulamin usługi
 - └─ Kalendarz zdarzeń
 - └─ Dokument Wiki

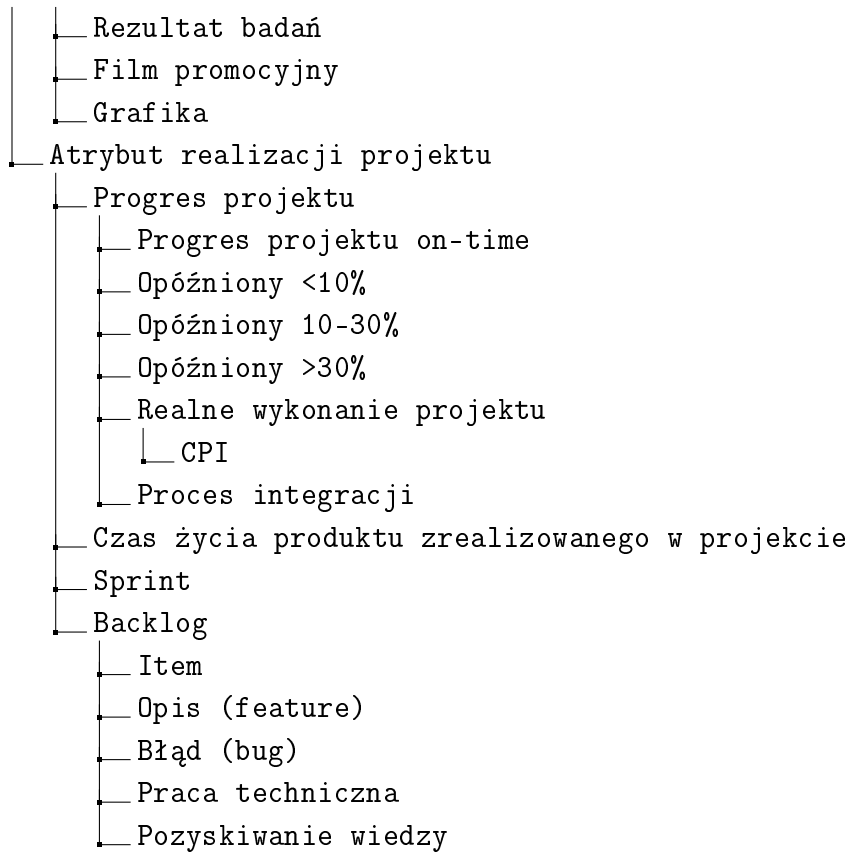
3.3.4.4. Pojęcia związane z projektowaniem oraz samym projektem

Taksonomie przedstawione w niniejszym rozdziale znajdują się w elemencie Project ontologii Prosecco.

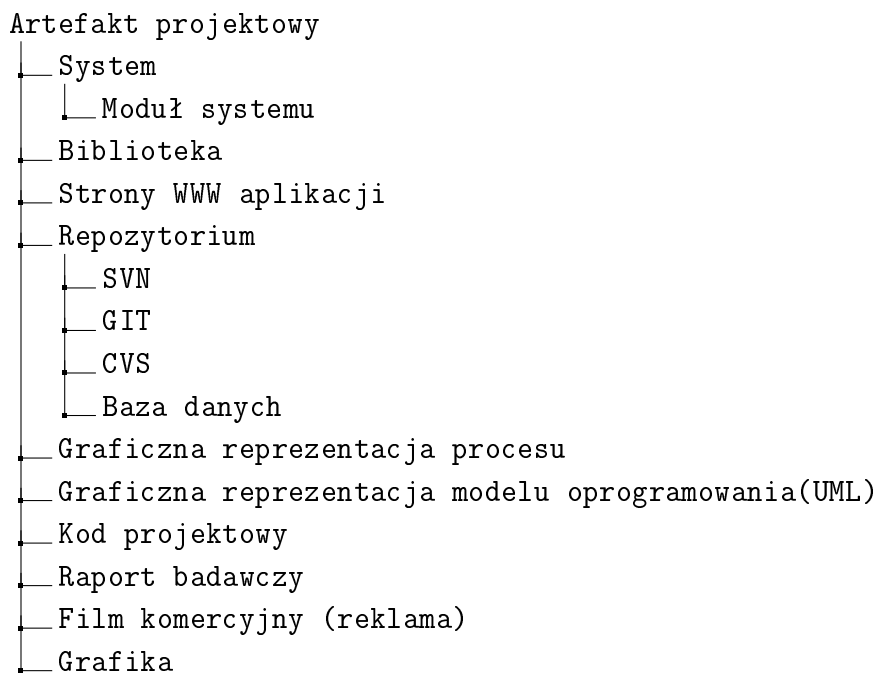
Projekt oraz elementy z nim związane:

- Thing
 - └─ Projekt
 - └─ Projekt miękki
 - └─ Badanie rynku
 - └─ Projekt twardy
 - └─ Wykonanie systemu informatycznego/projektu
 - └─ Wykonanie materiału reklamowego (mniej twardy niż powyższy)
 - └─ Forma finansowania
 - └─ Projekt komercyjny
 - └─ Projektu strukturalny
 - └─ Projekt unijny
 - └─ Projekt wewnętrzny
 - └─ Inwestycja
 - └─ Metodologia wykonania projektu
 - └─ SCRUM
 - └─ Waterfall
 - └─ Pracownicy w projekcie
 - └─ Kierownik projektu
 - └─ Kierownik Zespołu
 - └─ Specjalista ds promocji

- Specjalista ds rekrutacji
- Specjalista ds rozliczeń finansowych
- Specjalista ds monitoringu
- Księgowy/a projektu
- Pracownik administracyjny
- Wykonawca projektu
- Inny pracownik
- Przebieg projektu
 - Etap
 - Zadanie
 - Podzadanie
 - Odpowiedzialność
 - Odpowiedzialność za wykonanie projektu w terminie, zgodnie z budżetem i wymaganiami
 - Odpowiedzialność za rozliczenie projektu
 - Odpowiedzialność za wykonanie zadania projektu
 - Czas trwania
 - Harmonogram realizacji projektu
 - Wykres Gantt'a
 - Termin realizacji projektu
 - Rezerwy czasowe
 - Kamień milowy projektu
 - Zdarzenie
 - Deliverable (w projekcie unijnym)
 - Wymaganie
 - Specyfikacja
- Atrybut budżetu projektu
 - Biznes plan
 - Okres budżetowania
 - Rodzaj kosztów
 - Miejsce powstawania kosztów
 - Nośnik kosztów
 - Budżet kosztów
 - Możliwość pozyskania środków
 - Płynność finansowa
 - Zatwierdzenie budżetu
 - Koszt kwalifikowany
- Atrybut wyniku projektu
 - Program - aplikacja



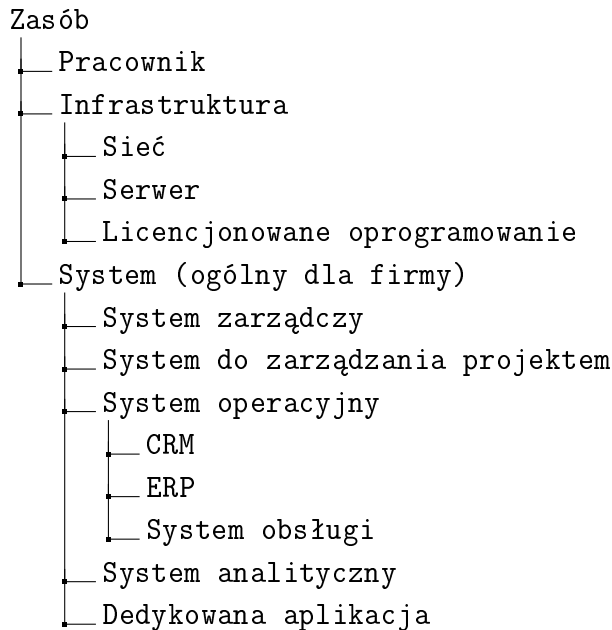
Artefakty projektowe



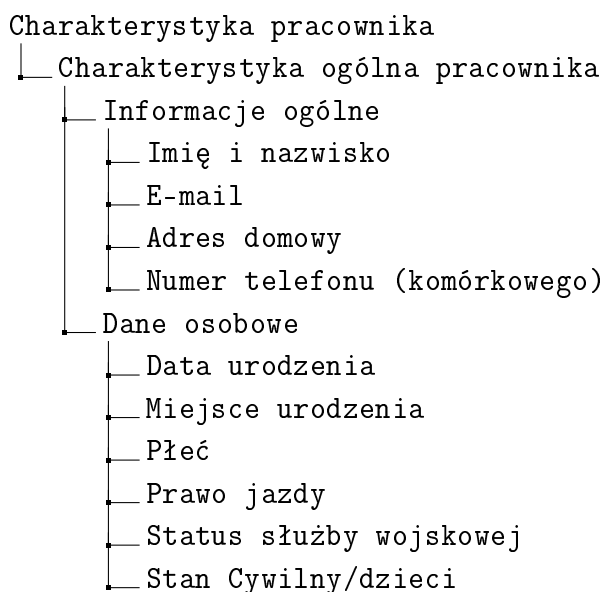
3.3.4.5. Zasoby

Taksonomie przedstawione w niniejszym rozdziale znajdują się w następujących elementach ontologii Prosecco: Resource, Person oraz Employee.

Pojęcia związane z zasobami:



Atrybuty pracownika jako zasobu według Resume RDF [28] oraz Description of a Career [29]:



- Informacje o wykształceniu
 - Nazwa uczelni oraz wydziału
 - Specjalność
 - Data ukończenia studiów
- Informacje o doświadczeniu zawodowym
 - Przeszły projekt
 - Referencja
 - Język
- Kwalifikacje
 - Obszar kompetencji projektowania systemu i programowania
 - Obszar kompetencji materiałów reklamowych
 - Projekt
 - Certyfikat
- Charakterystyka projektowa pracownika
 - Funkcja/Stanowisko - obowiązki, komu podlega w firmie, kim zarządza
 - Kwalifikacja
 - Status premiowy
 - Przynależność do projektu
 - Stawka godzinowa
 - Aktualna zajętość w projekcie
 - Procent zaangażowania
 - Przyszła zajętość w projekcie
 - Dostępność w nowym projekcie w czasie
 - Zarobki, premie
- Charakterystyka zdolności do pracy w firmie

3.3.4.6. Procesy w firmie

Taksonomie przedstawione w niniejszym rozdziale znajdują się w elemencie Process ontologii Prosecco.

Szczegółowe procesy projektowe:

- Szczegółowy proces projektowy
 - Odbycie spotkania
 - Odbycie spotkania informacyjnego
 - Odbycie spotkania merytorycznego
 - Odbycie spotkania typu standup
 - Wdrożenie

- └─ Wdrożenie na platformie demo
- └─ Wdrożenie na platformie produkcyjnej
- └─ Wdrożenie na platformie testowej
- └─ Wykonanie sprintu
 - └─ Wykonanie backloga
 - └─ Usunięcie błędów
- └─ Testowanie
 - └─ Testowanie kodu przez współpracownika
 - └─ Testowanie kodu przez użytkownika
- └─ Wykonanie szkolenia

Procesy projektowe w firmie implementowane w narzędziu:

Proces atomowy

- └─ Proces zarządzania zasobami ludzkimi
 - └─ Dodanie pracownika
 - └─ Dodanie kandydata
 - └─ Dodanie współpracownika
 - └─ Zarządzanie urlopami
 - └─ Zarządzanie środkami trwałymi
- └─ Procesy związane ze sprzedażą (CRM)
 - └─ Dodanie kontrahenta
 - └─ Dodanie kontraktu
 - └─ Dodanie kontaktu z kontrahentem
 - └─ Utworzenie płatności
 - └─ Dodanie aktywności
- └─ Procesy projektowe (PMS)
 - └─ Utworzenie nowego projektu
 - └─ Dodanie wymagania
 - └─ Dodanie zadania
 - └─ Dodanie zgłoszenia
 - └─ Proces raportowania postępu prac
- └─ Inny proces
 - └─ Rejestracja w systemie

Procesy administracyjne:

Rekrutacja

- └─ Dokonanie zgłoszenia rekrutacyjnego

- └─ Ogłoszenie o poszukiwaniu pracownika
- └─ Dokonanie rozmowy kwalifikacyjnej
- └─ Dokonanie selekcji kandydatów
- └─ Dokonanie przydziału do działu
- └─ Podpisanie umowy o pracę na okres
 - └─ Podpisanie umowy o pracę na okres próbny
 - └─ Podpisanie umowy o pracę na czas określony
 - └─ Podpisanie umowy o pracę na czas nieokreślony
- └─ Podpisanie umowy zlecenia/ o dzieło
- └─ Zgłoszenie pracownika do ZUS
- └─ Wykonanie czynności niezbędnych do przyjęcia pracownika
 - └─ Wykonanie badań lekarskich
 - └─ Szkolenie
 - └─ Szkolenie BHP

Zwolnienie/odejście pracownika:

- Rozwiązanie współpracy
 - └─ Dokonanie negatywnej oceny pracy
 - └─ Nieprzedłużenie umowy
 - └─ Napisanie uzasadnienia rozwiązania umowy o pracę
 - └─ Rozwiązanie umowy o pracę przez pracodawcę
 - └─ Rozwiązanie umowy o pracę przez pracownika
 - └─ Wypowiedzenie
 - └─ Wypisanie świadectwa pracy
 - └─ Archiwizacja danych pracownika

3.3.4.7. Inne elementy

Taksonomie przedstawione w niniejszym rozdziale znajdują się elemencie Object ontologii Prosecco.

- Inne
 - └─ Zakup
 - └─ Mały zakup
 - └─ Duży zakup
 - └─ Biuro
 - └─ Kanał sprzedaży firmy
 - └─ Dostawca
 - └─ Stały dostawca

3.3.4.8. Zbiór relacji pomiędzy pojęciami

W niniejszym rozdziale przedstawiono relacje, które wystąpią w docelowej ontologii Prosecco. Poniższe relacje umożliwiają powiązanie ze sobą instancji pojęć pochodzących z różnych elementów ontologii. Stąd podjęto decyzję o ich wspólnym zaprezentowaniu. Relacje oraz przykłady ich zastosowania (w przykładach wykorzystano pojęcia ontologiczne, stąd forma gramatyczna w niektórych przypadkach jest niepoprawna):

- *podlega*: Dyrektor Działu *podlega* Zarząd, Kierownik Projektu *podlega* Zarząd,
- *należy do*: Pracownik *należy do* Zespół Projektowy, Pracownik *należy do* Dział,
- *pracuje w*: Pracownik *pracuje w* Firma,
- *kieruje*: Kierownik *kieruje* Projekt,
- *składa się z*: Zarząd *składa się z* Członek Zarządu,
- *zatwierdza*: Kierownik *zatwierdza* Decyzja projektowa,
- *dotyczy*: Polecenie dokonania płatności *dotyczy* Faktura,
- *rozlicza*: Zarząd *rozlicza* Projekt,
- *wydaje*: Zarząd *wydaje* Zarządcza decyzja projektowa,
- *obciąża*: Faktura *obciąża* Firmę,
- *przydziela*: Kierownik *przydziela* Zadanie,
- *jest wynikiem*: Projekt *jest wynikiem* Zlecenie.
- *zawiera*: Zadanie *zawiera* Czynność.

3.3.4.9. Uwagi

Pomimo obszernej listy pojęć warto zaznaczyć, że bez wykorzystania reguł (np. wyrażonych w języku Semantic Web Rule Language¹⁵) nie da się uzyskać instancji pewnych złożonych pojęć bądź relacji. Przykładowo, załóżmy że chcemy zdefiniować pojęcie *KomitetSterujący*, który definiujemy słownie jako komitet składający się z kierowników projektu pracujących w różnych firmach (z każdej firmy może być tylko jeden przedstawiciel). W logice deskrypcyjnej definiujemy pojęcie *KomitetSterujący*, które składa się z co najmniej 2 kierowników Projektu:

$$\begin{aligned} \text{KomitetSterujący} &\equiv \\ &\geq 2 \text{ składaSięZ}. \\ &(\text{Kierownik} \sqcap \exists \text{ kieruje.Projekt} \sqcap \exists \text{ pracujeW.Firma}) \\ &\sqcap 1 \text{ dotyczy.Projekt} \end{aligned}$$

Powyższy zapis oznacza, że komitet sterujący to byt, na który składa się co najmniej dwóch kierowników Projektu, przy czym komitet dotyczy tylko i wyłącznie jednego Projektu (relacja *dotyczy* jest relacją funkcyjną). Nie jest to jednak najlepsze rozwiązanie, ponieważ nie sprawdzi ono czy dwóch kierowników pochodzi z tej samej czy różnej firmy czy też nie – nie da się tego wyraźnie zaznaczyć w logice deskrypcyjnej. Oznacza to, że nasza definicja nie jest poprawna. Można w taki przypadku posłużyć się regułą umożliwiającą wywnioskowanie, że dany Komitet jest Komitetem Sterującym. Reguła zadziała dla co najmniej dwóch Kierowników z różnych Firm,

¹⁵SWRL, <http://www.w3.org/Submission/SWRL/>

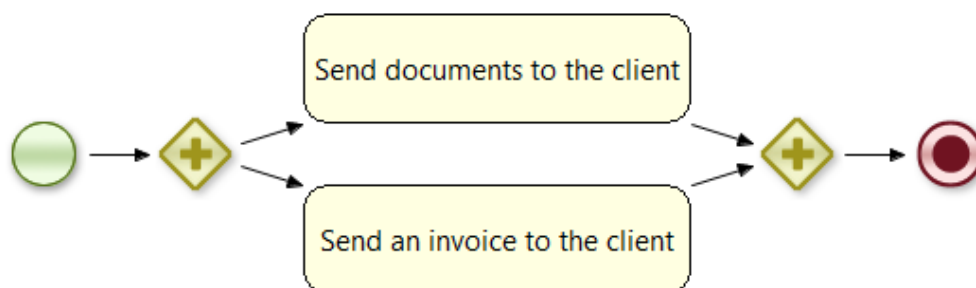
kórzcy kieruj dokadnie tym samym Projektem:

$Komitet(?x)$, $skadaSięZ(?x, ?y1)$, $skadaSięZ(?x, ?y2)$, $Kierownik(?y1)$, $Kierownik(?y2)$,
 $kieruje(?y1, ?p)$, $kieruje(?y2, ?p)$, $Projekt(?p)$, $pracujeW(?y1, ?f1)$, $pracujeW(?y2, ?f2)$,
 $dotyczy(?x, ?p)$, $?y1 \neq ?y2$, $?f1 \neq ?f2$
 $\rightarrow KomitetSterujcy(?x)$

Wad takiego podejcia jest jawne wystpienie instancji pojęcia Komitet - dopiero na tej podstawie możemy wywnioskować zakwalifikowanie Komitetu jako instancji klasy KomitetSterujcy. W ramach dalszych prac w projekcie (np. Zadanie 5), podczas implementacji ontologii, należy rozważyć, które pojęcia należałoby definiować w powyższy sposób w celu uzyskania instancji danego pojęcia/relacji.

3.3.5. Przykad zastosowania taksonomii w procesach

Opracowany model taksonomiczny znajdzie zastosowanie m.in. w semantycznym opisie procesów. Rysunek 9 prezentuje przykadowy proces skadajcy się z 6 elementów. Proces ma za zadanie opisywać przesył rowno rodzaju dokumentów do klienta. Zaznaczono przy tym wysłanie dokumentów (*Send documents to the client*) oraz faktury (*Send an invoice to the client*). Oba te zadania mają być wykonywane równolegle.



Rysunek 9: Przykadowy proces.

Diagram procesu przedstawiony na rysunku 9 zawiera następujące elementy wraz z identyfikatorami:

- ID_1 Start_Event - początek procesu,
- ID_2 Task(Send documents to the client) - zadanie wysłania dokumentów do klienta,
- ID_3 Task(Send an invoice to the client) - zadanie wysłania faktury do klienta,
- ID_4 Gateway_AND_Start - początek równoległych zadań,
- ID_5 Gateway_AND_Stop - koniec (połączenie) równoległych zadań,
- ID_6 End_Event - koniec procesu.

Powyższe elementy możemy opisać semantycznie wykorzystując zaproponowany model taksonomiczny (ontologię Prosecco). W wyniku jego zastosowania zaprezentowane elementy procesu wyrazić można za pomocą następujących trójek RDF¹⁶:

ID_1	rdf:type	bpmn:StartEvent
ID_2	rdf:type	bpmn:SendTask
ID_3	rdf:type	bpmn:SendTask
ID_4	rdf:type	bpmn:GatewayDirectionDiverging
ID_5	rdf:type	bpmn:GatewayDirectionConverging
ID_6	rdf:type	bpmn:EndEvent

Wykorzystując etykiety zadań (*Send...*) można również doprecyzować ich znaczenie poprzez odpowiednią klasyfikację:

ID_2	prosecco:zawiera	_ID_2_1
_ID_2_1	rdf:type	activity:Czynność

¹⁶<http://www.w3.org/RDF/>

```
ID_2      prosecco:zawiera  _ID_2_2
_ID_2_2   rdf:type           document:Dokument

ID_3      prosecco:zawiera  _ID_3
_ID_3     rdf:type           document:Faktura
```

Na podstawie zaprezentowanych zbiorów trójek system wnioskujący (wykorzystujący model taksonomiczny) może wywnioskować następujące (nowe) trójki RDF:

```
ID_1      rdf:type      bpmn:CatchEvent
ID_1      rdf:type      bpmn:Event
ID_1      rdf:type      bpmn:InteractionNode
ID_1      rdf:type      bpmn:BaseElement

ID_2      rdf:type      bpmn:Task
ID_2      rdf:type      bpmn:Activity
ID_2      rdf:type      bpmn:FlowNode
ID_2      rdf:type      bpmn:FlowElement
ID_2      rdf:type      bpmn:BaseElement

_ID_2_1   rdf:type      bpmn:Activity
_ID_2_1   rdf:type      bpmn:FlowNode
_ID_2_1   rdf:type      bpmn:FlowElement
_ID_2_1   rdf:type      bpmn:BaseElement

ID_3      rdf:type      bpmn:Task
ID_3      rdf:type      bpmn:Activity
ID_3      rdf:type      bpmn:FlowNode
ID_3      rdf:type      bpmn:FlowElement
ID_3      rdf:type      bpmn:BaseElement
```

```
_ID_3    rdf:type    document:Dokument

ID_4     rdf:type    bpmn:Gateway
ID_4     rdf:type    bpmn:FlowNode
ID_4     rdf:type    bpmn:FlowElement
ID_4     rdf:type    bpmn:BaseElement

ID_5     rdf:type    bpmn:Gateway
ID_5     rdf:type    bpmn:FlowNode
ID_5     rdf:type    bpmn:FlowElement
ID_5     rdf:type    bpmn:BaseElement

ID_6     rdf:type    bpmn:ThrowEvent
ID_6     rdf:type    bpmn:Event
ID_6     rdf:type    bpmn:InteractionNode
ID_6     rdf:type    bpmn:BaseElement
```

Przedstawiony przykład pozwala na lepsze zrozumienie idei włączania taksonomii w mechanizm konstrukcji procesów. Wykorzystując mechanizmy wnioskowania logicznego docelowy system mógłby proponować użytkownikowi uściślenie o jakie konkretnie dokumenty chodzi w zadaniu *Send documents to the client*. Zwłaszcza, że w w procesie zdefiniowane jest konkretne zadanie przesłania faktury, która również jest rodzajem dokumentu.

Istotnym elementem jest zagadnienie dotyczące etykiet zadań. Ograniczając się do listy zadań do wyboru mamy pewność, że ich semantyka będzie prawidłowo zdefiniowana, jednakże nie jesteśmy w stanie przewidzieć wszystkich możliwych etykiet. Dając użytkownikowi możliwość tworzenia własnych etykiet (w elemencie User ontologii Prosecco) należałoby zaproponować mechanizmy sprawdzania poprawności tak dodanych etykiet. Innym, potencjalnym zastosowaniem, jest wykorzystanie

mechanizmów analizy języka naturalnego i automatyczne opisywanie etykiet za pomocą pojęć i relacji z ontologii. Kwestia ta zostanie rozwiązana w kolejnych zadaniach projektu Prosecco.

4. Efekty zadania

Wprowadzenie semantycznego opisu taksonomicznego ma pozwolić na uspoźnienie i ujednoznacznienie pojęć używanych w systemie zarządzania małymi i średnimi przedsiębiorstwami. Taki opis pozwoli na uproszczenie procesu zarządzania co bezpośrednio przełoży się na obniżenie jego kosztów. Pozostałe efekty przeprowadzonych i opisanych w niniejszym raporcie prac przedstawiają się następująco:

1. Docelowo ontologia będzie modelem danych i będzie mogła służyć do odpytania repozytorium; natomiast w procesach modelowana będzie dynamika. Możliwy jest wariant docelowego systemu bez procesów, a jedynie z ontologią.
2. W zakres pytań kompetencyjnych docelowej ontologii wchodzi pytania typu CRUD: odczyt, wyszukiwanie z filtrami, warunkami nałożonymi na relacje. Zatem struktura docelowej ontologii powinna zawierać
 - (a) "core CRUD ontology",
 - (b) ontologię artefaktów firmy wytwarzanych w procesach,
 - (c) ontologię wysokopoziomową dla procesów (obejmująca BPMN, EPC itp.)
3. W szerszej perspektywie wartość dodana semantyzacji polega przede wszystkim na konfigurowalności po stronie użytkownika (dodawaniu nowych obiektów, ich relacji z innymi, umożliwienie operacje CRUD).

5. Załączniki

Poniżej przedstawiono tekstową reprezentację prototypów taksonomii analizowanych firm (bez metapredykatów, przykładów i pomocy) oraz ich wizualizacje. Pełne źródła podlinkowane są w wiki projektu na stronie: <http://wiki.softhis.com/prosecco:zadania:zadanie2>.

5.1. Prototyp taksonomii PromoAgency

```
%% Klasy %  
c(klient, 'Klient').  
c(pracownik, 'Pracownik').  
c(new_business, 'New Business').  
c(acc_manager, 'Account Manager').  
c(dyrektor_zarz, 'Dyrektor Zarządzający').  
c(office_manager, 'Office Manager').  
c(acc_exec, 'Account Executive').  
c(pm, 'Project Manager').  
c(traffic_manager, 'Traffic Manager').  
c(copywriter, 'Copywriter').  
c(programista, 'Programista').  
c(grafik, 'Grafik/Designer').  
c(spec_social, 'Specjalista ds. Social Media').  
c(dyrektor_kreat, 'Dyrektor kreatywny').  
c(dyrektor_art, 'Dyrektor artystyczny').  
c(webdeveloper, 'Web Developer').  
c(webdesigner, 'Web Designer').  
c(zespol, 'Zespół').
```

```
%% Działy, jednostki, miejsca                                %
c(dzial, 'Dział').
c(client_service, 'Client Service').
c(dz_strategii, 'Dział Strategii').
c(dom_mediowy, 'Dom mediowy').

%% Artefakty                                                %
c(brief, 'Brief'). %dokument zawierający konkretne informacje dot. akcji,
    produktu i zakresu, wraz z deadlinem.
c(faktura, 'Faktura').
c(harmonogram, 'Harmonogram').
c(kanal_sprzedazy, 'Kanał sprzedaży firmy').
c(milestone, 'Milestone (kamień milowy)'). % W opisie procesu Realizacja
    projektu jest wpisany jako etap procesu (?)
c(notatka, 'Notatka (np. po spotkaniu)').
c(oferta, 'Oferta').
c(sprzedaz_bezposr, 'Sprzedaż bezpośrednia').
c(zapytanie_ofertowe, 'Zapytanie Ofertowe').
c(zapr_do_sprzedazy, 'Zaproszenie do sprzedaży').
c(zapytanie_rekomendacja, 'Zapytanie z rekomendacji').
c(linia, 'Linia').
c(mailing, 'Mailing').
c(landing_page, 'Landing page').

%% Procesy                                                  %
c(proces, 'Proces biznesowy').
```

```
c(debriefing, 'Debriefing').
c(podpisanie_umowy, 'Podpisanie umowy').
c(obsługa_zap_ofert, 'Obsługa zapytania ofertowego').
c(realizacja_prj, 'Realizacja projektu').
c(prowadzenie_kampanii, 'Prowadzenie kampanii').

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Atrybuty %
%% Atrybuty klas %
a(brief, faza, [definiowanie,planowanie,przyg_oferty,zlozenie_eferty]).

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Relacje %
%% Hierarchia klas %

r(new_business jest pracownik).
r(acc_manager jest pracownik).
r(dyrektor_zarz jest pracownik).
r(office_manager jest pracownik).
r(acc_exec jest pracownik).
r(pm jest pracownik).
r(traffic_manager jest pracownik).
r(copywriter jest pracownik).
r(programista jest pracownik).
r(grafik jest pracownik).
r(spec_social jest pracownik).
r(dyrektor_kreat jest pracownik).
```


r(dyrektor_art jest pracownik).

r(webdeveloper jest pracownik).

r(webdesigner jest pracownik).

r(zapytanie_ofertowe jest kanal_sprzedazy).

r(zapr_do_sprzedazy jest kanal_sprzedazy).

r(zapytanie_rekomendacja jest kanal_sprzedazy).

r(sprzedaz_bezposr jest kanal_sprzedazy).

r(client_service jest dzial).

r(dz_strategii jest dzial).

r(debriefing jest proces).

r(podpisanie_umowy jest proces).

r(obsługa_zap_ofert jest proces).

r(realizacja_prj jest proces).

r(prowadzenie_kampanii jest proces).

r(linia jest projekt).

r(rozl_za_efekt jest rozliczenie_z_klientem).

r(rozl_staly_buzdet jest rozliczenie_z_klientem).

r(rozl_kreacja_i_dom_mediowy jest rozliczenie_z_klientem).

% Odpowiedzialności

%

r(dom_mediowy przyjmuje zlecenie_kamp_dom_mediowy).

r(dom_mediowy rozsyla mailing).

r(dom_mediowy rozsyla display).

r(new_business prowadzi sprzedaz_bezposr).

r(new_business prowadzi spotkanie_NB).

r(new_business wyszukuje klient).

r(new_business sporzadza notatka).

r(acc_manager analizuje zapytanie_ofertowe).

r(dyrektor_zarz ocenia_wartosc zapytanie_ofertowe).

r(office_manager obsluguje zgloszenie_mail).

r(acc_manager odpowiada_za kontakt_z_klientem). % w zal. od wielkosci projektu

AM lub PM

r(pm odpowiada_za kontakt_z_klientem). % w zal. od wielkosci projektu AM lub PM

r(pm odpowiada_za termin_wyk_projektu).

r(pm wykonuje archiwizacja_projektu).

r(pm wykonuje backup).

r(programista wykonuje szacowanie_czasu_na_zad).

r(programista wykonuje wykonanie_zadania).

r(programista wykonuje raport_ile_trwalo_zad).

r(traffic_manager wykonuje weryfikacja_dost_zespolu).

r(zespol wykonuje przyg_oferty).

r(zespol wykonuje realizacja_prj).

% Inne zaleznosci

%

r(new_business pracuje_w client_service).

r(acc_manager pracuje_w client_service).

r(office_manager pracuje_w client_service).

r(acc_exec pracuje_w client_service).

r(copywriter pracuje_w dz_strategii).

r(programista pracuje_w dz_strategii).

r(grafik pracuje_w dz_strategii).

r(spec_social pracuje_w dz_strategii).

r(spotkanie_NB prowadzi_do zapytanie_ofertowe).

r(zgloszenie_mail prowadzi_do zapytanie_ofertowe).

r(debriefing zawiera_aktywnosc przyg_oferty).

r(debriefing zawiera_aktywnosc kontakt_z_klientem).

r(debriefing zawiera_aktywnosc weryfikacja_dost_zespolu).

r(debriefing zawiera_aktywnosc przyg_oferty).

r(realizacja_prj sklada_sie_z stworzenie_zespolu).

r(realizacja_prj sklada_sie_z oprac_harmonogramu).

r(realizacja_prj sklada_sie_z spotkania).

r(realizacja_prj sklada_sie_z rozbicie_na_zadania).

r(realizacja_prj sklada_sie_z przydzielenie_zadan).

r(realizacja_prj sklada_sie_z realizacja_zadan).

r(realizacja_prj sklada_sie_z przyg_dokumentacji).

```
r(realizacja_prj sklada_sie_z milestone).
r(realizacja_zadan sklada_sie_z szacowanie_czasu_na_zad).
r(realizacja_zadan sklada_sie_z wykonanie_zadania).
r(realizacja_zadan sklada_sie_z raport_ile_trwalo_zad).
r(milestone sklada_sie_z prot_odbioru). % skrot myslowy z opisu procesu
r(milestone sklada_sie_z archiwizacja_projektu).
r(milestone sklada_sie_z backup).
r(milestone sklada_sie_z wystawienie_faktury).
r(milestone sklada_sie_z wplata_klienta).

r(prowadzenie_kampanii sklada_sie_z przyjecie_zlecenia).
r(prowadzenie_kampanii sklada_sie_z oprac_briefu).
r(prowadzenie_kampanii sklada_sie_z przeslanie_prop_grafiki).
r(prowadzenie_kampanii sklada_sie_z zlecenie_kamp_dom_mediowy). % zlecenie
    kampanii domowi mediowemu
r(prowadzenie_kampanii sklada_sie_z wyniki_kampanii).
r(prowadzenie_kampanii sklada_sie_z rozliczenie_z_klientem).

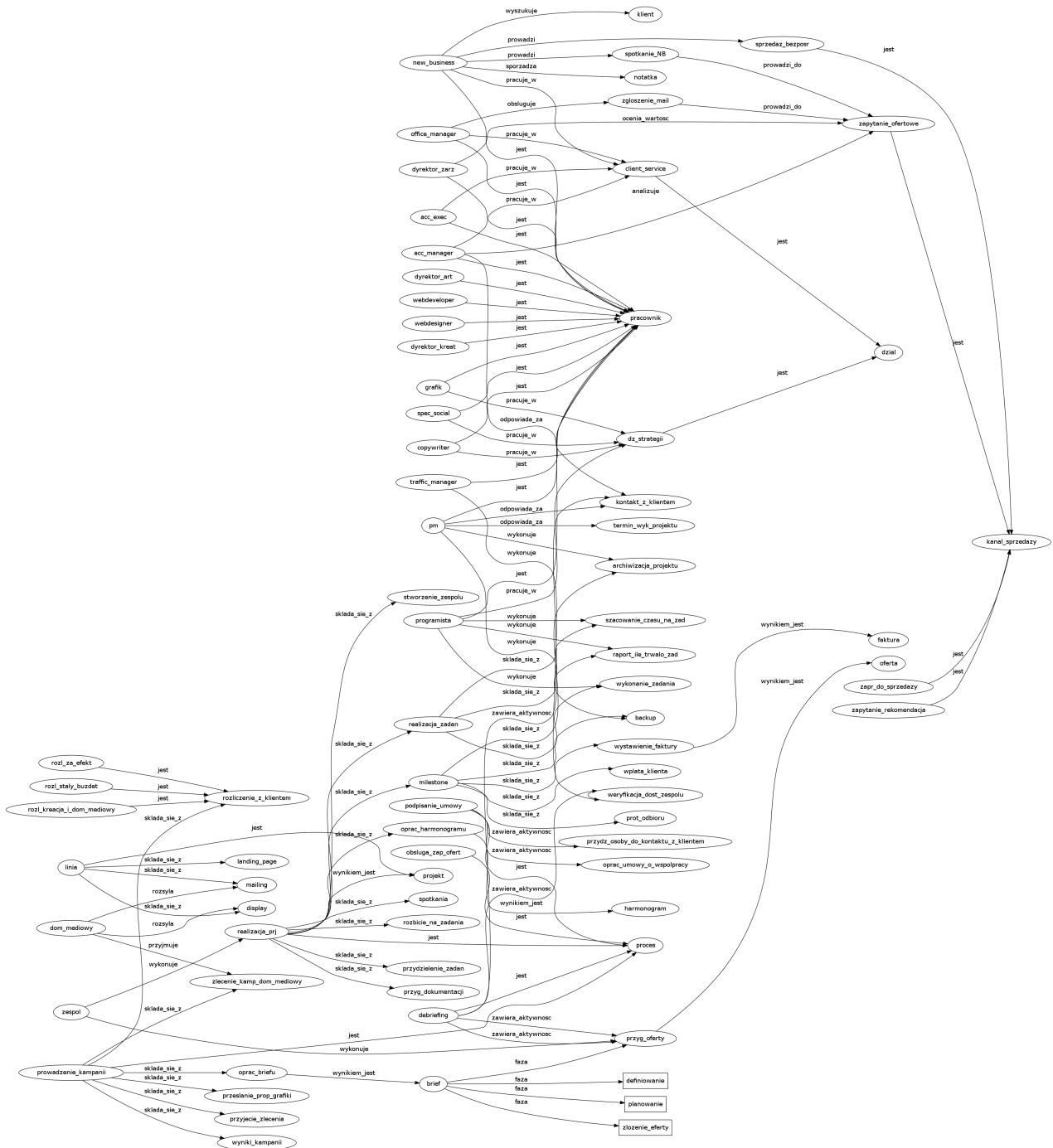
r(podpisanie_umowy zawiera_aktywnosc oprac_umowy_o_wspolpracy).
r(podpisanie_umowy zawiera_aktywnosc przydz_osoby_do_kontakt_z_klientem).

r(przyg_oferty wynikiem_jest oferta).
r(oprac_harmonogramu wynikiem_jest harmonogram).
r(oprac_briefu wynikiem_jest brief).
r(realizacja_prj wynikiem_jest projekt).
r(wystawienie_faktury wynikiem_jest faktura).
```

```
r(linia sklada_sie_z mailing).  
r(linia sklada_sie_z landing_page).  
r(linia sklada_sie_z display).
```

5.2. Prototyp taksonomii IQON Consulting

```
%% Klasy %  
%% Osoby/Role %  
c(pracownik, 'Pracownik').  
c(opiekun_klienta, 'Opiekun klienta').  
c(czlonек_zarzadu, 'Członek Zarządu').  
c(odp_za_projekt, 'Osoba odpowiedzialna za projekt').  
c(prezes, 'Prezes').  
c(zewn_wspolprac, 'Zewnętrzny współpracownik').  
c(sprzedawca, 'Sprzedawca').  
  
%% Działy, jednostki, miejsca %  
c(jednostka_org, 'Jednostka organizacyjna firmy').  
c(sprzedaz, 'Sprzedaż').  
c(projekty, 'Projekty').  
c(zarząd, 'Zarząd').  
  
%% Artefakty %  
c(dokument, 'Dokument').  
c(umowa, 'Umowa').  
c(umowa_o_poufnosci, 'Umowa o poufności').  
c(faktura, 'Faktura').
```



Rysunek 10: Wizualizacja taksonomii firmy PromoAgency.

```
c(lista_zakupow, 'Lista zakupów').
c(lista_faktur, 'Lista faktur').
c(lista_urlopow, 'Lista urlopów').
c(lista_zwolnien_lekarskich, 'Lista zwolnień lekarskich').
c(lista_wydatkow_kartą_firmowa, 'Lista wydatków kartą firmową').

c(zadanie, 'Zadanie').
c(podzadanie, 'Podzadanie').

% Typowe dla firmy:
c(projekt, 'Projekt').
c(projekt_twardy, 'Projekt twardy (inwestycyjny)').
c(projekt_miekki, 'Projekt miękki (nieinwestycyjny)').

%% Aktywności %
c(spotkanie_info, 'Spotkanie informacyjne').
c(spotkanie_meryt, 'Spotkanie merytoryczne').
c(negocjacje_umowy, 'Negocjacje warunków umowy').
c(podpisanie_umowy, 'Podpisanie umowy').
c(wystawienie_faktury, 'Wystawienie i przesłanie faktury').
c(przygotowanie_projektu, 'Przygotowanie projektu').
c(przygotowanie_wniosku, 'Przygotowanie wniosku').
c(akceptacja_klienta, 'Akceptacja klienta (przygotowanego wniosku)').
c(zlozenie_wniosku, 'Złożenie wniosku').
c(poprawki_i_wyjasnienia, 'Przesłanie poprawek i wyjaśnień').

%% Procesy %
```

c(realizacja_zlecenia, 'Realizacja zlecenia').

c(zakupy, 'Zakupy').

c(obsługa_ksiegowa, 'Księgowość').

%%%

%%% Atrybuty %

a(klient, forma_klienta, [fundacja, spółka]).

a(umowa, forma, [papierowa, skan]).

a(wniosek, jezyk, [polski]).

a(przekaz_dok_ksieg, jak_czesto, [co_tydzien, na_biezaco]).

a(dok_ksieg, forma, [papierowa, skan]).

%%%

%%% Relacje %

% Hierarchia klas %

r(realizacja_zlecenia jest proces).

r(zakupy jest proces).

r(obsługa_ksiegowa jest proces).

r(projekt_twardy jest projekt).

r(projekt_miekkie jest projekt).

r(sprzedawca jest pracownik).

r(opiekun_klienta jest pracownik).

r(czlonek_zarzadu jest pracownik).

r(odp_za_projekt jest pracownik).

r(zewn_wspolprac jest pracownik). % opcjonalnie, tak bylo kiedyś, teraz raczej nie

r(prezes jest czlonek_zarzadu).

r(sprzedaz jest jednostka_org).

r(projekty jest jednostka_org).

r(zarząd jest jednostka_org).

r(umowa jest dokument).

r(umowa_o_poufnosci jest dokument).

r(faktura jest dokument).

r(lista_zakupow jest dokument).

r(lista_faktur jest dokument).

r(lista_urlopow jest dokument).

r(lista_zwolnien_lekarskich jest dokument).

r(lista_wydatkow_karta_firmowa jest dokument).

r(podzadanie jest zadanie).

% Odpowiedzialności

%

r(klient uczestniczy_w spotkanie_info).

r(opiekun_klienta uczestniczy_w spotkanie_info).

r(klient uczestniczy_w spotkanie_meryt).

r(opiekun_klienta uczestniczy_w spotkanie_meryt).

r(sprzedawca uczestniczy_w negocjacje_umowy).

r(odp_za_projekt uczestniczy_w delegacja_zadan).

r(odp_za_projekt zleca bn_1). %bn - blank node

r(bn_1 co podzadanie). % [rp]

r(bn_1 komu pracownik). % [rp]

r(czlonек_zarzadu uczestniczy_w zatwierdzenie_zakupow).

r(czlonек_zarzadu uczestniczy_w zlecenie_zakupow).

% Inne zaleznosci

%

r(projekt_twardy sklada_sie_z wniosek).

r(projekt_twardy sklada_sie_z zalaczniki).

r(projekt_twardy sklada_sie_z biznesplan).

r(projekt_miekkі sklada_sie_z wniosek).

r(projekt_miekkі sklada_sie_z zalaczniki).

r(realizacja_zlecenia zawiera_aktywnosc spotkanie_info).

r(realizacja_zlecenia zawiera_aktywnosc spotkanie_meryt).

r(realizacja_zlecenia zawiera_aktywnosc podpisanie_umowy).

r(realizacja_zlecenia zawiera_aktywnosc wystawienie_faktury).

r(realizacja_zlecenia zawiera_aktywnosc przygotowanie_projektu).

r(realizacja_zlecenia zawiera_aktywnosc przygotowanie_wniosku).

r(realizacja_zlecenia zawiera_aktywnosc akceptacja_klienta).

r(realizacja_zlecenia zawiera_aktywnosc zlozenie_wniosku).

r(realizacja_zlecenia zawiera_aktywnosc poprawki_i_wyjasnienia).

r(spotkanie_meryt zawiera_aktywnosc negocjacje_umowy).

r(spotkanie_meryt zawiera_aktywnosc podpisanie_umowy).

r(przygotowanie_projektu zawiera_aktywnosc pozysk_info_firmy).
r(przygotowanie_projektu zawiera_aktywnosc doprec_informacji).
r(przygotowanie_projektu zawiera_aktywnosc badania_ryнку).
r(przygotowanie_projektu zawiera_aktywnosc komunikacja_z_instytucjami).

r(przygotowanie_wniosku zawiera_aktywnosc delegacja_zadan).

r(zakupy zawiera_aktywnosc zgłaszanie_potrzeb).
r(zakupy zawiera_aktywnosc zatwierdzenie_zakupow).
r(zakupy zawiera_aktywnosc zlecenie_zakupow).

r(obsługa_ksiegowa zawiera_aktywnosc przekaz_dok_ksieg).

r(przekaz_dok_ksieg komu ksiegowosc).

r(przekaz_dok_ksieg czego dok_ksieg).

%dodatkowa zaleznosc: dok papierowa przekazywana jest do tydzien, skany na biezaco.

r(na_biezaco kiedy skan).

r(co_tydzien kiedy papierowa).

r(obsługa_ksiegowa zawiera_aktywnosc uzupełnianie_dok).

r(uzupełnianie_dok dotyczy lista_faktur).

r(uzupełnianie_dok dotyczy lista_urlopow).

r(uzupełnianie_dok dotyczy lista_zwolnien_lekarskich).

r(uzupełnianie_dok dotyczy lista_wydatkow_kartą_firmowa).

5.3. Prototyp taksonomii SimplyUser

Plik `psc2013ontology-simplyuser-extra.pl` (zawierający dodatkowe relacje dotyczące kolejności aktywności w procesach):

%%% Klasy %

% Osoby/Role %

```
c(firma, 'Firma (Simply User)').
```

```
c(caly_zespol_su, 'Cały zespół Simpy User').
```

```
c(zespol_prj, 'Zespół projektowy').
```

```
c(zespol_bad, 'Zespół badawczy').
```

```
c(zespol_dev, 'Zespół developerski').
```

```
c(zespol_sprzedaz, 'Zespół ds. sprzedaży').
```

```
c(klient, 'Klient').
```

```
r(zespol_prj jest_czescia caly_zespol_su).
```

```
r(zespol_bad jest_czescia caly_zespol_su).
```

```
r(zespol_dev jest_czescia caly_zespol_su).
```

```
r(zespol_sprzedaz jest_czescia caly_zespol_su).
```

```
a(zespol_prj, liczba_osob, [2]).
```

```
a(zespol_bad, liczba_osob, [1]).
```

```
a(zespol_dev, liczba_osob, [2]).
```

```
a(zespol_sprzedaz, liczba_osob, [1]).
```

%% Artefakty %

```
c(projekt, 'Projekt').
```

```
r(maly_projekt jest projekt).
r(duzy_projekt jest projekt).
r(bduzy_projekt jest projekt).
a(projekt, czas_trwania, [dwa_tyg, cztery_tyg, miesiace]).
a(maly_projekt, czas_trwania, [dwa_tyg]).
a(duzy_projekt, czas_trwania, [cztery_tyg]).
a(bduzy_projekt, czas_trwania, [miesiace]).
```

```
c(umowa_badania, 'Umowa z osobą do badań').
c(raport_badania, 'Raport z badań dla klienta').
c(umowa_klient, 'Umowa z klientem').
```

```
%Typowe dla firmy pojęcia
c(warsztaty, 'Warsztaty').
c(makieta, 'Makieta').
c(persona, 'Persona').
```

```
%% Procesy %
c(realizacja_projektu, 'Realizacja projektu').
    r(realizacja_projektu jest proces).
c(realizacja_badan, 'Realizacja badań').
    r(realizacja_badan jest podproces).
    r(podproces jest proces).
c(sprzedaz, 'Sprzedaż').
    r(sprzedaz jest proces).
c(wdrozenie, 'Wdrożenie').
    r(wdrozenie jest proces).
```

```
%Kolejność procesów
r(sprzedaz poprzedza realizacja_projektu).
r(realizacja_projektu poprzedza wdrozenie).

r(sprzedaz obejmuje rozmowa_tel).
    r(klient uczestniczy_w rozmowa_tel).
    r(firma uczestniczy_w rozmowa_tel).
    r(rozmowa_tel poprzedza przyg_oferty_wstepnej).
r(sprzedaz obejmuje przyg_oferty_wstepnej).
    r(przyg_oferty_wstepnej poprzedza spotkanie).
r(sprzedaz obejmuje spotkanie).
    r(spotkanie poprzedza decyzja_o_wspolpracy).
r(sprzedaz obejmuje decyzja_o_wspolpracy).
    r(decyzja_o_wspolpracy poprzedza warsztaty).
r(sprzedaz obejmuje warsztaty).
    r(warsztaty efektem_jest notatka).
    r(warsztaty poprzedza opracowanie_oferty).
r(sprzedaz obejmuje opracowanie_oferty).
    r(opracowanie_oferty poprzedza umowa_klient).
r(sprzedaz obejmuje umowa_klient).
    r(klient uczestniczy_w umowa_klient).
    r(firma uczestniczy_w umowa_klient).
    r(umowa_klient zawiera nda). % opcjonalnie
    r(umowa_klient poprzedza archiwizacja_dok).

r(realizacja_projektu obejmuje warsztaty).
```

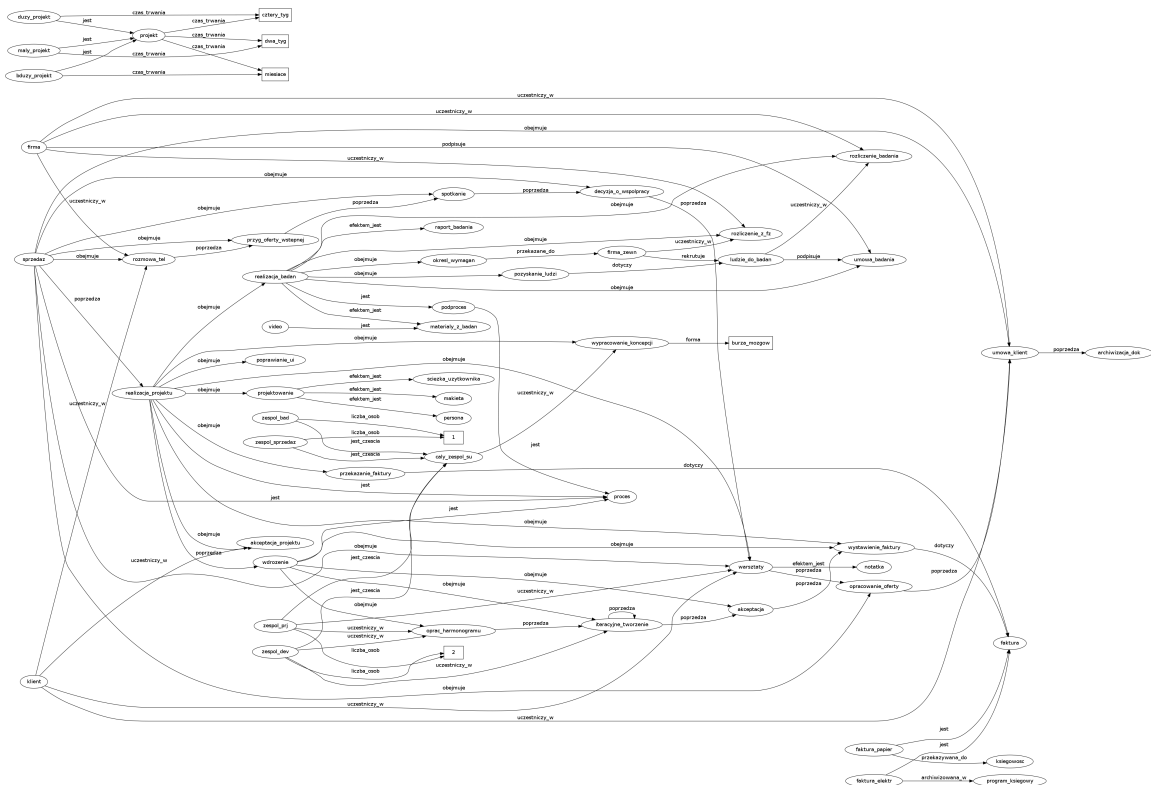
```
r(klient uczestniczy_w warsztaty).
r(zespol_prj uczestniczy_w warsztaty).
r(realizacja_projektu obejmuje wypracowanie_koncepcji).
r(caly_zespol_su uczestniczy_w wypracowanie_koncepcji).
a(wypracowanie_koncepcji, forma, [burza_mozgow]).
r(realizacja_projektu obejmuje projektowanie).
r(projektowanie efektem_jest persona).
r(projektowanie efektem_jest sciezka_uzytkownika).
r(projektowanie efektem_jest makieta).
r(realizacja_projektu obejmuje realizacja_badan).
r(realizacja_badan efektem_jest raport_badania).
r(realizacja_badan efektem_jest materialy_z_badan).
r(video jest materialy_z_badan).
r(realizacja_projektu obejmuje poprawianie_ui).
r(realizacja_projektu obejmuje akceptacja_projektu).
r(klient uczestniczy_w akceptacja_projektu).
r(realizacja_projektu obejmuje wystawienie_faktury).
r(wystawienie_faktury dotyczy faktura).
r(realizacja_projektu obejmuje przekazanie_faktury).
r(przekazanie_faktury dotyczy faktura).
r(faktura_papier jest faktura).
r(faktura_elektr jest faktura).
r(faktura_papier przekazywana_do ksiegowosc).
r(faktura_elektr archiwizowana_w program_ksiegowy).

r(realizacja_badan obejmuje okresl_wymagan).
r(okresl_wymagan przekazane_do firma_zewn).
```



```
r(realizacja_badan obejmuje pozyskanie_ludzi).
  r(pozyskanie_ludzi dotyczy ludzie_do_badan).
  r(firma_zewn rekrutuje ludzie_do_badan).
r(realizacja_badan obejmuje rozliczenie_z_fz).
  r(firma uczestniczy_w rozliczenie_z_fz).
  r(firma_zewn uczestniczy_w rozliczenie_z_fz).
r(realizacja_badan obejmuje umowa_badania).
  r(ludzie_do_badan podpisuje umowa_badania).
  r(firma podpisuje umowa_badania).
r(realizacja_badan obejmuje rozliczenie_badania). % opcjonalnie
  r(firma uczestniczy_w rozliczenie_badania).
  r(ludzie_do_badan uczestniczy_w rozliczenie_badania).

r(wdrozenie obejmuje oprac_harmonogramu).
  r(zespol_prj uczestniczy_w oprac_harmonogramu).
  r(zespol_dev uczestniczy_w oprac_harmonogramu).
  r(oprac_harmonogramu poprzedza iteracyjne_tworzenie).
r(wdrozenie obejmuje iteracyjne_tworzenie).
  r(zespol_dev uczestniczy_w iteracyjne_tworzenie).
  r(iteracyjne_tworzenie poprzedza iteracyjne_tworzenie).
  r(iteracyjne_tworzenie poprzedza akceptacja).
r(wdrozenie obejmuje akceptacja).
  r(akceptacja poprzedza wystawienie_faktury).
r(wdrozenie obejmuje wystawienie_faktury).
```

Rysunek 13: Wizualizacja rozszerzonego wariantu taksonomii firmy SimpleUser.

Literatura

- [1] Mike Bergman. A brief survey of ontology development methodologies. <http://www.mkbergman.com/906/a-brief-survey-of-ontology-development-methodologies/>, August 30, 2010. Accessed: 2013.
- [2] Dean Jones, Trevor Bench-capon, and Pepijn Visser. Methodologies for ontology development. pages 62–75, 1998.
- [3] Mike Uschold, Michael Gruninger, Mike Uschold, and Michael Gruninger. Ontologies: Principles, methods and applications. *Knowledge Engineering Review*, 11:93–136, 1996.
- [4] Dave Reynolds, Epimorphics Ltd., W3C, An organization ontology, <http://www.w3.org/TR/vocab-org/>
- [5] Jelena Jovanovic, Melody Siadaty, IntelLEO Organization Ontology, <http://intelleo.eu/ontologies/organization/spec/>
- [6] The London Gazette Ontology, Organisation Module, <http://www.london-gazette.co.uk/ontology/organisation.owl>
- [7] The London Gazette ontology, <http://www.london-gazette.co.uk/ontology/gazette.owl>
- [8] The London Gazette, Re-using Gazette data, <http://www.london-gazette.co.uk/reuse>
- [9] PROTON Ontology (PROTo ONtology), <http://proton.semanticweb.org/>
- [10] Friend of a Friend Ontology, <http://www.foaf-project.org/>
- [11] Dublin Core Ontology, <http://dublincore.org/>

- [12] Andreas L. Opdahl, Giuseppe Berio, Mounira Harzallah, Raimundas Matulevicius, An ontology for enterprise and information systems modelling. *Applied Ontology* 7(1): 49-92 (2012).
- [13] Diego Magro, Anna Goy, A core reference ontology for the customer relationship domain. *Applied Ontology* 7(1): 1–48 (2012).
- [14] Juha Tiihonen, Mikko Heiskala, Andreas Anderson, Timo Soininen, WeCoTin - A practical logic-based sales configurator. *AI Commun.* 26(1): 99–131 (2013).
- [15] DIP Data, Information and Process Integration with Semantic Web Services project FP6 – 507483, WP3: Service Ontologies and Service Description, D3.3 A Business Data Ontology.
- [16] Oberle, D., Lamparter, S., Grimm, S., Vrandecic, D., Staab, S. Gangemi, Towards ontologies for formalizing modularization and communication in large software systems. *Applied Ontology*, 1(2), 163—202 (2006).
- [17] Anna Goy, Diego Magro, User-Friendly Interaction in an On-line System Based on Semantic Technologies. *WEBIST (Selected Papers) 2011*: 163–176
- [18] Yalan Yan, Xianjin Zha, Applying OWL to Build Ontology for Customer Knowledge Management. *JCP* 5(11): 1693–1699 (2010).
- [19] Dolce Foundational Ontology, <http://www.loa.istc.cnr.it/DOLCE.html>
- [20] Jan Recker, Marta Indulska, Michael Rosemann and Peter Green, The ontological deficiencies of process modeling in practice. *European Journal of Information Systems*, 19 5: 501–525 (2010).
- [21] Jan Recker, Michael Rosemann, Peter Green and Marta Indulska, Do ontological deficiencies in modeling grammars matter?. *MIS Quarterly*, 35 1: 57–79 (2011).

