

AKADEMIA GÓRNICZO-HUTNICZA

Wydział Elektrotechniki, Automatyki, Informatyki i Elektroniki



KATEDRA INFORMATYKI

Integracja usługi google translator z systemem CMS Nuxeo

Wersja **0.3** z dnia **30.11.2009**

Kierunek, rok studiów:

Informatyka, IV

Przedmiot:

IOSR

Prowadzący zajęcia:

Dr inż. Dominik Radziszowski

Rok akad:

2009/2010

Zespół autorski:

Maria Szymczak

szymczak@student.agh.edu.pl

Bartłomiej Czopyk

czopyk@student.agh.edu.pl

Tomasz Handzlik

thandzli@student.agh.edu.pl

Tomasz Lewicki

tlewicki@student.agh.edu.pl

Kraków, październik 2009

Niniejsze opracowanie powstało w trakcie i jako rezultat zajęć dydaktycznych z przedmiotu wymienionego na stronie tytułowej, prowadzonych w Akademii Górniczo-Hutniczej w Krakowie (AGH) przez osobę (osoby) wymienioną (wymienione) po słowach "Prowadzący zajęcia" i nie może być wykorzystywane w jakikolwiek sposób i do jakichkolwiek celów, w całości lub części, w szczególności publikowane w jakikolwiek sposób i w jakiegokolwiek formie, bez uzyskania uprzedniej, pisemnej zgody tej osoby (tych osób) lub odpowiednich władz AGH.

Copyright © iosr Akademia Górniczo-Hutnicza (AGH) w Krakowie

Spis treści

1. Sformułowanie zadania projektowego	4
1.1. Obszar i przedmiot projektowania.....	4
1.1.1. Opis dziedziny problemu.....	4
1.1.2. Zakres systemu.....	4
1.1.2.1. Tłumaczenie asynchroniczne.....	4
1.1.2.2. Tłumaczenie synchroniczne.....	4
1.1.2.3. Tłumaczenie do wielu języków naraz.....	4
1.1.2.4. Obsługę różnych silników tłumaczenia.....	4
1.1.2.5. Obsługę plików w różnych formatach.....	4
1.1.3. Kontekst systemu.....	5
1.1.3.1. Silnik tłumaczenia (domyślnie: google translator).....	5
1.1.3.2. CMS Nuxeo.....	5
1.1.3.3. Kodery / dekodery tekstu do tłumaczenia – standard Xliff.....	5
2. Opis wymagań przyszłych użytkowników	7
2.1. Wymagania funkcjonalne	7
2.2. Wymagania нефункционалне	9
3. Interfejsy	11
3.1. Webservice'y	11
4. Diagramy	12
4.1. Architektury	12
4.2. Komponentów.....	13
4.3. Sekwencji	14
5. Narzędzia deweloperskie	15
5.1. Koordynacja pracy w grupie:.....	15
5.1.1. Projekt:.....	15
5.1.2. Podział zadań:.....	15
5.1.3. Svn: 15	
5.1.4. Wiki: 15	
5.1.5. Raporty z Mavena:.....	15

6. Webservice'y	16
6.1. Opis	16
6.2. Objasnienie interfejsów.....	16
6.2.1. <i>Interfejs odpowiedzialny za tłumaczenie</i>	16
6.2.2. <i>Interfejs odpowiedzialny za odbiór rezultatów tłumaczenia</i>	17
6.3. Wykorzystywane typy danych.....	17
6.4. Dostęp do webservice'u tłumaczącego.....	17
7. Roboczy słownik pojęć	18

1. Sformułowanie zadania projektowego

1.1. Obszar i przedmiot projektowania

1.1.1. Opis dziedziny problemu

Zadaniem projektowym jest integracja usługi google translator z systemem CMS Nuxeo. Do zadań docelowego systemu należy tłumaczenie jednego lub więcej plików zgodnie z wybranymi ustawieniami użytkownika. Możliwa jest obsługa więcej niż jednego języka naraz oraz plików o różnych formatach. Planowana jest niezależność systemu od silnika tłumaczenia.

1.1.2. Zakres systemu

System będzie warstwą pośredniczącą między CMS Nuxeo a danym silnikiem tłumaczenia. Zakres systemu obejmuje:

1.1.2.1. Tłumaczenie asynchroniczne

opcja pozwalająca na asynchroniczne wysłanie danego pliku (lub plików) do tłumaczenia i oznaczenia jego statusu jako pliku w trakcie tłumaczenia. Rezultat operacji jest zapisywany jako wskazany wcześniej, osobny plik. Obsługa tego rodzaju tłumaczenia jest wymagana wobec wszystkich wspieranych silników.

1.1.2.2. Tłumaczenie synchroniczne

opcja pozwala na szybkie wygenerowanie podglądu danego pliku w wybranym języku. Ten rodzaj tłumaczenia nie jest obsługiwany przez wszystkie silniki.

1.1.2.3. Tłumaczenie do wielu języków naraz

użytkownik może zażądać tłumaczeń pliku w kilku językach. Opcja wspierana tylko w czasie tłumaczenia asynchronicznego. Tłumaczenie w każdym z języków generuje osobny plik wynikowy.

1.1.2.4. Obsługę różnych silników tłumaczenia

system docelowo jest niezależny od silnika (domyślny to google translator). Administrator może dodać dowolnie wiele silników, a użytkownik do tłumaczenia wybrać dowolny z nich.

1.1.2.5. Obsługę plików w różnych formatach

tłumaczenie obejmuje pliki w różnych formatach, nie tylko tekstowym. Obsługiwane są m.in. pliki .pdf i .doc.

1.1.3. Kontekst systemu

1.1.3.1. Silnik tłumaczenia (domyślnie: google translator)

wymagana obsługa co najmniej 2 języków i akceptowalny czas zwrócenia rezultatu;

1.1.3.2. CMS Nuxeo

1.1.3.2.1. WSTĘP:

Nuxeo jest open-sourceową platformą do rozwijania systemów klasy CMS opartą o J2EE. Zapewnia szkielet systemu zarządzania treścią, definiuje podstawowe widoki i funkcjonalności CMS (pliki, fora), zapewnia kontrolę dostępu (autentykacja, autoryzacja), definiuje model danych i metadanych, implementuje środowisko do wdrażania dedykowanych rozszerzeń platformy (extension points), komponenty wspomagające integrację (serwisy wewnętrzne), silnik wyszukiwania, obsługę zdarzeń, zarządzanie cyklem życia i wiele innych.

1.1.3.2.2. ARCHITEKTURA:

Nuxeo opiera się na architekturze komponentowej, skierowanej na standardy (modele programowania: J2EE, OSGi; usługi sieciowe: SOAP; definiowanie treści: XML Schemas). Z racji pewnych wad architektury J2EE (brak mechanizmów rozszerzania modelu komponentów, mechanizmów deklarowania zależności), Nuxeo definiuje warstwę środowiska uruchomieniowego (runtime layer), której zadaniem jest wyeliminowanie tych problemów.

Warstwa ta między innymi definiuje mechanizm extension points, najciekawsze z punktu widzenia developerów. Komponenty podlegające temu mechanizmowi, mogą być aktywowane, bądź dezaktywowane przez zewnętrzną konfigurację. Poprzez dodawane zasoby, podobnie do pluginów, można modyfikować wygląd oraz zachowanie serwisu. Zasobem takim zazwyczaj jest XML, może być ponadto kod javy (POJO, Seam), xhtml (JSF), jpg, bundle zasobów, itd. Dodawane zasoby, zwane kontrybucjami, mogą nadpisywać konfigurację komponentów wystawiających extension points.

Na warstwie środowiska uruchomieniowego bazuje Nuxeo Core, udostępniające serwisy wewnętrzne platformy, min: usługi repozytorium, wersjonowania, bezpieczeństwa, zarządzania cyklem życia, obsługi zdarzeń. Ostatnią warstwą jest warstwa prezentacji, podobno w funkcjonalności do odpowiedniej warstwy klasycznych systemów klasy Enterprise. Wyświetla ona zawartość treści, możliwe opcje oraz definiuje nawigację między widokami.

1.1.3.2.3. TECHNOLOGIE PLATFORMY:

Nuxeo domyślnie dostarczane jest z serwerem aplikacji JBoss. Implementacja opiera się o framework Seam z rozbudowaną warstwą prezentacji JSF (własne biblioteki tagów) wzbogaconą przez Facelets oraz (od najnowszej wersji 5.2) RichFaces. Ponadto wykorzystywany jest ajax4jsf. Nuxeo domyślnie korzysta z implementacji warstwy persystencji opartej o bazę danych Hipersonic, dostarczaną wraz z serwerem Jboss.

1.1.3.3. Kodery / dekodery tekstu do tłumaczenia – standard Xliff

1.1.3.3.1. IDEA:

Xliff - XML Localization Interchange File Format to specjalny format plików oparty na XML. Został stworzony, aby ułatwić ekstrakcję tekstu, który ma zostać przetłumaczony. Translator może skupić się na samym tekście, nie biorąc pod uwagę jego formatowania. Co najważniejsze, użycie takiego standardu stwarza uniwersalne narzędzie do obsługi różnych formatów plików. Używamy biblioteki file2xliff4j, której główną zaletą jest szeroka gama obsługiwanych rodzajów plików (xml, html, txt, pliki open office'owe oraz microsoftowe).

1.1.3.3.2. TŁUMACZENIE:

Proces tłumaczenia za pomocą formatu xiff wygląda następująco:

1. Ekstrakcja tekstu do tłumaczenia z oryginalnego pliku;
2. Konwersja pliku do formatu xiff;
3. Wysłanie pliku xiff do tłumaczenia;
4. Pobranie przetłumaczonego tekstu ze zwrotnego pliku i zmapowanie go w odpowiednie miejsce do oryginalnego dokumentu;

Oryginalny plik, oprócz tekstu, zawiera także znaczniki dotyczące formatowania tekstu (np. w pliku html). Wszystkie te nietłumaczone detale dotyczące formatowania i wyglądu tekstu zapisane zostaną w specjalnym pliku 'skeleton', gdzie tłumaczone zdania zostają zastąpione specjalnymi znacznikami: '%%n%%' (gdzie 'n' to kolejny numer fragmentu tekstu - będzie to ważne przy mapowaniu przetłumaczonego tekstu z powrotem do pliku). Każdy fragment tłumaczonego tekstu natomiast znajdzie się w pliku xiff pomiędzy tagami 'trans-unit'. Tagi te posiadają również atrybuty id (oznaczające odpowiednie miejsce do późniejszego wstawienia w pliku skeleton) oraz zagnieżdżone tagi 'source' z atrybutem 'xml:lang' (oznaczającymi język, w którym napisany jest tekst). Cały mechanizm nie jest skomplikowany, problemem jest tylko implementacja odpowiednich filtrów, co w przypadku plików xml czy html jest dość proste, może natomiast rodzić problemy przy gorzej udokumentowanych formatach (np. windowsowych).

W trakcie tłumaczenia w tagach 'trans-unit' zostają zagnieżdżone tagi 'alt-trans' zawierające przetłumaczone odpowiednie kawałki tekstu. Ten właśnie przetłumaczony tekst zostanie następnie zmapowany z plikiem skeleton, dając w rezultacie poprawnie sformatowany plik.

2. Opis wymagań przyszłych użytkowników

2.1. Wymagania funkcjonalne

2.1.1 Asynchroniczne tłumaczenie z zapisem do wyznaczonego pliku

Użytkownik zaznacza określony dokument z folderu, a następnie za pomocą menu kontekstowego lub przycisku umieszczonego pod widokiem folderu – wybiera opcję *‘Translate’*, na ekranie pojawia się okno tłumaczenia. Użytkownik wybiera z listy pożądaną język docelowy, może również określić język źródłowy. Wymagane jest także podanie docelowej lokalizacji i nazwy przetłumaczonego dokumentu. Po zatwierdzeniu – dokument jest tłumaczony, a gdy proces zostanie zakończony - przetłumaczony plik zostaje zapisany we wskazanej lokalizacji.

2.1.2 Tłumaczenie wielu plików naraz

Użytkownik podczas procedury opisanej w wymaganiu 1.1, może zaznaczyć wiele dokumentów do przetłumaczenia. W tym przypadku jednak nazwy plików z tłumaczeniem zostaną ustalone automatycznie na podstawie nazwy plików wejściowych.

2.1.3 Tłumaczenia na wiele języków naraz

W oknie tłumaczenia, użytkownik może zaznaczyć wiele języków docelowych. W ten sposób uzyskamy tłumaczenie wybranych dokumentów na każdy z zaznaczonych języków. Przetłumaczone pliki automatycznie otrzymają w nazwie przyrostek będący skrótem od języka na jaki został przetłumaczony ten plik.

2.1.4 Synchroniczne tłumaczenie pliku (funkcja ‘podgląd’)

Po naciśnięciu na wybrany dokument, użytkownik wybiera z w panelu zarządzania dokumentem zakładkę ‘Podgląd’. Z lewej strony ekranu pojawia się przycisk ‘Przetłumacz’ oraz listy z których można wybrać język źródłowy i docelowy – naciśnięcie przycisku spowoduje, iż obok oryginalnego tekstu pojawia się przetłumaczona treść. W przypadku gdy niedostępny jest podgląd pliku, niedostępna jest także możliwość jego synchronicznego tłumaczenia.

2.1.5 Przetłumaczone pliki posiadają odpowiednie własności ustalone na podstawie pliku oryginalnego i odpowiednich reguł

W systemie Nuxeo, każdy plik może być określony przez wiele istotnych własności– stąd przetłumaczone pliki otrzymają odpowiednie własności ustalone na podstawie następujących zasad:

- Własność „Created at” oraz “Last modified At” otrzymają datę ukończenia tłumaczenia pliku
- Własność „Language” otrzyma wartość stosowną dla języka tłumaczenia pliku
- Własność „Title” będzie taka jak nazwa przetłumaczonego pliku
- Reszta własności otrzyma taką samą wartość jak w pliku oryginalnym

2.1.6 Przetłumaczone pliki posiadają relacje typu „tłumaczenie” wiążącą ją z oryginalnym dokumentem.

Taka relacja tworzona jest automatycznie w momencie przetłumaczenia dokumentu i opiera się ona na standardowym mechanizmie powiązań plików oferowanym przez system Nuxeo i możemy ją zobaczyć w zakładce „Relations” w panelu zarządzania dokumentem.

2.1.7 Zapamiętywanie i ułatwiony dostęp do ulubionych języków użytkownika

Użytkownik w ustawieniach osobistych (panel „Personal Workspace”) może wybrać zakładkę konfiguracji tłumaczeń. W zakładce tej można wybrać dla danego silnika translacji maksymalnie sześć ulubionych języków tłumaczeń. Dzięki temu, w menu podręcznym dotyczącym tłumaczenia dokumentu, będziemy mogli wybrać język docelowy spośród ulubionych.

2.1.8 Widok przedstawiający status tłumaczonych dokumentów użytkownika

Użytkownik może zobaczyć listę plików zleconych do tłumaczenia wraz z aktualnym statusem tłumaczenia – jest ona dostępna w panelu dashboard użytkownika.

2.1.9 Dodawanie silnika translacji przez administratora

Użytkownik o prawach administratora, może w zakładce konfiguracji tłumaczenia, definiować nowe silniki translacji. Administrator musi wpisać nazwę silnika translacji, opcjonalny opis oraz wskazać plik ze stosownym pluginem dla obsługi tego silnika. Na podstawie tego pliku – system sprawdza czy dostępne są wersje synchroniczne i asynchroniczne tłumacza oraz inne funkcjonalności silnika takie jak: detekcja języka źródłowego, określenie jakości tłumaczenia, określenie ostatecznego terminu ukończenia tłumaczenia. Usługa tłumacza nie musi być dostępna w momencie definicji. Dla dodanego silnika można również zdefiniować timeout.

2.1.10 Edycja silników translacji przez administratora

Użytkownik o prawach administratora, może w zakładce konfiguracji tłumaczenia, przeglądać listę dostępnych silników translacji oraz edytować bądź usunąć wybrany silnik. W opcjach edycji może zmienić nazwę, opis, plik z pluginem oraz czas timeout.

2.1.11 Wybór silnika translacji poprzez użytkownika

Użytkownik może w zakładce konfiguracji tłumaczeń wybrać jeden spośród dostępnych silników translacji. Wyświetlone zostaną podstawowe informacje o tym silniku takie jak: nazwa, opis, czas timeout, deklarowana przez silnik dostępność wersji synchronicznej i asynchronicznej, a także możliwość detekcji języka źródłowego.

2.1.12 Detekcja języka źródłowego

Wykorzystanie funkcji detekcji języka źródłowego w przypadku google translator lub innych silników translacji rozpoznających języki. Użytkownik podczas tłumaczenia nie musi podawać języka źródłowego – wówczas silnik sam rozpozna język źródłowy i przed rozpoczęciem tłumaczenia, użytkownik zostanie zapytany o to czy język został prawidłowo rozpoznany poprzez pojawienie się odpowiedniego okna. Użytkownik może zatwierdzić bądź zmienić język źródłowy. Użytkownik może też wybrać język źródłowy ale zostać powiadomiony o rozpoznaniu przez silniki innego języka – w tym przypadku również zostanie wyświetlony stosowny komunikat.

2.1.13 Notyfikacje o ukończeniu tłumaczenia oraz ewentualnie występujących problemach

Użytkownik może skorzystać z mechanizmu notyfikacji mailowych Nuxeo aby zostać powiadomiony o zakończeniu tłumaczenia dokumentu lub ewentualnie występujących problemach.

2.1.14 Tłumaczenie różnych formatów dokumentów

Użytkownik może wybrać do tłumaczenia pliki o różnych formatach – w przypadku gdy zasotanie zaznaczony plik o nieobsługiwanym formacie, w oknie tłumaczenia pojawi się stosowna informacja na ten temat i taki plik zostanie zignorowany.

2.1.15 Wybór pożądanego jakości tłumaczenia

Użytkownik w momencie zlecenia translacji dokumentu, w oknie tłumaczenia ma możliwość wyboru jakości tłumaczenia. Obecność tej opcji zależy od możliwości silnika translacji.

2.1.16 Określenie ostatecznego terminu ukończenia tłumaczenia

W oknie tłumaczenia istnieje możliwość wybrania daty, która określa maksymalny termin ukończenia tłumaczenia. Obecność tej opcji zależy od silnika translacji.

2.1.17 Ustawienie statusu „W tłumaczeniu” dla tłumaczonych plików oraz właściwości określającej postęp w procesie tłumaczenia.

Status „W tłumaczeniu” jest częścią mechanizmu „workflow” systemu Nuxeo i ustawiony jest automatycznie.

2.1.18 Sygnalizacja niedostępności usług zewnętrznych oraz innych błędów

Użytkownik będzie informowany o dostępności usług tłumaczenia na wiele sposobów i na różnych etapach:

2.1.18.1 Przed zleceniem zadania tłumaczenia (zarówno w przypadku tłumaczenia synchronicznego i asynchronicznego)

odpowiednie przyciski bądź też pola menu kontekstowego - w przypadku niedostępności usługi - będą oznaczone jako nieaktywne poprzez ich wyszarzenie.

2.1.18.2 Dla tłumaczenia asynchronicznego – sygnalizacja problemów z wysyłaniem treści do tłumaczenia

wyświetlane będzie okno ze stosownym komunikatem opisującym problem. Użytkownik ma możliwość określenia czasu timeout dla danego silnika translacji w zakładce konfiguracji tłumaczenia.

2.2. Wymagania niefunkcjonalne

2.2.1. Intuicyjny interfejs

Bardzo krótki czas uczenia się obsługi – poniżej godziny oraz dostępne strony pomocy;

2.2.2. Odporność na błędy użytkownika

Wykorzystanie możliwości detekcji języka jeśli oferuje ją silnik translacji. Komunikaty ostrzegające o zaznaczeniu plików o nieobsługiwanych przez plugin formatach.

2.2.3. Konfigurowalność, niezależność od silnika tłumacza.

Istnieje możliwość korzystania z wielu silników translacji, a użytkownik nie jest świadom realizacji szczegółów komunikacji z tym silnikiem. Różnice między silnikami translacji objawiać się mogą jedynie w oferowanej funkcjonalności.

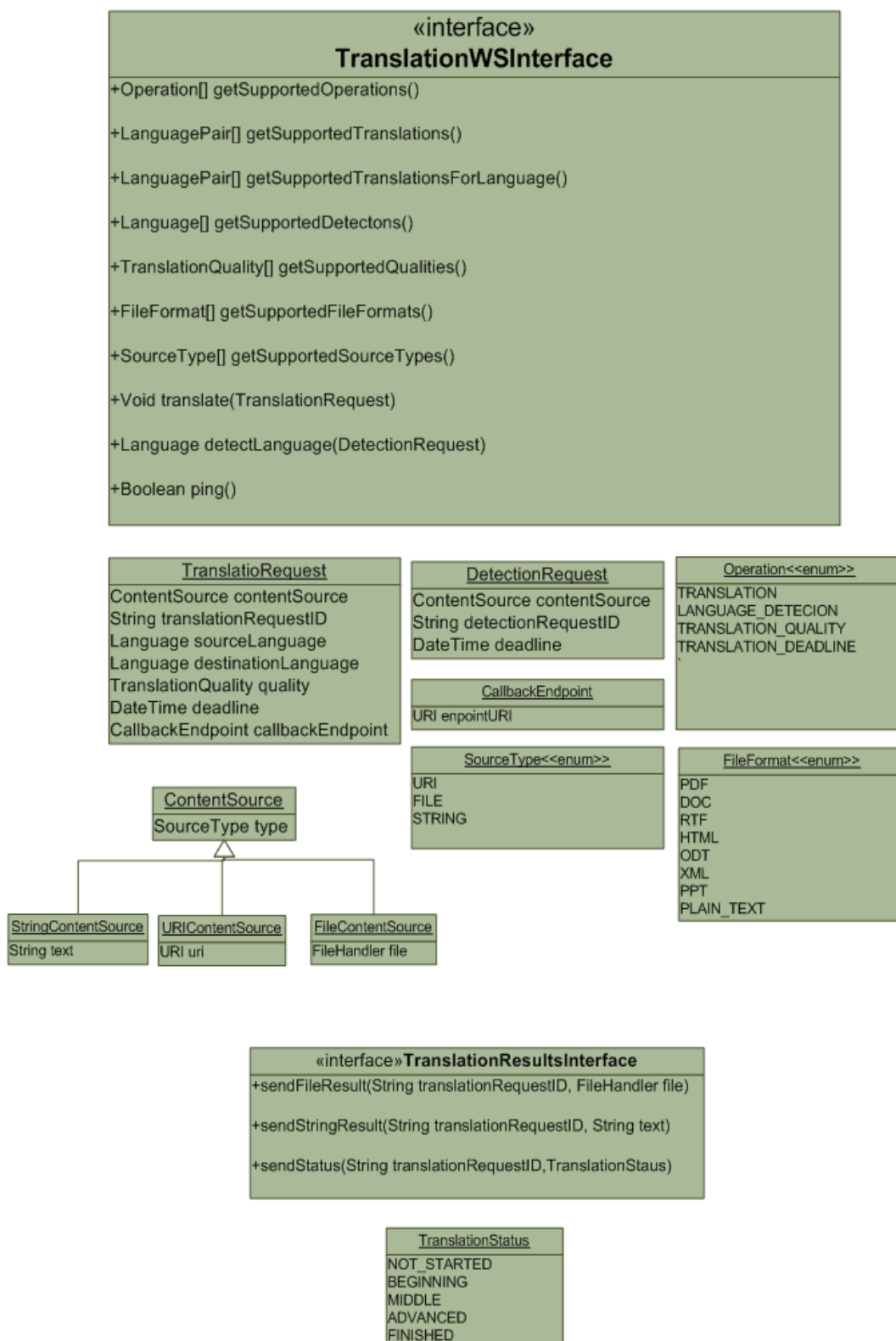
2.2.4. Niezawodność i prawidłowa integracja z systemem Nuxeo

Plugin tłumaczący prawidłowo integruje się z systemem Nuxeo nie zmniejszając jego niezawodności i nie wpływa na prawidłowość działania pozostałych funkcji systemu.

2.2.5. Plugin prawidłowo funkcjonuje na konfiguracji sprzętowej wymaganej dla systemu Nuxeo, nie stawiając dodatkowych wymagań.

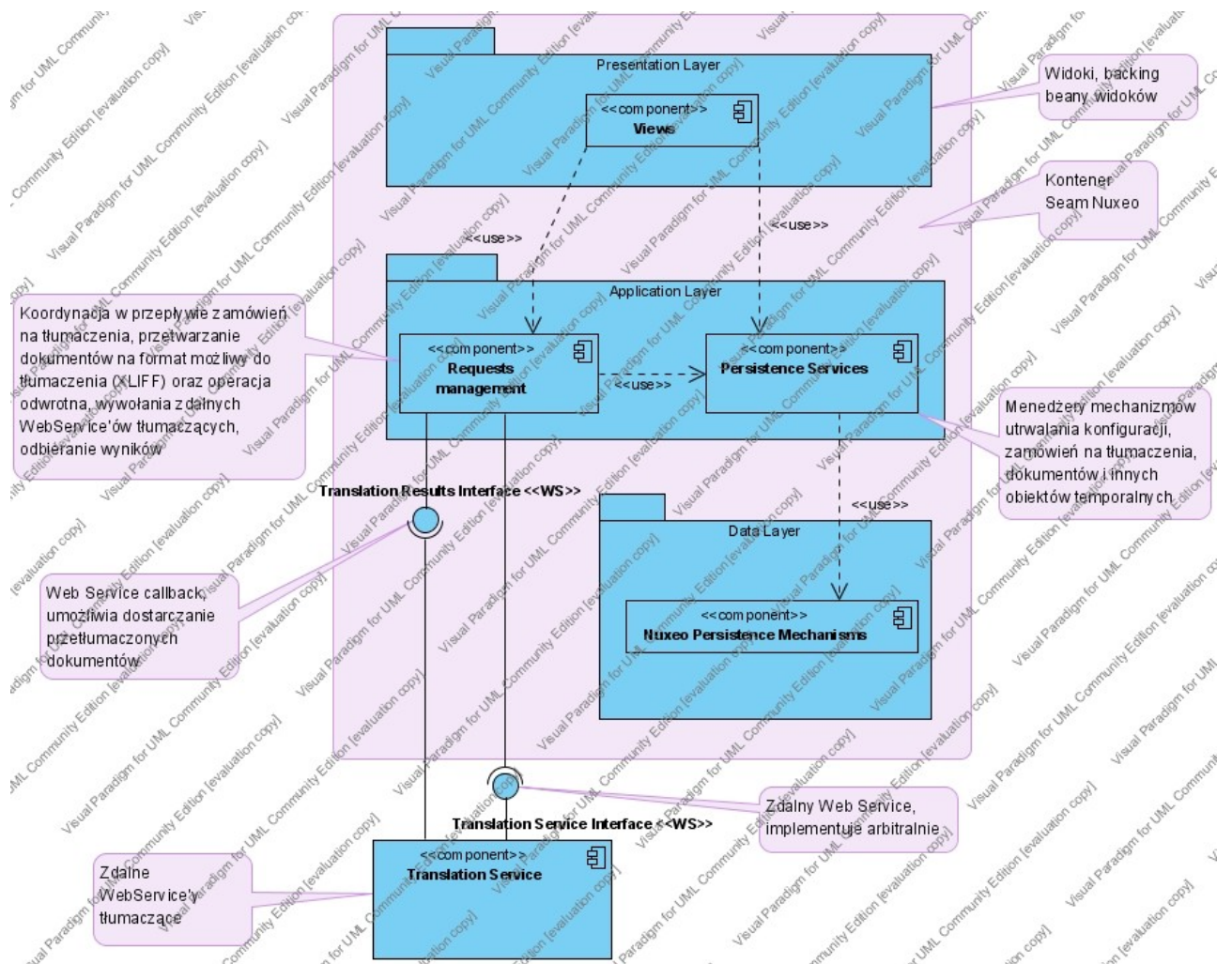
3. Interfejsy

3.1. Webservice'y

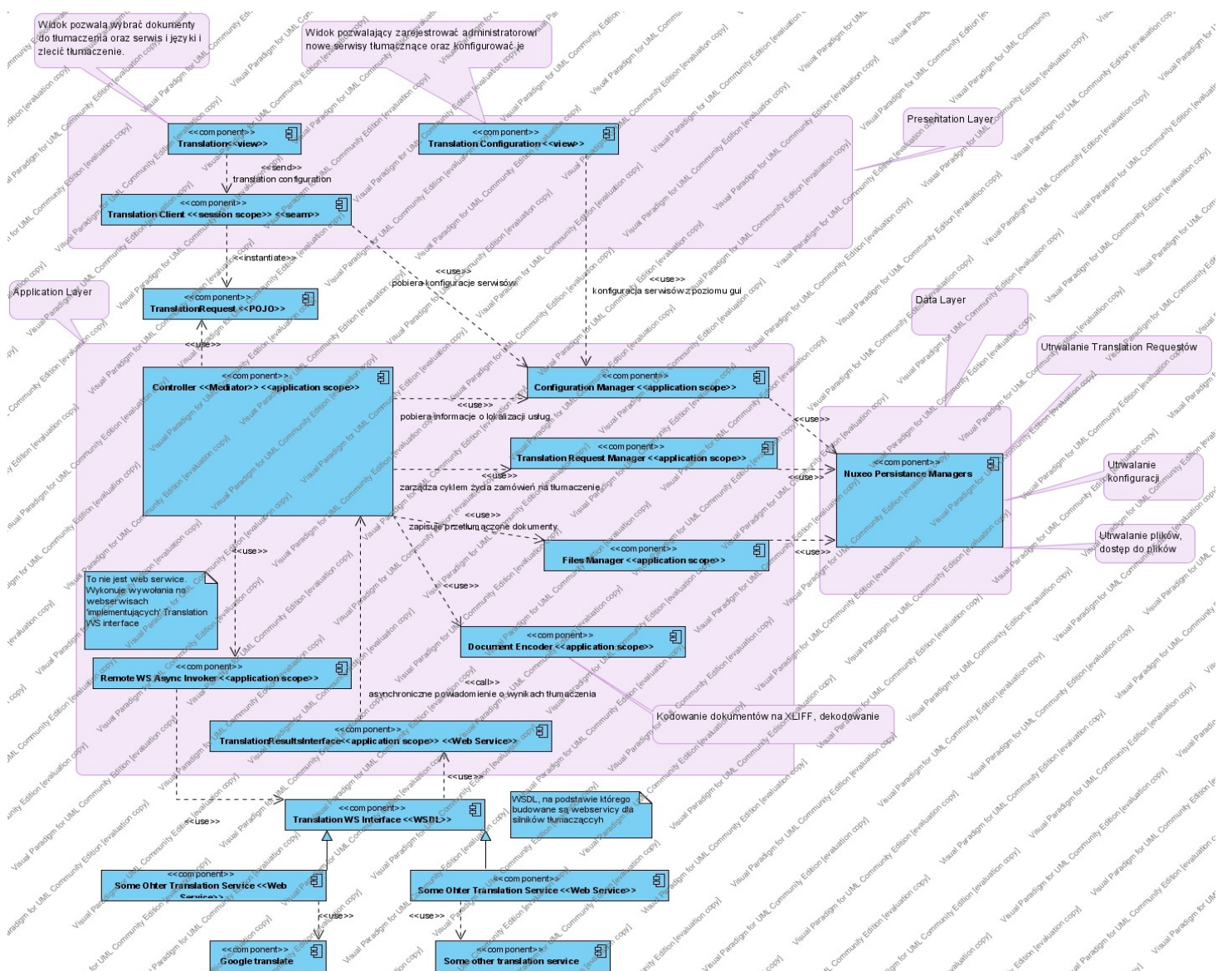


4. Diagramy

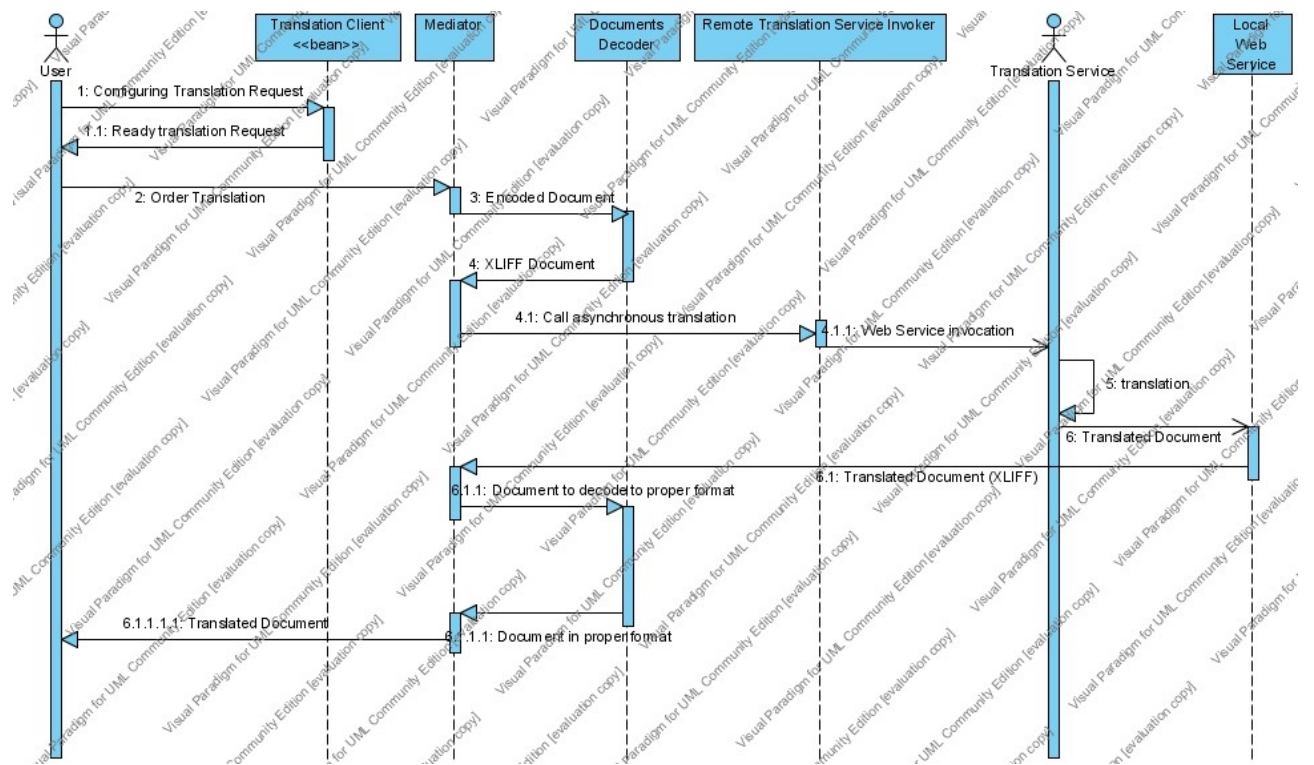
4.1. Architektury



4.2. Komponentów



4.3. Sekwencji



5. Narzędzia deweloperskie

5.1. Koordynacja pracy w grupie:

5.1.1. Projekt:

<http://code.google.com/p/iosr-nuxeo/>

5.1.2. Podział zadań:

<http://code.google.com/p/iosr-nuxeo/issues/list>

5.1.3. Svn:

<https://iosr-nuxeo.googlecode.com/svn>

5.1.4. Wiki:

<http://code.google.com/p/iosr-nuxeo/w/list>

5.1.5. Raporty z Mavena:

6. Webservice'y

6.1. Opis

Warstwę aplikacji odpowiedzialną za wysyłanie oraz odbieranie dokumentów do tłumaczeniu zdecydowaliśmy się zbudować w oparciu o webserwisy. Rozwiązanie takie zostało wybrane ze względu na konieczność zapewnienia asynchroniczności wywołań. Wśród możliwych technologii webserwisy wydają się być rozwiązaniem najlepszym, m.in. ze względu na bardzo dobre działanie niezależnie od swojej lokalizacji i szczegółów implementacji.

W porównaniu z np. CORBA czy RMI webserwisy cechują się gorszą wydajnością, jednak w przypadku realizowanej przez nas aplikacji, nie jest to istotna wada. Dużą zaletą jest natomiast przejrzystość i prostota procesu tworzenia webserwisu.

W celu zapewnienia asynchroniczności w aplikacji działają dwa rodzaje webserwisów. Do pierwszego rodzaju należą webserwisy implementujące interfejs służący do przesyłania pliku do przetłumaczenia. Każdy dostawca usługi tłumaczenia będzie posiadać własną implementację.

Drugi rodzaj (istnieje tylko jedna implementacja wewnątrz aplikacji) służy do odbierania rezultatu tłumaczenia i jest on implementowany przez nas.

6.2. Objasnienie interfejsów

6.2.1. Interfejs odpowiedzialny za tłumaczenie

Interfejs ten jest zdefiniowany za pomocą pliku *wsdl*. Jest on bardzo obszerny, gdyż został zaprojektowany tak, aby był jak najbardziej uniwersalny. W związku z tym dopuszcza się sytuacje, że pewne metody nie zostaną w pełni zaimplementowane.

Aby nie dopuścić do sytuacji gdy w wyniku wywołania niezaimplementowanej metody zostaje rzucony wyjątek, przewidziano metodę „*getSupportedOperations*”. Przyjęliśmy założenie, że interfejs będzie oferował szereg metod typu „*getSupportedXXX*”, dzięki którym każdy serwis tłumaczący, mimo że jest oparty o ten sam *wsdl*, będzie mógł określić charakterystyczne dla siebie właściwości i ograniczenia.

Różni usługodawcy mogą mieć różne wymagania odnośnie dostarczenia pliku do tłumaczenia. Plik może być przekonwertowany do postaci tekstowej i wysłany jako string, można podać url z lokalizacją pliku a także przesłać go w klasie go opakowującej. Do wykrycia możliwych sposobów dostarczenia pliku służy metoda „*getSupportedSourceTypes*”.

W przypadku przesyłania pliku do tłumaczenia, powinniśmy także wiedzieć jakie formaty plików obsługuje serwis tłumaczący. Aby uzyskać taką informację, korzystamy z metody „*getSupportedFileFormats*”.

Usługodawcy mogą określić pary języków między którymi możliwe jest tłumaczenie – metoda „*getSupportedTranslations*”, istnieje również operacja pozwalająca sprawdzić możliwe tłumaczenia z danego języka źródłowego – „*getSupportedTranslationsForLanguage*”.

W przypadku gdy silnik translacji, udostępnia możliwości detekcji języka, metoda „*getSupportedDetections*” powinna zwrócić rozpoznawane przez serwis języki.

Jeżeli serwis tłumaczący określił, poprzez metodę „*getSupportedOperations*”, iż umożliwia on wybór jakości tłumaczenia, wówczas możemy sprawdzić oferowane poziomy jakości poprzez metodę „*getSupportedQualities*”.

Do zlecenia tłumaczenia służy metoda „*translate*”. Wywoływana jest ona z parametrem „*TranslationRequest*”.

W celu ujednolichenia sposobu przesyłania tekstu źródłowego przewidziano klasę „*ContentSource*”.

Wysłane żądanie tłumaczenia jest jednoznacznie identyfikowane przez „*translationRequestID*”. Jest on niezbędny przy późniejszym odesłaniu przetłumaczonego pliku lub informacji o nim.

Informacja o webserwisie obsługującym odpowiedzi od tłumaczy znajduje się w polu *CallbackEndpoint*. Pola *deadline* i *quality* mogą być wypełniane w przypadku gdy serwis oferuje możliwość ustalenia ostatecznego terminu ukończenia tłumaczenia oraz jego jakości tego tłumaczenia.

6.2.2. Interfejs odpowiedzialny za odbiór rezultatów tłumaczenia

Ten interfejs również jest określony za pomocą pliku *wsdl*. Posiada on dwie operacje pozwalające na zwrócenie wyników w formie *String* lub pliku. Aby jednoznacznie zidentyfikować tłumaczenie, usługodawca powinien przesłać *translationRequestID*, który został mu przekazany w momencie zlecenia tego tłumaczenia. Ponadto istnieje również metoda za pomocą której, usługodawca może nas powiadomić o postępach w procesie tłumaczenia.

6.3. Wykorzystywane typy danych

Podczas definiowania interfejsów dla ww. webserwisów, istotną kwestią było również dobór odpowiednich typów danych przesyłanych między naszym systemem a usługodawcami. Typy danych określone zostały za pomocą *XML Schema* – standardu opracowanego przez *W3C*, służącego do definiowania struktury plików XML.

Zdefiniowaliśmy wiele własnych enumeracji. Jedną z nich jest np. „Operation” – enumeracja służąca do określania operacji wspieranych przez serwis tłumaczący.

Wykorzystaliśmy również takie standardowe typy danych *XML Schema* jak *dateTime*, *URI* czy *string*. Ponadto dla określenia języków, używany jest typ *language*, który jest zgodny z *ISO-639* – norma, definiuje jednoznaczne reprezentacje nazw języków za pomocą krótkich łańcuchów znaków. *ISO-639* wykorzystuje klasa *Locale* języka Javy, a także *Google Translator*.

Zdecydowaliśmy się także w przypadku typów *ContentSource*, użyć typów abstrakcyjnych oraz mechanizmu rozszerzania oferowany przez *XML Schema*. Alternatywnym rozwiązaniem jest konstrukcja *choice*, jednak posiada ona nieintuicyjne mapowanie na język Java oraz oferuje ograniczoną rozszerzalność.

6.4. Dostęp do webservice'u tłumaczącego

Dostęp do webservice'u tłumaczącego od strony aplikacji realizowany jest poprzez poniższy interfejs:

«interface»RemoteWSInvoker
+ translateAsync(WSDescriptor webservice, TranslationRequest request)
+ List<Operation> getSupportedOperations(WSDescriptor webservice)
+ List<LanguagePair> getSupportedTranslations(WSDescriptor webservice)
+ List<Locale> getSupportedTranslationsPerLanguage(WSDescriptor webservice, Locale lang)
+ List<String> getSupportedQualities(WSDescriptor webservice)
+ List<FileFormat> getSupportedFileFormats(WSDescriptor webservice)
+ List<SourceType> getSupportedSourceTypes(WSDescriptor webservice)

WSDescriptor
String endpoint

7. Roboczy słownik pojęć

Zakładka konfiguracji tłumaczeń – Jest to jedna z zakładek w panelu „Personal Workspace” systemu Nuxeo, w której możemy definiować, edytować (tylko administratorzy) oraz wybierać silniki translacji, a także ustalić zbiór ulubionych języków

Własność pliku – jest to atrybut opisujący dany plik w systemie Nuxeo – atrybuty te możemy zobaczyć poprzez zakładkę „Summary” oraz „Edit” w panelu zarządzania dokumentem

Silnik translacji – zewnętrzna usługa oferująca możliwość tłumaczenia dokumentów, wykorzystywana przez plugin tłumaczeń. Silnikami translacji zarządzamy poprzez zakładkę konfiguracji tłumaczeń.

Panel zarządzania dokumentem – zestaw zakładek, który pojawia się po kliknięciu w dany plik. Umożliwiają one zarządzanie dokumentem.

Okno tłumaczenia – okno które pojawia się w trakcie zlecenia zadania asynchronicznego tłumaczenia – pomaga użytkownikowi określić język źródłowy, docelowy oraz docelową nazwę i lokalizację przetłumaczonego dokumentu.

Panel dashboard – panel który możemy wybrać z górnego paska menu systemu Nuxeo. W nim znajduje się widok na tłumaczone dokumenty.

Menu podręczne tłumaczenia – menu związane z zadaniem tłumaczenia asynchronicznego pojawiające się po kliknięciu prawego przyciska myszy wskazując na wybrany dokument.