



Akademia Górniczo-Hutnicza im. Stanisława Staszica
Wydział Elektrotechniki, Automatyki, Informatyki
Elektroniki

Rozprawa doktorska

**Kooperacyjne aplikacje
do wspomagania zdalnego
nauczania z wykorzystaniem
ontologicznej integracji wiedzy
w systemach P2P**

Aleksander Bułkowski

Promotorzy:

Prof. dr hab. inż. Andrzej Duda

Prof. dr hab. inż. Edward Nawarecki

Kraków, 2011

Składam szczególne podziękowania Panu Prof. Andrzejowi Dudzie i Panu Prof. Edwardowi Nawareckiemu za pomoc i wsparcie, dzięki którym ta praca mogła powstać.

Chciałbym również podziękować Panu Dr. Markowi Kisiel-Dorohinickiemu za cenne uwagi, które pomogły w przygotowaniu tej pracy.

Spis treści

Spis treści	i
Wstęp	1
1 Wprowadzenie	4
1.1 Istota technik P2P – Peer-to-Peer	6
1.2 Rozwój e-learningu z wykorzystaniem P2P	8
1.3 Proponowane rozwiązania	11
1.4 Teza i zakres pracy	12
1.5 Przegląd zawartości pracy	13
2 Stan wiedzy w dziedzinie e-learningu i systemów P2P	17
2.1 Rodzaje kształcenia e-learningowego	17
2.2 System LMS i standardy e-learningu	20
2.3 Standard SCORM	22
3 Kooperatywne nauczanie	26
3.1 Podstawy technik P2P	26
3.2 System Napster	27
3.3 Sieć Gnutella	28
3.4 Sieć FastTrack (Kazaa)	29
3.5 P2P a e-learning	30
3.5.1 P2P a PKM – Personal Knowledge Management . . .	30
3.5.2 Helpmate	32
3.5.3 Edutella	32
3.5.4 PADLR	34
3.5.5 Kierunki rozwoju istniejących rozwiązań	34
3.6 Wybór typu sieci P2P	35

4 Sieć P2P BitTorrent	37
4.1 Sieć BitTorrent z trackerem	39
4.2 BitTorrent w wersji trackerless	42
4.2.1 Zasady działania tablic mieszających DHT	44
4.2.2 Kademlia – implementacja DHT	45
5 Propozycja wykorzystania sieci P2P	51
5.1 Koncepcja obiektu nauczania	52
5.2 Indeksacja obiektów nauczania	57
5.3 Proces wyszukiwania obiektów nauczania	63
5.4 Semantyczna indeksacja i wyszukiwanie obiektów nauczania	64
5.4.1 Wielopoziomowa indeksacja semantyczna	65
5.4.2 Rozszerzenie o hierarchię pojęć	67
5.4.3 Proces wyszukiwania	68
5.5 Kooperacja pomiędzy uczestnikami grupy	69
6 Prototyp propagujący obiekt nauczania	73
6.1 Obiekt nauczania w prototypie GLEN	75
6.2 Prototyp GLEN	77
6.2.1 Struktura prototypu GLEN	77
6.2.2 Przykład użycia prototypu GLEN	81
6.2.3 Dyskusja	84
7 Ocena efektywności proponowanych rozwiązań	85
7.1 Wybór symulatora sieci P2P	87
7.2 Symulator propagacji obiektów nauczania	87
7.3 Rezultaty symulacji	92
7.3.1 Efektywność standardowej sieci DHT Kademlia	94
7.3.2 Efektywność sieci DHT Kademlia z propagacją obiektów	99
8 Podsumowanie i wnioski końcowe	101
Bibliografia	107
Dodatki	113
A Opis prototypu GLEN	114
A.1 Moduły przetwarzania	114
A.1.1 SCORM	114
A.1.2 LO	115
A.1.3 ONTOLOGIA	115

A.1.4	DHT	115
A.1.5	DOSTĘP	116
A.2	Interfejs użytkownika	116
A.2.1	ADD	116
A.2.2	INDEX	116
A.2.3	ONTOLOGY	116
A.2.4	SEARCH	117
A.2.5	RESULTS	117
Spis rysunków		118
Wykaz terminów i skrótów		121

Wstęp

Wykorzystanie dostępnych środków elektronicznych oraz powstających narzędzi programistycznych wspiera rozwój szkoleń oraz generuje tworzenie nowych systemów edukacyjnych. Rozwój systemów informatycznych powoduje olbrzymi wzrost informacji przechowywanych w światowych zasobach danych. Dotyczy to również szeroko pojętych treści edukacyjnych. Duża ilość przechowywanych danych utrudnia skuteczny dostęp do nich, jak również wybór właściwego pod względem merytorycznym materiału. Także sposób selekcji, szybkość transferu, możliwość komunikacji między uczestnikami procesu edukacyjnego stanowi barierę ich rozwoju. W konsekwencji, prowadzone są ustawicznie prace nad tworzeniem i modyfikacją systemów propagacji wiedzy z wykorzystaniem najnowszych technologii informatycznych.

E-learning reprezentuje nowoczesną formę kształcenia z uwzględnieniem różnorodnych mediów elektronicznych takich jak WWW, materiały multimedialne (audiovideo), telewizję interaktywną, nośniki pamięci CD i DVD. Uczeń ma możliwość wyboru oferowanego materiału edukacyjnego, tempa przyswajania i weryfikacji wiedzy. Tym samym przejmuje on większą odpowiedzialność za swoje doksztalcenie.

Wprowadzanie nowych rozwiązań w e-learningu stanowi o postępie i uzupełnieniu tradycyjnych metod nauczania. Obecne rozwiązania opierają się na wypróbowanych tradycyjnych technikach WWW lub na przesyłaniu strumieni multimedialnych. Powstające obiekty nauczające mogą również

stanowią narzędzie dla tworzonych spontanicznie kooperacyjnych grup zainteresowań. Dotychczas technologie typu Peer-to-Peer (P2P) nie zostały jeszcze docenione jako sposób, który mógłby wzbogacić funkcjonalność systemów e-learningu. Techniki P2P mogą pozwolić na:

- łatwe i szybkie rozpowszechnianie materiałów nauczania,
- demokratyzację uczenia się,
- tworzenie kooperacyjnych grup uczących się z wykorzystaniem wzajemnej współpracy,
- ułatwienie komunikacji pomiędzy wszystkimi uczestnikami (autor, nauczający, uczący się).

Aby zrealizować powyższe cele systemy tego typu muszą przede wszystkim zapewniać:

- szerokie upowszechnianie materiałów dydaktycznych,
- precyzyjną indeksację,
- efektywne poszukiwania pożądaných materiałów,
- ułatwianie powstawania więzi społecznych pomiędzy użytkownikami systemu.

W niniejszej pracy rozważono problem utworzenia systemu edukacyjnego ułatwiającego tworzenie kooperacyjnych grup nauczania i propagację wiedzy sklasyfikowanej z wykorzystaniem ontologii oraz najnowszych technologii informatycznych. Głównymi celami pracy są:

- analiza możliwości wykorzystania systemów P2P do zwiększenia dostępności obiektów wiedzy,
- rozpatrzenie użycia systemów P2P do tworzenia grup kooperacyjnego nauczania,
- ułatwienie kontaktów pomiędzy użytkownikami,

- zaproponowanie rozwiązań na bazie technologii P2P oraz VoIP (Voice over IP) w e-learningu poprzez dostarczenie narzędzi informatycznych wzbogacających proces kooperatywnego nauczania,
- walidacja i ocena efektywności proponowanych rozwiązań.

Niniejsza praca jest rezultatem aktualnych badań wykonywanych we współpracy pomiędzy Akademią Górniczo-Hutniczą w Krakowie i Politechniką w Grenoble (Institut Polytechnique de Grenoble – Ensimag, Laboratoire d’Informatique de Grenoble).

Bardziej precyzyjne sformułowanie zamierzeń i określenie przyjętej tezy zostanie dokonane w następnym rozdziale (Rozdział 1) po przedstawieniu przeglądu podstawowych pojęć i scharakteryzowaniu aktualnego stanu wiedzy.

Rozdział 1

Wprowadzenie

E-learning jest jednym z najbardziej dynamicznie rozwijających się obszarów edukacji. Jego funkcjonowanie opiera się głównie na dwóch podstawowych technikach. W pierwszej tradycyjny wykład prezentowany przez wykładowcę nagrywany jest jako dokument audio-wizualny i rozpowszechniany jednocześnie (on-line) lub z opóźnieniem (off-line) za pomocą strumieni multimedialnych do grupy uczących się. Obraz i głos emitowany przez wykładowcę mogą być wzbogacone poprzez inne materiały nauczania, takie jak np. slajdy PowerPoint, tworząc złożoną strukturę multimedialną, w której każdy element zawartości jest zsynchronizowany w czasie, tak aby jego prezentacja wypadła w odpowiednim momencie.

Przykładem technologii umożliwiających takie funkcje są np. narzędzia firmy RealNetworks (RealAudio, RealPlayer) [44] jak również SMIL W3C (Synchronized Multimedia Integration Language) [54] oraz zespół norm definiujących zsynchronizowane dokumenty multimedialne wypracowany przez Konsorcjum W3C [55]. Nowym narzędziem, które zyskuje obecnie szerokie powodzenie w rozpowszechnianiu wykładów dźwiękowych jest *podcast*, dzięki któremu można otrzymać nagrania dźwiękowe do odtwarzania na urządzeniach MP3 (nazwa pochodzi od iPod, playera MP3 Apple).

Druga technika e-learningu opiera się na materiałach nauczania o często złożonej strukturze, które udostępniane są poprzez serwery WWW. Nauczający tworzy dokumenty na określony temat, przewiduje ich podział na jednostki nauczania (lekcje), wzbogaca je poprzez ćwiczenia lub zadania i kontroluje postęp nauki formułując sprawdziany lub egzaminy.

Takie materiały nauczania mogą mieć złożoną strukturę i wymagają odpowiednich standardów do zdefiniowania formatu plików, tak aby umożliwić ich wymianę i wykorzystanie. Rozpowszechnianie materiałów opiera się na standardowych systemach WWW i protokołach komunikacyjnych takich jak HTTP (HyperText Markup Language).

Zaletą opisanych technik zdalnego nauczania jest znaczna niezależność uczącego się od nauczającego – uczący może korzystać z materiałów w dowolnej chwili. Jednak często wymaga to pomocy nauczającego, co jest realizowane poprzez narzędzia komunikacyjne, takie jak: e-mail, forum, chat. Standardy SCORM (Sharable Courseware Object Reference Model) [2], ADL (Advanced Distributed Learning) [2] i IMS (Instructional Management Systems) [24] zaczynają być używane do integrowania materiałów w tym typie zdalnego nauczania. SCORM jest obecnie ogólnie akceptowanym standardem obok jego odpowiednika opracowanego przez IEEE LTSC (Learning Technology Standards Committee) – Learning Object Metadata (LOM) [23].

Pomimo dynamicznego rozwoju, e-learning nie osiągnął jeszcze możliwej do uzyskania popularności i efektywności. Istnieje kilka przyczyn takiego stanu rzeczy [18].

1. Dotychczasowe metody przedstawiają tradycyjny asymetryczny model nauczania – nauczający przekazuje wiedzę lub umiejętności uczącemu się.
2. W wypadku nauczania typu on-line, wymagana jest jednoczesna obecność nauczającego i uczących się.

3. Uczący się pracują zwykle w izolacji, brak jest pomiędzy nimi koordynacji i komunikacji.
4. Aktualnie stosowane rozwiązania wymagają stosunkowo dużych inwestycji (sprzęt do nagrywania, serwer WWW), nakładów operacyjnych (zarządzanie, uaktualnianie informacji) i fachowej obsługi.
5. Zdalne wykłady nie są szeroko rozpowszechnione z uwagi na potrzebę wyspecjalizowanych narzędzi (e.g. player/interpreter formatu SCORM) i potrzebną wiedzę techniczną do ich użytkowania.

1.1 Istota technik P2P – Peer-to-Peer

W tradycyjnych *systemach rozproszonych* (*distributed systems*) komputery wyposażane są w oprogramowanie umożliwiające współdzielenie zasobów systemowych i wykonywanie operacji na dystans. Funkcjonują one według modelu *klient-serwer*, w którym jeden zbiór oprogramowania zapewnia usługi dla innego zbioru, a z usług jednego serwera korzysta wielu klientów. Najlepszym przykładem systemu *klient-serwer* jest infrastruktura WWW, która od wielu lat używana jest jako podstawowa technika propagacji treści nauczających.

Model ten jest obecnie wypierany przez *peer-to-peer* (P2P), co w dosłownym tłumaczeniu z języka angielskiego oznacza *równy z równym*. Jest to stosunkowo młody model komunikacji w sieci internetowej, który w przeciwieństwie do modelu *klient-serwer* opiera się na równorzędności obu stron procesu komunikacji. W sieci P2P każdy komputer może jednocześnie pełnić obie funkcje zarówno klienta jak i serwera.

Najbardziej rozpowszechnionym przykładem stosowania modelu P2P są programy służące do wymiany plików w Internecie. Każdy komputer użytkownika tworzy tzw. *węzeł* sieci i odgrywa rolę serwera przyjmując połączenia od innych użytkowników jak i klienta, łącząc się i pobierając dane z innych węzłów działających w tej samej sieci, przy czym wymiana danych

prowadzona jest bez pośrednictwa centralnego serwera. Funkcjonowanie sieci P2P wykorzystuje zasadę *overlay* – węzeł sieci nawiązuje kontakt z innymi węzłami na poziomie protokołu TCP (Transmission Control Protocol) lub UDP (User Datagram Protocol) w pełnym odizolowaniu od struktury fizycznej Internetu. Struktura sieci P2P podlega zatem nieustannej dynamice w zależności od tego jakie komputery, z jakimi udostępnionymi danymi są aktualnie do niej podłączone. Można obecnie zaobserwować fakt, że coraz większa część ruchu w sieci Internet nie pochodzi od protokołu HTTP, który pozwala na dostęp do serwerów WWW, lecz jest generowana przez systemy wymiany plików typu P2P.

Systemy P2P konstruują rozproszoną strukturę poszukiwania i dostępu do plików muzycznych MP3 lub filmowych takich jak DivX lub MPEG4. Ich podstawą są rozproszone tablice haszujące (DHT – Distributed Hash Tables), które odwzorowują unikalny identyfikator pliku do jego lokalizacji. Dostęp może być również przyspieszony poprzez równoległe ściąganie różnych części pliku z kilku lub kilkunastu serwerów. Aktualne systemy nie pozwalają na współpracę użytkowników, a indeksacja zawartości plików jest trywialna. Poszukiwanie odbywa się na podstawie tytułu lub autora, co następnie pozwala na określenie identyfikatora i odnalezienie pliku.

Problematyka wykorzystania sieci P2P [1] stanowi przedmiot badań wielu ośrodków naukowych, które analizują i rozbudowują technikę obsługującą większość przesyłów w Internecie. Chociaż dotychczas dla wytwórni muzycznych i filmowych oraz stacji telewizyjnych wiąże się to z problem piractwa i naruszania praw autorskich to jednak nie da się pominąć tej formy dystrybucji materiałów audio-video poprzez Internet. Sieci P2P oferują niewykorzystane możliwości przesyłu plików i innych materiałów bez użycia własnej infrastruktury. Przy aprobacie Unii Europejskiej i jej wsparciu finansowym w 2008 roku uruchomiono program P2P-Next do którego

przystąpiły instytuty badawcze z wielu krajów Europy, jak również Europejskie Stowarzyszenie Nadawców, BBC i Pioneer. Inicjatywy te świadczą o dynamicznym rozwoju systemów P2P.

Z punktu widzenia e-learningu systemy P2P przedstawiają szerokie możliwości, jednak wyspecjalizowane są przede wszystkim do poszukiwania i rozpowszechniania plików audio lub video. Istnienie tak bogatej technologii wymiany informacji skłania do rozpatrzenia możliwości wykorzystania jej również do celów rozpowszechniania materiałów nauczania. Jednak dodatkową przeszkodą jest zbyt prosta forma indeksowania plików na podstawie słów kluczowych, podczas gdy materiały nauczania mogą posiadać wysoce złożoną zawartość, która może wymagać indeksacji na podstawie metadanych.

Inny problem to kwestia nawiązywania kontaktu z użytkownikami systemów P2P i ewentualnej komunikacji z nimi. W systemach P2P, użytkownik poszukując specyficznych informacji nie ma możliwości nawiązania kontaktu pomiędzy swoim systemem, oczekującym pewnej zawartości a systemem, który ją udostępnia. Aktualne systemy P2P nie dostarczają również informacji o innych osobach pobierających materiały o tej samej tematyce. W wielu przypadkach interesujące byłoby tworzenie grup zainteresowań i ich kooperacji na podstawie dostępu do plików o podobnych charakterystykach.

1.2 Rozwój e-learningu z wykorzystaniem P2P

W jaki sposób technologie P2P mogą być wykorzystane do nauczania? Co mogą wnieść nowego do e-learningu? Wydaje się, że ta nowa technologia pozwoli na wzbogacenie możliwości e-learningu poprzez kooperacyjne zdalne nauczanie. Można to będzie osiągnąć na podstawie technik P2P dzięki:

1. definicji nowych złożonych obiektów nauczania rozprowadzanych poprzez zmodyfikowane technologie P2P,

2. asocjacji metadanych z obiektami nauczania dla precyzyjnego indeksowania ich zawartości,
3. zaadoptowaniu ontologii dziedzinowych do precyzyjnego opisu zawartości obiektów nauczania,
4. wzbogaceniu obiektów nauczania o możliwości łatwego kontaktu pomiędzy użytkownikami, tak aby umożliwić tworzenie się spontanicznych grup współpracy (wymagać to będzie na przykład zintegrowania komunikacji interaktywnej z obiektami nauczania).

Pierwszy problem dotyczy wykorzystania technologii P2P do udostępniania materiałów edukacyjnych, a nie tylko plików muzycznych lub video. Materiały dydaktyczne mają zazwyczaj złożoną strukturę i skład, na przykład są to artykuły, prezentacje PowerPoint, zarejestrowane segmenty video lub audio, częściowo lub całkowicie zsynchronizowane. Refleksja na ten temat musi dotyczyć formatów aktualnie zdefiniowanych w e-learningu, takich jak SCORM lub IMS i ich użycie w definicji obiektów rozprowadzanych poprzez technologie P2P.

Zawartość złożonych obiektów nauczania powinna być odpowiednio indeksowana dla łatwego wyszukiwania. Pierwszą próbą w tym kierunku była Edutella [39]. System ten wykorzystuje standardy metadanych zdefiniowanych dla stron WWW i opartych na specyfikacji RDF (Resource Description Framework) [53]. Edutella używa zbioru protokołów komunikacyjnych JXTA [29] pozwalających na komunikację bez pośrednictwa serwerów centralnych. Edutella uzupełnia proces nauczania w systemie e-learningu, ale go nie zastępuje – dążeniem twórców Edutelli było stworzenie zdecentralizowanej infrastruktury do przeszukiwania zasobów e-learningowych.

Podobne idee na temat zastosowania technik P2P do celów edukacyjnych pojawiły się w literaturze. Berman i Annexstein proponują integrację technik P2P z personalnymi systemami zarządzania wiedzą (Personal Knowledge Management), aby stworzyć nowy system nauczania [9]. Ich propozycja pozostała jednak w sferze idei bez stosownej implementacji.

Istotnym jest też rozpatrzenie możliwości kooperacyjnego korzystania z materiałów do nauczania na bazie technologii P2P. Obecne systemy edukacyjne przystosowane są do użytkowania przez jednego ucznia, natomiast efektywniejszy proces nauczania można osiągnąć poprzez wspólną naukę w grupie kilku osób, które wspólnie pracują nad tym samym zagadnieniem. Efektywność takiego nauczania wychodzi z nowych postulatów pedagogicznych takich jak *learning by teaching*, gdzie uczeń doskonale się uczy, gdy musi wytłumaczyć jakiś problem komuś innemu. Konkludując można stwierdzić, że obok zawartości rozpowszechnianej technologią P2P, potrzebna jest również komunikacja interaktywna. Taką technologią jest Voice over IP (VoIP) [50]. Jest ona wykorzystywana przez komunikatory ogólnie używane oraz w telefonii internetowej umożliwiając łatwą, tanią komunikację pomiędzy użytkownikami sieci. Najpopularniejszym przedstawicielem tych technologii jest system Skype [22, 6].

Wydaje się zatem, że nowoczesne systemy edukacyjne będą mogły wiele zyskać na bazie technologii P2P wykorzystanej do tworzenia kooperacyjnych grup nauczania i wspomaganej poprzez interaktywną komunikację. Przykładami takich możliwości mogą być dwa następujące scenariusze:

1. grupy nauki kooperacyjnej: kilka osób korzysta z materiałów wykorzystywanych do nauczania otrzymanych drogą P2P i dzięki nim nawiązuje ze sobą kontakt, aby dyskutować problemy, których sami nie potrafią rozwiązać,
2. kooperacja nauczający – uczeń: osoba korzysta z materiałów do nauczania otrzymanych drogą P2P i dzięki nim nawiązuje kontakt z nauczającym lub autorem materiałów, który efektywnie pomaga rozwiązać problemy lub odpowiada na zadane pytania.

Wzbogacenie aktualnych systemów nauczania o takie funkcje wymaga więc rozpatrzenia problemu w jaki sposób należy zmodyfikować technologie

P2P do celów nauczania i jak je zintegrować z innymi technologiami takimi jak VoIP, aby umożliwić realizację tego nowego modelu zdalnego nauczania.

1.3 Proponowane rozwiązania

Celem niniejszej pracy jest zbadanie możliwości wykorzystania systemów P2P w zdalnym nauczaniu [13, 11] poprzez zdefiniowanie nowego typu pakietu rozpowszechniającego zawartości oraz stworzenia nowych procedur indeksacji w oparciu o szczegółowy opis semantyczny (metadane opisujące zawartość) [38]. Należy również usystematyzować zasoby, poprzez ontologiczną integrację wiedzy o tychże zasobach, na podstawie istniejącego standardu tworzenia platform e-learningowych – SCORM.

Dodatkowym elementem, który może rozszerzyć funkcje kooperacyjnych aplikacji wspomagania zdalnego nauczania jest rozwój systemów umożliwiających łatwą i tanią komunikację poprzez VoIP (Voice over IP). Celem badań jest integracja komunikacji typu VoIP z pakietami nauczania dla tworzenia grup kooperacyjnej nauki. Zamierzeniem prowadzonych prac jest zatem zrealizowanie następujących zadań:

- Zdefiniowanie nowego typu pliku reprezentującego zawartości wzbogacające zdalne nauczanie – pakietu nauczania. W takim pakiecie można zintegrować różne typy danych i dokumenty (tekst, prezentacja PowerPoint, zapis dźwiękowy lub video, formularze HTML), tak aby tworzyły jedną całość.
- Usystematyzowanie zasobów poprzez ontologiczną integrację wiedzy o pakietach nauczania. Wychodząc z istniejącego standardu wymiany dokumentów nauczania SCORM, należy rozpatrzyć rozszerzenie jego funkcji celem umożliwienia precyzyjnej indeksacji zawartości.
- Połączenie pakietu z jego autorem, nauczającym lub z jakąkolwiek inną osobą, która mogłaby interaktywnie współpracować podczas

nauki. Pakiet będzie zawierał obiekt przełączający do identyfikatora VoIP osoby koordynującej naukę.

- Rozpowszechnianie pakietu nauczania w sieci P2P i efektywne jego wyszukiwanie. Pakiet i jego metadane będą indeksowane tak aby ułatwić jego efektywne wyszukiwanie.
- Umożliwienie tworzenia kooperacyjnych grup nauki. Rozpowszechnianie obiektu nauczania w systemie P2P pozwoli na utworzenie grup nauczających zainteresowanych w dziedzinie danego obiektu. Tak utworzone zespoły korzystające z pakietu nauczania mogłyby dyskutować nad zawartością repozytorium i pogłębiać swą wiedzę używając technik łatwej komunikacji na bazie VoIP. Taka forma kooperacji umożliwia nowy sposób nauczania, w której uczestnik, gdy zrozumie pewne części wykładu, mógłby odpowiedzieć na pytania lub wyjaśnić te aspekty innym członkom grupy.

1.4 Teza i zakres pracy

Teza niniejszej pracy jest następująca:

- Na bazie systemów P2P istnieje możliwość integracji wiedzy o rozproszonych zasobach nauczania oraz możliwość konstrukcji kooperacyjnych aplikacji zdalnego nauczania.

Prawdziwość tezy zostanie wykazana poprzez:

- analizę aktualnych sieci P2P oraz ocenę ich przydatności do zdalnego nauczania,
- rozpatrzenie standardu SCORM i możliwości jego wykorzystania do usystematyzowania pakietów nauczania,
- propozycję rozszerzenia istniejących sieci P2P do tworzenia kooperacyjnych grup nauczania,

- specyfikację metod precyzyjnego indeksowania i odszukiwania pakietów nauczania,
- specyfikację prototypu integrującego wybrane systemy,
- implementację podstawowych komponentów prototypu,
- walidację i ocenę efektywności prototypu.

Działania te określają zakres prowadzonych badań.

1.5 Przegląd zawartości pracy

Tworzony podczas badań obiekt nauczania stanowić miał narzędzie edukacyjne umożliwiające jego wykorzystanie przez dowolne grupy kooperacyjne powstałe na gruncie wspólnoty zainteresowań. Rozdział 2 analizuje aktualny stan wiedzy w dziedzinie e-learningu i istniejących obiektów nauczania. Standaryzacja nauczania w e-learningu stanowi główny czynnik ukierunkowania rozwoju tej dziedziny wiedzy. Prym w tym zakresie wiodą renomowane instytucje międzynarodowe zajmujące się definicją złożonych obiektów nauczania i ich propagowaniem. Następnie omówiono przykładowe platformy edukacyjne oraz sposób zarządzania nimi. Przedstawiono główne światowe standardy tworzenia platform e-learningowych. Szczególną uwagę poświęcono standardowi SCORM wykorzystywanemu do budowy prototypu obiektu nauczania. Rozdział ten omawia również problemy i niedociągnięcia aktualnie działających rozwiązań e-learningowych.

Wprowadzenie do sieci P2P jest tematem następnego rozdziału (Rozdział 3). Przybliżono w nim najistotniejsze systemy P2P – ich wady i zalety. Celem rozważań była możliwość ich wykorzystania w zakładanej konstrukcji obiektu nauczania. Przedstawiono podstawowe systemy P2P takie jak Napster, Gnutella i FastTrack. Rozpatrzono również pierwsze zastosowania sieci P2P do propagowania metadanych o obiektach nauczania w postaci systemu Edutella i do zarządzania zasobami wiedzy. Rozważania skutkują wskazaniem sieci P2P BitTorrent wybranej do realizacji zakładanego celu.

Konsekwencją wyboru BitTorrenta jako bazy walidacji proponowanych rozwiązań jest jego przedstawienie w Rozdziale 4. Szczegółowa analiza klienta BitTorrent Azureusa miała decydujący wpływ na jego późniejszą adaptację do potrzeb propagacji proponowanego obiektu nauczania. Fakt użycia rozproszonych tablic mieszających DHT do umieszczania i wyszukiwania obiektów w sieci P2P skutkowało późniejszym kształtem zastosowanych rozwiązań. Szczególną rolę odegrał protokół Kademlia wykorzystany w propagacji prototypu do nowatorskiego sposobu indeksacji złożonego obiektu nauczania. Rozdział 4 kończy pierwszą część pracy poświęconą sformułowaniu tezy i przedstawieniu aktualnego stanu wiedzy.

Rozdział 5 przedstawia oryginalne rozwiązania umożliwiające osiągnięcie celów niniejszej pracy poprzez propozycje zastosowania technik P2P do tworzenia grup kooperacyjnego nauczania. Zdefiniowana została struktura złożonego obiektu nauczania przystosowana do rozpowszechniania różnych treści, nie tylko edukacyjnych, w systemie P2P i jego asocjacje z metadanymi. Zaproponowano metodę reprezentacji relacji pomiędzy komponentami złożonego obiektu nauczania tak, aby można je było zapisać do struktury DHT. Przedstawiono również techniki indeksacji na bazie ontologii i ich sposób zapisywania do struktury DHT, co pozwala na efektywne odszukiwanie obiektów.

Omawiany Rozdział 5 rozpatruje formułowanie zapytań przez użytkownika i jednopoziomową metodę odszukiwania polegającą na odczytywaniu z DHT wszystkich par związanych z danym zapytaniem i ich redukcji w zależności od relacji logicznej zdefiniowanej w zapytaniu. Zaproponowano również rozszerzenie hierarchicznej metody wielopoziomowego poszukiwania obiektów w DHT o użycie ontologii jako podstawy do definicji stopniowo szczegółowych zapytań. W przeciwieństwie do metody jednopoziomowej, indeksy nie zawierają wprost identyfikatorów obiektów, lecz inne coraz bardziej wyspecjalizowane zapytania. Użytkownik rozpoczyna poszukiwanie od sformułowania zapytania ogólnego i stopniowo odkrywa zapytania

bardziej specjalizowane, aż do uzyskania jednoznacznej informacji o poszukiwanych obiektach, która pozwala na dostęp do nich.

Ostatnie części pracy dotyczą eksperymentalnej walidacji proponowanych rozwiązań poprzez implementacje prototypu i symulacje jego działania na szeroką skalę. Rozdział 6 przedstawia GLEN (Global Lecture Exchange Network) – prototyp systemu upowszechniania obiektów nauczania zbudowany jako rozszerzenie sieci BitTorrent i aplikacji Azureus, który implementuje proponowane rozwiązania. GLEN rozszerza działanie Azureusa i używa zewnętrznego systemu OpenDHT do zapamiętywania atrybutów indeksacji i relacji pomiędzy identyfikatorami. GLEN oferuje użytkownikowi funkcje dodawania nowego obiektu nauczania w formacie SCORM 1.2 do systemu propagacji.

Obiekt jest indeksowany słowami kluczowymi, a użytkownik może wprowadzić dodatkowe indeksujące pojęcia z dziedzinowej ontologii. Wszystkie potrzebne atrybuty zapisywane są do OpenDHT. Użytkownik ma dostęp do funkcji poszukiwania obiektów za pomocą wprowadzania słów kluczowych lub wyboru pojęć z dziedzinowej ontologii. Ta metoda pozwala na dokładne sprecyzowanie zapytań dotyczących obiektów jak również na progresywne odkrywanie obiektów począwszy od najbardziej ogólnych.

Rozdział zawiera opis interfejsu GLEN oraz sposoby umieszczania obiektu w sieci P2P i jego wyszukiwania. Przykłady ilustrują wprowadzanie metadanych wykorzystywanych do opisu oraz przyporządkowanie obiektu do drzewa ontologii dziedzinowej klasyfikującej pojęcia opisujące obiekt.

Ocenę efektywności propozycji prezentuje Rozdział 7, w którym opisano wykorzystywanie symulatora OMNet++ i OverSim do badania zachowania rozwiązań w sieci o dużej liczbie węzłów. Opisano zachowanie prototypu GLEN w sieci P2P wykorzystującej protokół Kademlii uwzględniając średni czas obecności węzła w sieci. Symulacje przedstawia wpływ generacji znacznej ilości zapisów par danych w tablicy mieszającej DHT na średni czas

zapisywania. Podobnie badano czas oczekiwania na rezultaty poszukiwań wielu zapytań w DHT.

Pracę kończą wnioski i propozycje przyszłych badań (Rozdział 8).

Rozdział 2

Stan wiedzy w dziedzinie e-learningu i systemów P2P

Głównym dążeniem w procesie edukacji jest rozwój systemów kształcenia poprzez dobór właściwych metod nauczania i tworzenie programów służących wzbogacaniu poziomu i kwalifikacji jej uczestników. E-learning jest jednym z instrumentów, który dla wielu osób radykalnie zmienił drogę dostępu do wiedzy. Fizyczne oddalenie wykładowcy i odbiorcy kształtuje kolejną dyscyplinę nauczania na odległość – *distance learning*. E-learning odnosi się zarówno do tego aspektu jak i do zastosowania nowoczesnych technologii informatycznych w nauczaniu. Znaczna część programów edukacyjnych przyjmuje w założeniu zmiany w systemie transferu wiedzy, bazując na nauczaniu interaktywnym przez Internet [32].

2.1 Rodzaje kształcenia e-learningowego

W kształceniu e-learningowym generalnie wyróżnia się cztery podstawowe formy:

- Samokształcenie. Uczący korzysta z zasobów wiedzy literaturowej bez udziału prowadzącego.

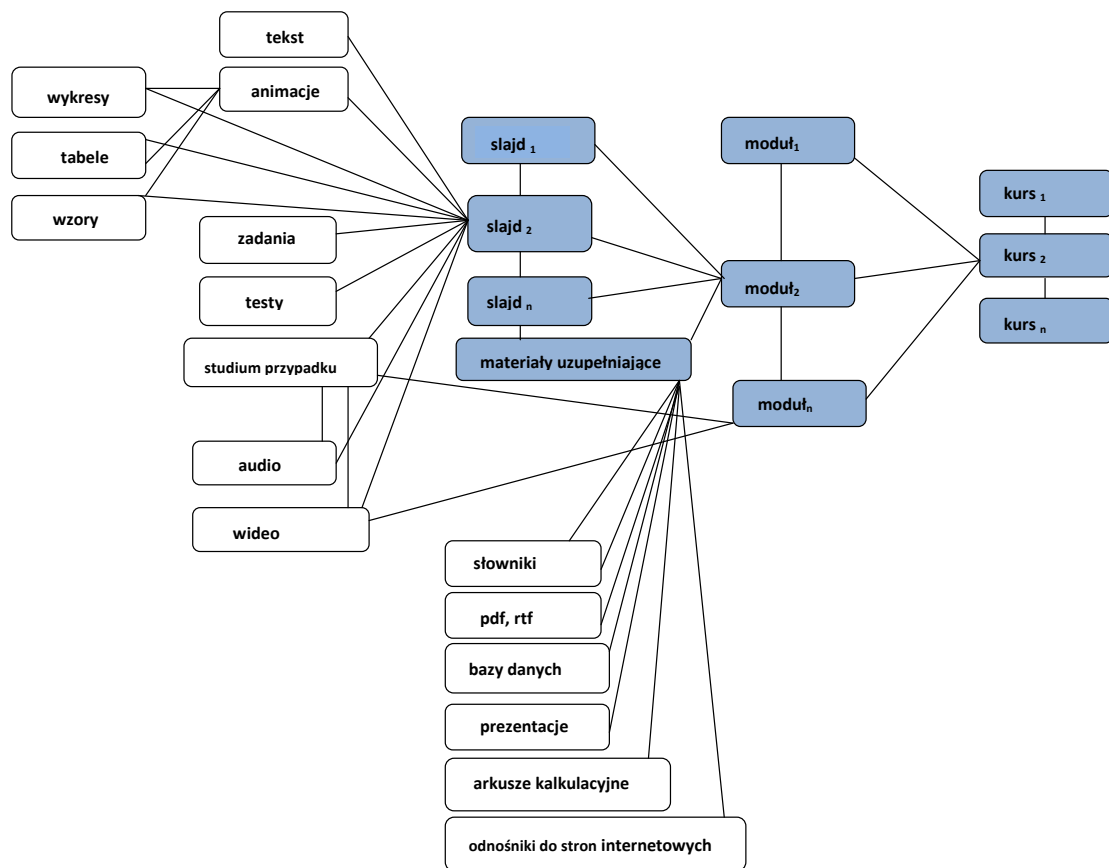
- Nauczanie asynchroniczne. Uczestnicy procesu edukacyjnego nie muszą być jednocześnie w tym samym miejscu i czasie. Uczeń korzysta z materiałów zgodnie ze swoimi oczekiwaniami i możliwościami. Kontakt z prowadzącym odbywa się według określonego przez niego reguł w *pokojach rozmów* (chat), poprzez konsultacje telefoniczne lub bezpośrednio. Ze względu na niskie koszty forma ta uzyskała dużą popularność.
- Nauczanie synchroniczne. Studenci i prowadzący muszą być obecni podczas procesu kształcenia w tym samym czasie. Nauczanie tego typu umożliwia interakcję w czasie rzeczywistym. Stwarza warunki do pracy indywidualnej i grupowej, prezentowania materiałów, prowadzenia dyskusji, bezpośrednią kontrolę studentów.
- Nauczanie mieszane (*blended learning*). Łączy nauczanie tradycyjne z internetowym.

Przykładowo Szkoła Główna Handlowa w Warszawie prowadzi działalność uzupełniania zajęć materiałami online. Uczelnia wspomaga tradycyjne formy kształcenia wykorzystując własną, autorską platformę nauczania przez Internet na stronie WWW.e-sgh.pl (cf. Rys. 2.1¹).

Platforma została zbudowana i jest rozwijana w oparciu o bezpłatne oprogramowanie. Do tworzenia materiałów wykorzystywany jest standard opisu dokumentów XML, dzięki czemu transfer i wykorzystanie treści dydaktycznych ma charakter uniwersalny [18]. Tworzone są całe pakiety umożliwiające logowanie się do kursu, korzystanie z bazy danych przygotowanej dla uczestników oraz łączność z zewnętrznymi źródłami informacji (strony WWW).

Generalnie dla uczestników kursu przygotowuje się system testów sprawdzających przyswajanie wiedzy oraz różne formy kontaktu z prowadzącym

¹Źródło: opracowanie własne. Wszystkie rysunki w niniejszej pracy pochodzą z opracowania własnego jeżeli nie jest podane inaczej.



Rysunek 2.1: Struktura treści dydaktycznych na platformie www.e-sgh.pl.

szkolenie. Część firm podjęła działania w celu zastosowania wspólnego standardu tworzenia platform e-learningowych. Standard ten umożliwia sporządzanie kursów za pomocą języków zorientowanych obiektowo (*learning objects*).

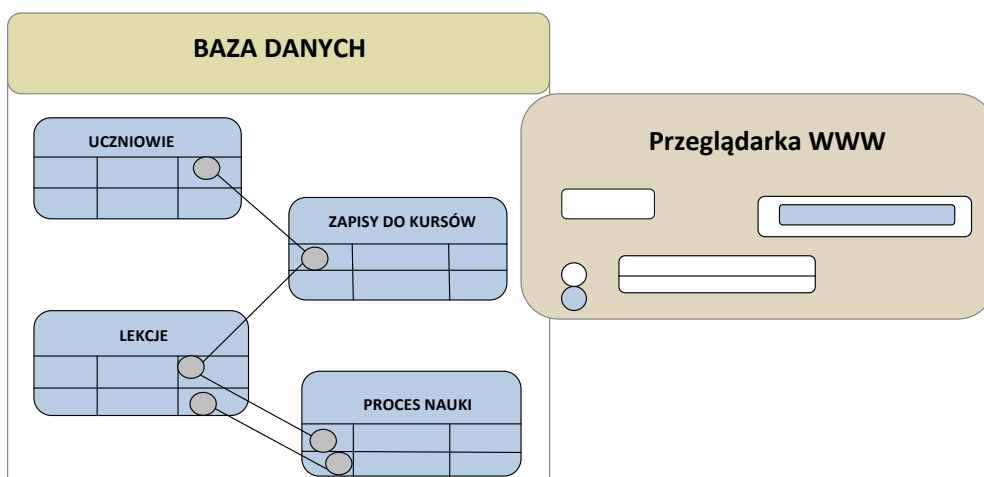
Wadami rozwiązań platform opartych na gromadzeniu treści edukacyjnych na serwerach WWW są:

- Serwer jest pojedynczym newralgicznym punktem, więc jego wyłączenie lub uszkodzenie uniemożliwia korzystanie z wykładów przez jego klientów.

- Konieczność zarządzania siecią i utrzymywanie infrastruktury dostępu do serwera WWW.

2.2 System LMS i standardy e-learningu

Instytucje wdrażające platformy e-learningowe tworzą dla nich systemy zarządzania procesem nauczania – LMS (Learning Management System) [56] oraz LCMS (Learning Content Management System) [56]. LMS umożliwia administrowanie kursami i prezentację treści, raportowanie aktywności szkoleniowej uczestnika kursu oraz decyduje o możliwości korzystania z poszczególnych modułów kursu w zależności od posiadanego statusu i nadanych uprawnień.



Rysunek 2.2: LMS jako aplikacja bazodanowa działająca w oparciu o przeglądarkę WWW.

Wykorzystanie narzędzi systemu LMS jako kanału komunikacji pomiędzy uczestnikami kursu i prowadzącym poprawiają efektywność szkolenia poprzez wzrost skuteczności dystrybucji materiałów dydaktycznych i ułatwienie wzajemnych kontaktów. Z technicznego punktu widzenia Learning

Management System jest aplikacją działającą w oparciu o przeglądarkę internetową (cf. Rys. 2.2) [30].

LMS zapisuje w odpowiedniej tabeli, w bazie danych informacje o przebiegu procesu nauczania. Gromadzi również dane o przyporządkowaniu studentów do odpowiednich kursów oraz wszelkie inne informacje potrzebne do poprawnego wykonywania funkcji przyporządkowanych LMS. Wykorzystuje on, jak większość aplikacji bazodanowych, relacyjny model baz danych, czyli taki model w którym dane są zapisywane w tabelach posiadających wzajemne odniesienia do siebie.

Interfejsem pomiędzy taką bazą danych a systemem e-learningowym jest przeglądarka WWW. Za pomocą LMS system zapisuje i pobiera dane. LMS jest procesem nadzorującym, działającym na każdej płaszczyźnie edukacji e-learningu. Każda z osób mających kontakt z tym systemem podlega bezpośredniej lub pośredniej interakcji z nim. Twórca kursu, za pomocą LMS uzyskuje informacje o postępach swoich studentów oraz o wszelkich zdarzeniach zachodzących w procesie nauczania. Studenci za pomocą tego systemu uzyskują dostęp do przeznaczonych dla nich kursów. Natomiast dla administratora stanowi on podstawowe narzędzie pracy wspierające wszelkie rodzaje zarządzania platformą, kursami oraz uczestnikami systemu e-learningu.

System zarządzania treścią szkoleniową – LCMS dodatkowo wspomaga tworzenie, zarządzanie i ulepszanie szkoleń. Stanowi on narzędzie będące integralną częścią e-learningu. Zaawansowane systemy LCMS kontrolują interakcje studenta z materiałem dydaktycznym i bazując na tej informacji dostarczają uczestnikowi kursu spersonalizowany tok nauczania. Niektóre z wiodących systemów LCMS umożliwiają współpracę i wymianę informacji pomiędzy studentami i nauczycielami. Posiadają również moduły służące do tworzenia treści dydaktycznych WBT (Web Based Training).

Dla autorów szkoleń system ten umożliwia tworzenie i przechowywanie treści dydaktycznych, a dla uczestników łatwy dostęp do nich. Istotnym aspektem funkcjonalności systemów LMS jak i LCMS jest wspieranie określonego standardu tworzenia platform e-learningowych. Chodzi o dostarczenie narzędzi umożliwiających eksportowanie utworzonego w nim kursu w powszechnie używanym i dostępnym standardzie. W efekcie utworzony kurs będzie “wyglądał” i “zachowywał się” tak samo w każdym systemie e-learningowym wspierającym ten standard.

Pomimo istnienia wielu opinii na temat tworzenia efektywnego przekazu wiedzy on-line można wyróżnić niektóre uniwersalne zasady tworzenia platform szkoleniowych. Kurs powinien instruować zainteresowanego o możliwości poruszania się w jego obrębie jak również komunikowania się z jego pozostałymi uczestnikami. Istotne są informacje dla jakiego odbiorcy szkolenie jest dedykowane i jakie narzuca wymagania odnośnie sprzętu posiadanego przez użytkownika.

Opis kursu jest szczególnie ważny dla osób, które wcześniej nie kształciły się on-line, jak również dla tych, którzy doskonalą się w ich korzystaniu. Należy mieć na uwadze, że ta forma nie ma na celu zastąpienia tradycyjnych metod nauczania, ale posiadając pewne wspólne z nimi cechy, ich wzbogacanie.

2.3 Standard SCORM

W niniejszej pracy skoncentrowano się na standardzie IMS/SCORM (Instructional Management System, Sharable Content Object Reference Model) [33]. Obecnie kilka organizacji zajmuje się tworzeniem standardu do opisu treści e-kursów. Należą do nich: Aviation Industry CBT Committee (AICC), IEEE Learning Technology Standards Committee, IMS Global Consortium Advanced Distributed Learning, Advanced Distributed Learning (ADL).

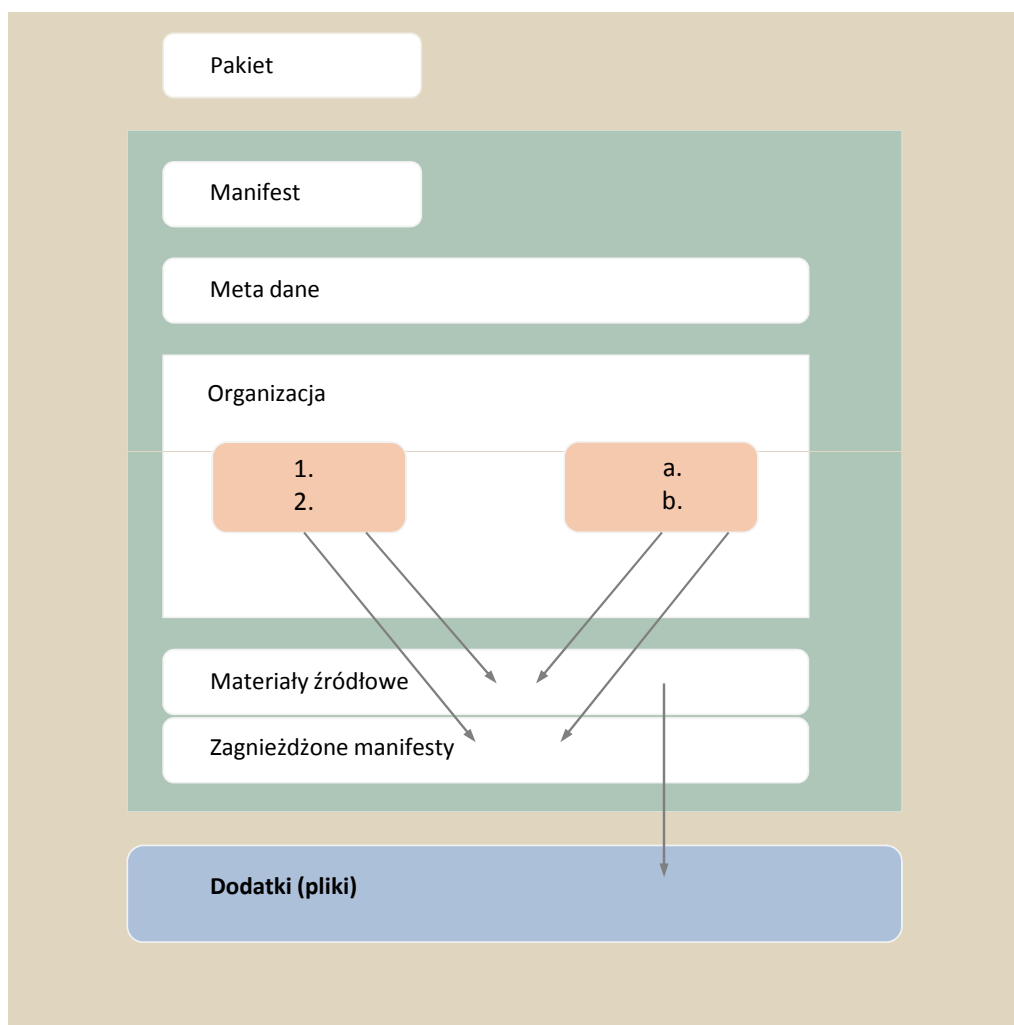
Instytucje tworzące SCORM nie proponują nowych standardów, lecz adoptują najlepsze rozwiązania innych pomysłodawców. Dlatego wiele z zaleceń SCORM pokrywa się z tym co proponują organizacje AICC czy IMS. Dzięki takiej polityce, ten sposób budowy platform e-learningowych jest obecnie najszerzej wspieranym standardem.

SCORM jest zbiorem specyfikacji, które opisują w jaki sposób tworzyć kursy oparte o media elektroniczne, aby mogły być łatwo przenoszone pomiędzy różnymi systemami zarządzania nauczaniem (LMS). Wskazują również jakie działania musi podjąć system, aby poprawnie przekazać uczestnikowi zawartość kursu w pakiecie SCORM. Standard ten nie opisuje wszystkich aspektów kursu w nauczaniu na odległość. Przykładowo nie mówi w jaki sposób i gdzie system LMS ma przechowywać treści dydaktyczne, nie wskazuje jaki jest model nauczania oraz jaki rodzaj materiałów zawiera pakiet SCORM.

Powyższy standard skupia się na takim opisie tworzenia kursu i reakcji systemu LMS, tak aby jego zawartość dydaktyczna była prezentowana kursantowi według zamierzeń jego twórcy. Podstawowy obiekt standardu SCORM, możliwy do wyświetlenia w przeglądarce WWW nosi nazwę SCO (Shareable Content Object) (cf. Rys.2.3). Wyróżnia się dwa typy obiektu SCORM:

- SCORM podstawowy, wersja SCORM 1.1 i 1.2 (zawiera strony HTML, dokumenty PDF itp.) – obiekt, który w minimalnym stopniu korzysta z interfejsu standardu SCORM w kontaktach z systemem LMS.
- SCORM rozbudowany, wersja SCORM 2004 – w pełni korzysta z interfejsu standardu SCORM, wysyłając i odbierając dane i komunikaty z LMS.

Całość kursu w standardzie SCORM, składająca się z pojedynczych obiektów SCO zawiera się w pakiecie stworzonym przez IMS. Jego struktura jest przedstawiona na Rysunku 2.3. Pakiet SCORM składa się z manifestu



Rysunek 2.3: Schemat pakietu IMS/SCORM.

i dodatkowych plików. Manifest jest to plik w języku XML składający się z następujących elementów:

- metadane – dane opisujące moduł,
- organizacja – zestaw jednej lub kilku map strukturalnych, pokazujących układ zależności poszczególnych materiałów źródłowych wchodzących w skład kursu,
- materiały źródłowe – odseparowane zestawy treści dydaktycznych, które mogą być wykorzystywane wielokrotnie z uwagi na możliwość referencji różnych map strukturalnych do pojedynczego materiału,

- zagnieżdżone manifesty – wykorzystywane do opisu mniejszych modułów będących jego częścią.

SCORM szeroko wykorzystuje standard XML. W celu dogodnego korzystania z jego możliwości powstały edytory przygotowujące materiały w standardzie HTML i XML. Godnym polecenia jest produkt o nazwie eXe (eLearning XHTML editor) [19] rozpowszechniany w ramach wolnego oprogramowania (*Open Source*) zalecany do stosowania na platformach e-learningowych.

Projekt eXe jest rozwijany pod patronatem rządu Nowej Zelandii na uniwersytetach University of Auckland oraz Auckland University of Technology. Materiały edytowane poprzez eXe są zgodne z ogólnym standardem metadanych, przyjętym przez ISO – Dublin Core (Dublin Core Metadata Element Set, DC, DCES) [25]. Standard ten wspiera szybkie i łatwe przeszukiwanie repozytoriów w celu znalezienia istotnych informacji.

Według opinii środowisk badawczych, SCORM jest uznawany za najważniejszy standard w dziedzinie e-learningu, pomimo wielu poważnych argumentów podnoszonych przez różnych autorów kwestionujących odpowiedniość tego formatu dla nauczania [57]. W efekcie, w niniejszej pracy zdecydowano się na użycie SCORM jako standardu rozpowszechniania obiektów wiedzy z wykorzystaniem technologii P2P.

Rozdział 3

Kooperatywne nauczanie oparte na technologiach P2P i VoIP

Podczas prowadzonych badań przewodnią ideą było użycie technologii P2P jako interesującego i powszechnego sposobu dystrybucji obiektów wiedzy, który pozwoli na tworzenie grup kooperatywnego nauczania. Sieci P2P mogą ułatwić dostęp do źródeł edukacyjnych bez dodatkowego udziału zewnętrznych serwerów sieciowych, które często wymagają zintegrowanego zarządzania i nakładów serwisowych. Podobnie jak w przypadku plików muzycznych lub video, obiekty wiedzy mogą być łatwo rozpowszechniane wśród zainteresowanych i uczących się. Sieci P2P działają autonomicznie i spontanicznie pomiędzy nieograniczoną liczbą osób z minimalnym zaangażowaniem odgórnego zarządzania. Zaletą też jest fakt, iż użytkownicy mogą używać systemu P2P praktycznie bez żadnego wstępnego przygotowania.

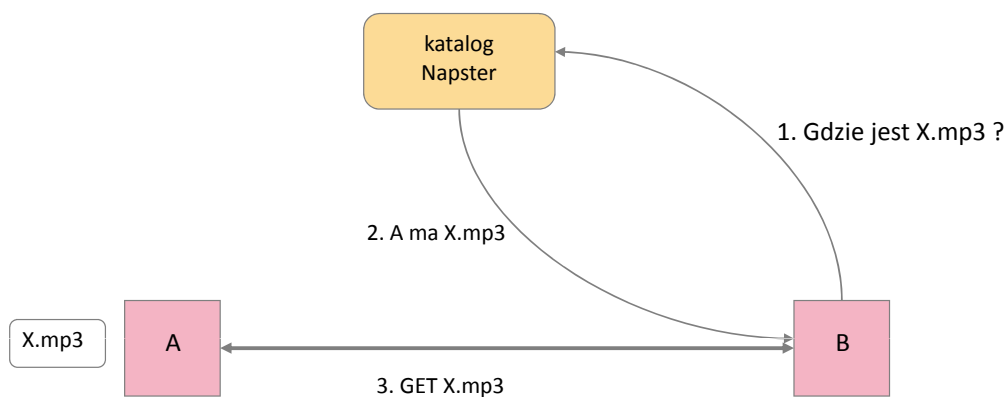
3.1 Podstawy technik P2P

P2P to model komunikacji dający wszystkim użytkownikom dostęp do informacji na zasadzie równości statusu każdego uczestnika. P2P jest obecnie najpopularniejszym sposobem wymiany plików w Internecie. Każdy zalogowany komputer tworzy tzw. „węzeł” sieci odgrywając rolę serwera jak

i klienta, pobierając dane z innych węzłów sieci. Zalogowani w sieci P2P uczestnicy wymiany danych każdorazowo tworzą nową strukturę połączeń.

3.2 System Napster

Pierwszym przykładem wykorzystania na szeroką skalę P2P był system Napster. Istotną przesłanką jego utworzenia był duży popyt na wymianę plików muzycznych w sieci. Napster - jako pierwsza aplikacja umożliwiał właścicielom komputerów udostępnienie wybranych zasobów swojego domowego PC innym użytkownikom. Opierał się na scentralizowanym katalogu plików, poprzez który można było znaleźć poszukiwane dane i skontaktować się z komputerem, który je udostępniał (nie był więc czystym przykładem modelu P2P).

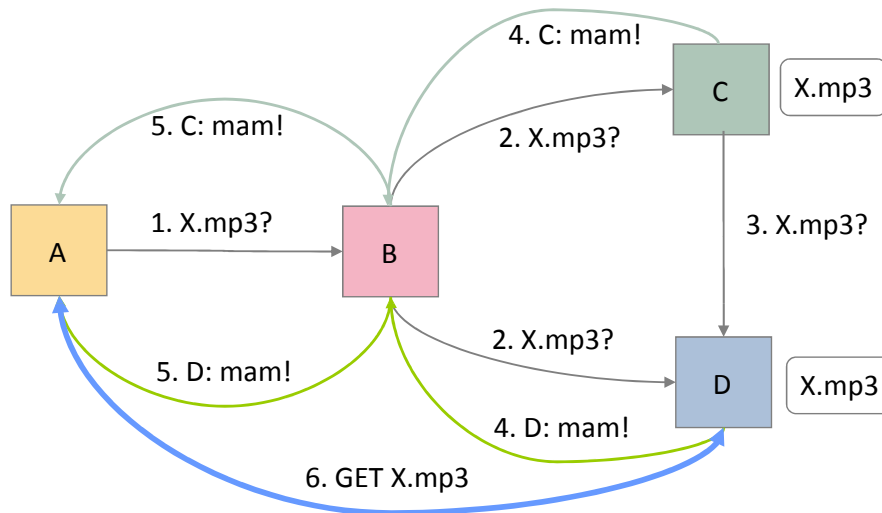


Rysunek 3.1: Zasada działania systemu Napster.

Zasada funkcjonowania Napstera zilustrowana jest na Rysunku 3.1: komputer B zadaje pytanie chcąc znaleźć plik X.mp3 (operacja 1), dowiadyuje się, że jest on na komputerze A (operacja 2) i kontaktuje się z nim, żeby go otrzymać (operacja 3). Wadą systemu Napster jest scentralizowana architektura, w której trzeba odwoływać się do katalogu.

3.3 Sieć Gnutella

System Gnutella proponuje rozproszoną strukturę poszukiwania opartą na propagowaniu zapytań w sieci, tak aby znaleźć poszukiwany plik. Rysunek 3.2 przedstawia proces poszukiwania.

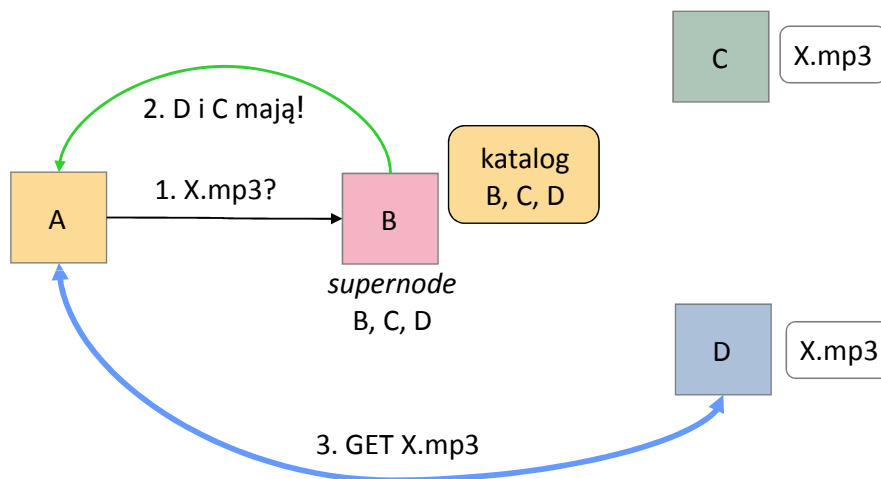


Rysunek 3.2: Zasada działania sieci Gnutella.

Zapytanie użytkownika kierowane jest do systemów znanych każdemu komputerowi (jego “peers”, operacje 1-3). Systemy, które mają dany plik odpowiadają a odpowiedź wraca poprzez ścieżkę odwrotną do tego, który wysłał zapytanie (operacje 4-5). Plik jest ostatecznie ściągany z wybranego systemu (operacja 6). Dostęp ten może być również przyspieszony, poprzez równoległe ściąganie różnych części pliku z kilku lub kilkunastu serwerów. Operacje 1-3 polegające na propagowaniu zapytania do wszystkich peerów w sieci są znane jako *flooding*, który implikuje wysokie zużycie zasobów każdego węzła i sieci. Skutkiem tego jest wydłużenie czasu wyszukiwania i zwiększenie kosztów. Użycie tej operacji stanowi wadę omawianego rozwiązania.

3.4 Sieć FastTrack (Kazaa)

Rozpowszechnianie zapytań w sieciach z dużą ilością peerów może być kosztowne i czasochłonne. System FastTrack w trakcie poszukiwań kieruje zapytanie do węzłów o stosunkowo większych możliwościach obliczeniowych. Biblioteka FastTrack, na której oparte są popularne systemy P2P (Kazaa, Grokster), wykorzystuje rozproszone tablice haszujące, które odwzorowują unikalny identyfikator pliku do jego lokalizacji. Niektóre węzły w takiej sieci mają rolę specjalną – są superwęzłami (supernodes), które przechowują część tablicy haszującej w lokalnym katalogu. Działanie sieci ilustruje Rysunek 3.3.



Rysunek 3.3: Zasada działania systemu FastTrack (Kazaa)

Węzeł B posiada informację o plikach przechowywanych w systemach B, C i D. Zapytanie o dany plik kierowane jest do pierwszego znanego superwęzła, który przegląda swoją tablicę i przesyła je do innych superwęzłów (operacja rozpowszechniania zapytania podobna jest do mechanizmu Gnutelli, jednak ograniczona jest tylko do superwęzłów). Jeżeli plik figuruje w katalogu, superwęzeł odpowiada pozytywnie (operacja 2), co pozwala na ściągnięcie pliku (operacja 3).

Można stwierdzić, że dotychczasowe funkcje systemów P2P obejmują głównie poszukiwania i rozpowszechniania plików audio lub video. Oprócz samego przesyłu danych interesującym może być dostęp do informacji o innych użytkownikach pobierających te same pliki i umożliwienie tworzenia kooperacyjnych grup zainteresowań.

3.5 P2P a e-learning

Możliwości rozwoju e-learningu poprzez technologie P2P omówiono w Rozdziale 1.2. Są one uwarunkowane przez zdefiniowanie nowego typu obiektów nauczania dostępnych poprzez zmodyfikowane technologie P2P oraz rozwiązanie problemu precyzyjnej indeksacji obiektów w asocjacji z metadanymi. Również stworzenie programistycznych narzędzi umożliwiających kontakt między uczestnikami grup tematycznych zainteresowań i ich kooperacyjnej współpracy będzie stanowiło rozszerzenie oferowanych dotychczas możliwości edukacyjnych. Prace w tym kierunku powinny wykorzystywać istniejące formaty platform e-learningowych takich jak SCORM i IMS.

Wkrótce po powstaniu pierwszych systemów P2P przeznaczonych głównie do wymiany plików muzycznych, kilka znaczących projektów zaproponowało wykorzystanie sieci P2P do tworzenia rozbudowanej struktury e-learningowej.

3.5.1 P2P a PKM – Personal Knowledge Management

Berman i Annexstein [9] proponują nowe narzędzie edukacyjne wykorzystujące sieci P2P. Połączenie P2P z PKM (Personal Knowledge Management) to sposób zarządzania zasobami wiedzy oraz nowy model struktury edukacyjnej. Istotnym jest wkład tego typu systemów w rozwój dostępnych narzędzi w 21 stuleciu określanym przez niektórych jako wiek informatyzacji. Podstawowymi zadaniami systemu byłyby:

- poszukiwanie i pozyskiwanie informacji,

- rozbudowywanie zasobów informacyjnych,
- organizacja informacji,
- analizowanie danych,
- prezentowanie danych,
- opracowywanie danych,
- tworzenie grup kooperacyjnych.

W swoich rozważaniach autorzy zakładają, że indywidualni użytkownicy dobrowolnie udostępniają swoje zasoby poprzez łącza internetowe. Jeżeli skonstruowana sieć obejmie dużą grupę wolontariuszy, stanowić ona będzie obszerne źródło informacji wewnątrz utworzonej struktury, niezależnej od jej indywidualnych uczestników (awaryjność, brak chęci udostępniania dokumentów w danej chwili). Autorzy wskazują, iż popularność P2P w technologiach IT czyni z niej w perspektywie bezpieczne jak również legalne źródło informacji.

Tworzenie wspólnot uczestników wymiany danych w połączeniu z ich zarządzaniem przez PKM stanowić będzie nowy model kooperacyjnej edukacji. Zaletą takiego podejścia jest możliwość propagowanie poprzez sieć P2P olbrzymich ilości materiałów dydaktycznych lokalizowanych w komputerach udostępniających, pobieranych ze zwielokrotnionych zasobów peerów i seedów. Możliwe jest tworzenie grup kooperacyjnych wokół prowadzących zajęcia oraz spośród społeczności uczestników – *classroom community*, *instructor community*, *student community*. Omawiany projekt dostarcza również narzędzi do wymiany materiałów naukowych przez prowadzących badania w ramach tworzonych grup zainteresowań – *scientific research community*.

Wyszukiwanie materiałów w tym rozwiązaniu opiera się na przeszukiwaniu sieci WWW poprzez znajomość na przykład nazwy artykułu, tytułu, autora, słów kluczowych, matadanych. Twórcy systemu wspierają metody wyszukiwania właściwych materiałów poprzez zastosowanie inteligentnych agentów.

Omawiany artykuł, pochodzący 2003 roku, był pierwszym, który sugerował wykorzystanie technologii P2P do nauczania. Pozostał on jednak jedynie w formie propozycji. Autorzy nie kontynuowali prac nad implementacją i walidacją tej koncepcji.

3.5.2 Helpmate

Podobnym rozwiązaniem jest system Helpmate [17] bazujący na technologii P2P. Zaprojektowany został do wymiany dokumentów zlokalizowanych na stronach WWW i umożliwiający ich modyfikację w czasie rzeczywistym. Deklarując chęć udziału w dyskusji dwóch lub więcej interlokutorów loguje się na stronie projektu co umożliwia im:

- wspólną pracę nad dokumentem i dokonywanie w nim zmian w czasie rzeczywistym,
- komunikację między nimi przy pomocy czatu (*chat room*),
- otrzymania opinii kooperantów zainteresowanych tematyką,
- połączenie audio i video (*web camera*),
- bieżący dostęp do oprogramowania w celu wprowadzania i weryfikacji zmian,
- korzystania z tłumaczeń dyskusji na swój język ojczysty.

Projekt jest kolejną próbą dostarczenia narzędzi bazujących na technologii P2P do ponad geograficznej swobodnej wymiany wiedzy w środowiskach naukowych, studenckich i wszelkich innych kooperacyjnych grup zainteresowań.

3.5.3 Edutella

Zawartość złożonych obiektów nauczania powinna być odpowiednio indeksowana dla łatwego ich wyszukiwania. Jedną z pierwszych prób w tym kierunku była Edutella [39]. System ten wykorzystuje standardy metadanych

zdefiniowane dla stron WWW oparte na specyfikacji RDF (Resource Description Framework) [53]. Edutella zapewnia komunikację bez pośrednictwa serwerów centralnych używając protokołu komunikacyjnego JXTA [29]. Była ona uzupełnieniem procesu nauczania w systemie e-learningu i prezentowała nowatorskie podejście do systemu komunikowania się uczestników procesu kształcenia.

Dążeniem twórców Edutelli było stworzenie zdecentralizowanej infrastruktury e-learningowej opisującej jej zawartość przy pomocy metadanych i indeksacji. Podczas wyszukiwania informacji w sieci WWW opis zasobów przy pomocy metadanych jest niezmiernie istotny jednakże ten sam opis w przypadku sieci P2P stanowi krytyczny problem późniejszego dostępu do nich.

Edutella wykorzystuje podstawowe zasady sieci JXTA, tworzy własne reguły wyszukiwania danych opierając się na trójczłonowym opisie tj. *zasobach* (np. strony WWW, dokumenty XML), *właściwościach* (autor, tytuł), *instrukcjach* (kombinacji zasobu, właściwości i wartości). Wśród pakietów tworzących projekt można znaleźć te, które odpowiadają za wspólny model danych, import zapytań, jak i usługi konsumenckie. W wersji podstawowej projektu Edutella jako języka zapytań do wyszukiwania informacji użyto Datalogu.

Najistotniejsze pakiety systemu to:

- moduł odpowiadający za system zapytań w formacie RDF w wewnętrznej bazie danych,
- moduł odpowiadający za system edytowania zapytań i ich zamiany na format RDF.

Wyszukiwanie i przygotowanych informacji przebiega przy udziale szeroko rozbudowanej biblioteki zapytań Edutella Query Language (EQL). Edutella działała na bazie Gnutelli, w której występował problem nadmierowości generowanych zapytań na skutek niekontrolowanego mnożenia się pakietów – *flooding*. Również zbiór otwartych protokołów JXTA nie uzyskał powodzenia wśród użytkowników sieci P2P. Edutella nie zastosowała ontologii do opisu repozytoriów. Powyższe czynniki spowodowały brak szerokiego rozpowszechnienia Edutelli.

3.5.4 PADLR – Personalized Access to Distributed Learning Repositories

PADLR (Personalized Access to Distributed Learning Repositories) [52] to modułowy system e-learningowy oparty na strukturze sieci Web wspierający edukację uniwersytecką. Korzysta z infrastruktury opisanej wcześniej Edutelli. Zrealizowany był jako *open source* na bazie międzynarodowych standardów wspieranych przez konsorcjum W3C. Wyszukiwanie realizowane jest przez dokładny system z wykorzystaniem inteligentnych agentów.

Zapis dokumentów bazuje na składni XML z wykorzystaniem opisu metadanych przy pomocy RDF. PADLR dostarcza narzędzi umożliwiających tworzenie nowej infrastruktury wspomagającej edukację na różnych poziomach kształcenia uniwersyteckiego. PADLR stwarza możliwości wyrażania swych opinii na temat opracowanych danych źródłowych. Projekt Couresware Watchdog stanowi część projektu PADLR [48].

3.5.5 Kierunki rozwoju istniejących rozwiązań

Wspomniane projekty realizowane zostały w uznanych ośrodkach uniwersyteckich zajmujących się m.in. propagowaniem materiałów edukacyjnych poprzez sieci P2P. W najstarszych z nich, ograniczeniem jest koncentracja na przygotowaniu repozytoriów. Tworzenie systemów zapytań i poszukiwań według kluczy nie sprostą rozwojowi nie tylko w tej dziedzinie. Późniejsze

projekty skłaniają się ku precyzyjnemu opisowi treści dydaktycznych. Ontologiczna klasyfikacja zasobów skutkuje zwiększeniem ich dostępności w sieci WWW. E-learning stanowi część ogólnych zasobów sieciowych podlegającym problemowi zarządzania wiedzą. Możliwa jest pełna organizacja materiałów dydaktycznych w obrębie danej platformy e-learningowej zgodnie z wymaganiami ich twórców i użytkowników, jednak w całej sieci problem ten musi uzyskać rozwiązanie strukturalne.

Koniecznym jest tworzenie i wyszukiwanie materiałów z uwzględnieniem ich struktury semantycznej. Znaczącym jest zastosowanie do materiałów edukacyjnych systemów LOM oraz opisów bazujących na Dublin Core. Ontologie ułatwiają dostęp, pozyskiwanie, interakcję i zarządzanie wiedzą. Omówione istotne projekty wzbogacające platformy e-learningowe o technologie P2P wskazują kierunek dalszej ich rozbudowy.

Przedstawione projekty należy rozwijać w kierunku możliwości wzajemnego kontaktu między uczestnikami procesu edukacyjnego, tworzenia grup zainteresowań na dowolnym poziomie zaawansowania tematycznego i wykorzystania ich do kooperacji w nauczaniu. Komunikacja w powstających ad hoc grupach tematycznych również wpłynie na atrakcyjność takiej formy współpracy. Powszechną dostępność może zapewnić bazowanie na oprogramowaniu i projektach z grupy *open source*.

3.6 Wybór typu sieci P2P

W dotychczas omówionych w pracy technologiach P2P (Napster, Gnutella, FastTrack) można wskazać następujące ich ograniczenia [47]:

- Niska przepustowość sieci. Prędkość pobierania przez wszystkich klientów uzależniona jest od prędkości komputera udostępniającego plik. Zwiększenie liczby klientów pobierających ogranicza prędkość pobierania danych.

- Większość klientów tylko pobiera dane w minimalnym stopniu udostępniając swoje zasoby.
- Statyczny charakter zasobów sieci. Udostępniane pliki są często bardzo obszerne. Kolejne ich udostępnienie może następować po ich pobraniu. W efekcie, w trakcie ich częstokroć wielogodzinnego pobierania nie są one udostępniane innym chętnym.

Celem analizy było rozpatrzenie technologii P2P i wybranie takich, które umożliwiają realizację wcześniej wskazanych wymagań. Rezultatem był wybór sieci BitTorrent z racji wielu interesujących właściwości. Detale systemu BitTorrent przedstawione są w następnym rozdziale.

Rozdział 4

Sieć P2P BitTorrent

Przedstawione niedogodności w sieciach P2P dotychczas omawianych, BitTorrent [10] eliminuje w następujący sposób:

- Udostępniane pliki są rozdzielone pomiędzy wszystkie węzły, które są podłączone do sieci. Dostęp do scentralizowanego serwera konieczny jest tylko w celu zlokalizowania fragmentów pliku (wersja *trackerless* pozbywa się tego krytycznego serwera).
- Sieć nie jest zarządzana odgórnie w przeciwieństwie do Napstera.
- Każdy węzeł sieci, pobierając plik, jednocześnie go udostępnia. W efekcie zwiększenie liczby klientów pobierających plik powoduje zwiększenie prędkości pobierania przez wszystkich klientów.
- Dzięki temu, że pobierane pliki są automatycznie udostępniane innym węzłom, w sieciach BitTorrent nie ma klientów, którzy tylko je pobierają (*free riding*).

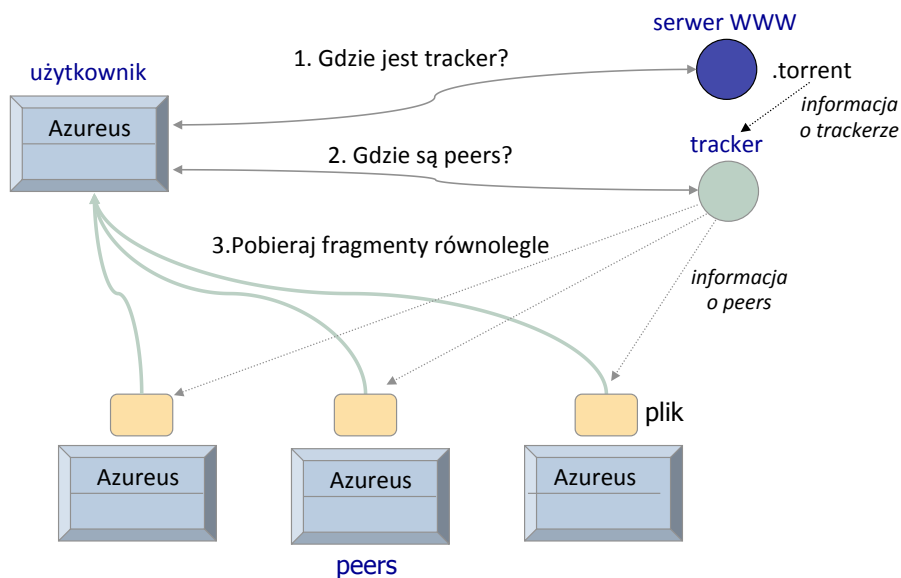
Dla zrozumienia działania sieci BitTorrent niezbędne jest zdefiniowanie podstawowych pojęć:

- *Peer*: węzeł pobierający lub udostępniający plik, uczestnik wymiany.
- *Seeder*: węzeł posiadający kompletny plik, który udostępnia go innym węzłom.

- *Leecher*: węzeł, który w danym momencie pobiera dany plik i udostępnia posiadane jego fragmenty. Nie posiada całego pliku. Po całym pobraniu stanie się *Seederem*.
- *Tracker*: serwer śledzenia oferujący informacje (głównie adresy IP) o innych węzłach posiadających lub pobierających dany plik.
- **.torrent**: metaplik śledzenia zawierający niezbędne informacje do rozpoczęcia pobierania pliku.

Wyróżnia się dwa podstawowe tryby działania tej sieci:

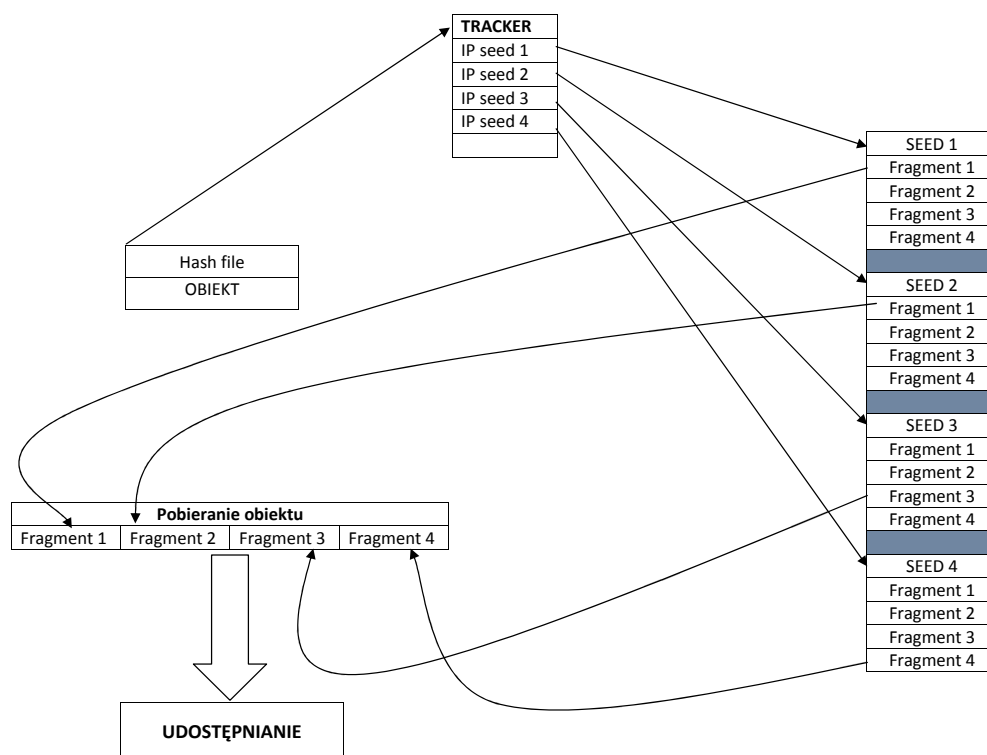
- tryb standardowy ze scentralizowanym trackerem, przechowującym informacje o węzłach posiadających fragmenty (ich adresy IP),
- tryb bez trackera – *trackerless*, informacja o węzłach posiadających fragmenty jest przechowywana w sposób całkowicie rozproszony w strukturze DHT Kademlia [34].



Rysunek 4.1: Zasada działania sieci BitTorrent z trackerem.

4.1 Sieć BitTorrent z trackerem

Rysunek 4.1 przedstawia zasadę działania sieci BitTorrent z trackerem. Użytkownik szuka metapliku `.torrent` na serwerach WWW, który zawiera informacje o fragmentach rozprowadzanego pliku i adres trackera. Następnie węzeł użytkownika, który chce rozpocząć pobieranie pliku kontaktuje się z trackerem i otrzymuje informacje o węzłach posiadających fragmenty. Ostatecznie węzeł wysyła do nich prośbę o wysłanie fragmentów, których transmisje przebiegają równoległe. Pobierający węzeł wysyła jednocześnie całkowicie pobrane fragmenty do innych węzłów, które chcą również pobrać dany plik.



Rysunek 4.2: Pobieranie pliku w sieci BitTorrent.

Rysunek 4.2 daje inne spojrzenie na pobieranie pliku: plik jest identyfikowany w sposób unikalny w metapliku `.torrent`, który dostarcza adres trackera. Kontakt z trackerem pozwala na odszukanie węzłów – Peers, które wysyłają fragmenty.

Metaplik `.torrent` zawiera następujące informacje o pliku lub plikach (BitTorrent pozwala na propagację archiwum `.zip` z wieloma plikami):

- **announce**: URL trackera,
- **info**: dane o pliku i jego fragmentach:
 - **files**: lista pliku lub plików
 - * **length**: rozmiar pliku w bajtach,
 - * **path**: nazwa, ścieżka pliku,
 - **name**: nazwa pliku,
 - **piece length**: rozmiar fragmentów w bajtach,
 - **pieces**: konkatencja identyfikatorów fragmentów, każdy fragment jest identyfikowany 20-bajtową wartością otrzymaną poprzez obliczenie haszu SHA1 (Secure Hash Algorithm) na zawartości fragmentu,

Udostępniane pliki w sieci BitTorrent składają się z fragmentów o wielkości od kilkuset KB do kilkuset MB (typowy rozmiar przesyłanego fragmentu to obecnie 256 KB). Klient kontaktuje się z trackerem używając protokołu HTTP lub HTTPS – wysyła komendę GET z następującymi głównymi atrybutami:

- **info_hash**: identyfikator pliku
- **peer_id**: identyfikator węzła pobierającego
- **ip**: adres IP lub nazwa DNS węzła pobierającego
- **port**: port używany przez węzeł pobierający

Identyfikatorem pliku jest 20-bajtowa (160 bitów) wartość otrzymana poprzez obliczenie haszu SHA1 na części **info** metapliku **.torrent**:

$$key = \text{SHA1}(\text{info}) \quad (4.1)$$

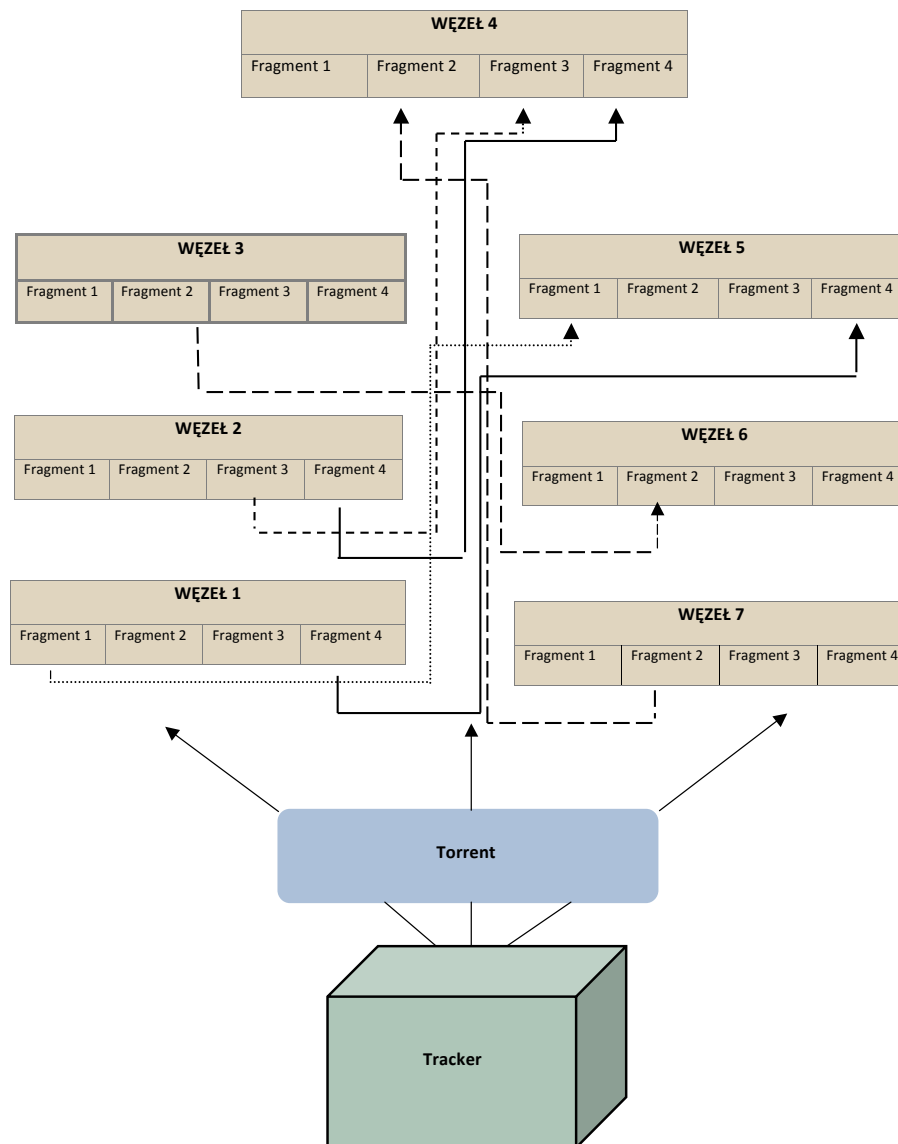
Tracker podaje listę Peers z informacją o każdym węźle:

- **peer_id**: identyfikator Peer
- **ip**: adres IP lub nazwa DNS Peer
- **port**: port używany przez Peer

Klient otrzymuje od trackera listę pięćdziesięciu Peerów posiadających dany plik. Nawiązuje kontakt z 30-40 Peerami przesyłając **info_hash** i swój **peer_id**. Każdy Peer odpowiada, przesyłając **bitfield**, tablicę binarną, która informuje o fragmentach posiadanych przez Peer i pozwala wysłać prośbę o ich przesłanie.

Po pobraniu pierwszego fragmentu zostaje on udostępniony innym użytkownikom, co pozwala na maksymalizację liczby węzłów rozprowadzających dany fragment. Każdy Peer udostępnia pliki czterem najszybszym pobierającym, a każdy pobierający – Leecher może udostępniać pobrane fragmenty kolejnym czterem chętnym. Prędkość transmisji jest sprawdzana co 10 sekund. Jeżeli jest ona niezadowalająca wówczas eliminowany jest najwolniejszy z czterech transferów, a w jego miejsce wznawiany jest transfer od innego Peera.

Rysunek 4.3 przedstawia przykład pobierania: węzeł 4 otrzymuje fragment 2 od węzła 7, a fragmenty 3 i 4 od węzła 2. Transmisje fragmentów pomiędzy węzłami sieci przebiegają równocześnie. Jeżeli transmisja ulega przerwaniu, zostaje ona wznowiona od ostatniego, w całości pobranego fragmentu.



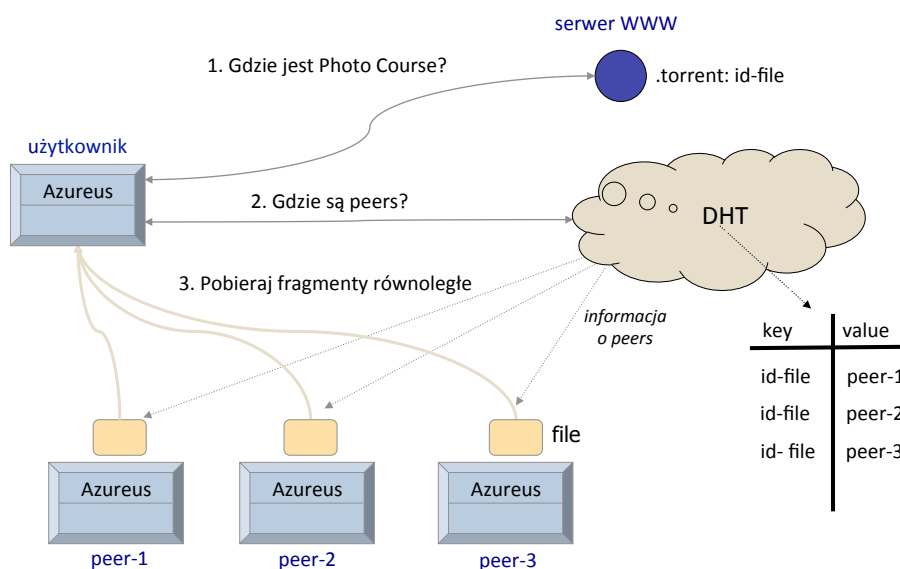
Rysunek 4.3: Przykład pobierania w sieci BitTorrent.

4.2 BitTorrent w wersji trackerless

Analizując pierwsze projekty sieci P2P sprzed dziesięciu lat dostrzec można ich centralizację. Z czasem mamy do czynienia z sieciami coraz

bardziej zdecentralizowanymi. Wadą BitTorrenta w wersji z trackerem jest uzależnienie sieci od jego działania. Trackery mogą ulegać awariom lub być całkowicie likwidowane. Dlatego twórcy BitTorrenta opracowali wersję bez trackera – trackerless BitTorrent.

Założeniem wersji *trackerless* jest to, iż każdy klient staje się jednocześnie mini-trackerem przechowując (w ramach pobierania) informacje o innych Peerach, udostępniających dany plik. Klient, poszukując informacji o Peerach, zwraca się o nią do innego Peera, a jeżeli on jej nie posiada to przesyła zapytanie do kolejnego uczestnika wymiany będącego na jego liście.



Rysunek 4.4: Działanie sieci BitTorrent bez trackera.

Pobieranie pliku z sieci polega na znalezieniu najpierw metapliku *.torrent* i przeszukaniu DHT, aby otrzymać listę Peers (cf. Rys.4.4). Poszukiwanie w DHT wymaga posiadania informacji o co najmniej jednym węźle podłączonym do sieci DHT (jego adres IP oraz numer portu). Jeżeli klient go nie posiada, zwraca się do specjalnego serwera z poleceniem Bootstrap w celu otrzymania adresu węzła i podłączenia się do sieci.

Poszukiwanie w DHT zwraca adresy węzłów posiadających plik lub jego fragmenty i klient może rozpocząć równoległe pobieranie.

W porównaniu z Rysunkiem 4.1, na Rysunku 4.4 przedstawiającym działanie sieci *trackerless*, w miejsce trackera pojawił się obiekt DHT, który realizuje w sposób rozproszony tabelę o kolumnach *key* i *value*. W tym użyciu DHT, identyfikator pliku jest kluczem, a odszukanymi wartościami są adresy Peerów, które posiadają fragmenty pliku.

4.2.1 Zasady działania tablic mieszających DHT

Koncepcja pracy bez trackera w BitTorrentcie polega na rozproszonych tablicach mieszających DHT. Sieć DHT składa się z pewnej liczby węzłów dzielących między sobą zbiór kluczy. Istotą działania DHT jest fakt, że węzły posiadają identyfikatory z przestrzeni kluczy, tzn. jeżeli kluczem jest 20 bajtowa wartość to węzły mają również 20 bajtowy identyfikator. Para (*key*, *value*) zapisywana jest na węźle o identyfikatorze *key* lub blisko niego w sensie pewnej metryki. Węzeł poszukujący dany klucz *key* wysyła zapytanie do sieci DHT, które przesyłane jest przez inne węzły coraz bliżej węzła, który zapisał parę (*key*, *value*) i może odpowiedzieć podając *value*.

BitTorrent tak jak inne systemy bazujące na DHT wykorzystuje zestaw czterech podstawowych funkcji:

- *Ping*: kontakt z innym Peerem, który wprowadza adres do tablic znanych Peerów. Udostępnia adres danego węzła innym Peerom.
- *Lookup(key)*: znajdź węzeł, którego adres jest bliski kluczowi i który zwróci *value*.
- *Store(key, value)*: zapisz dane na stacji bliskiej kluczowi.
- *Get(key)*: pobierz dane *value* przypisane do klucza.

Wielokrotnie zapisane wartości *value* mogą być związane z jednym kluczem. W takim wypadku, funkcja *Get(key)* zwraca zapisane wielokrotnie wartości.

4.2.2 Kademlia – implementacja DHT

Kademlia [34] jest implementacją DHT wykorzystywaną w różnych sieciach P2P w szczególności w sieci BitTorrent w wersji *trackerless*. Celem Kademlii jest oferowanie szybkiego dostępu do par $(key, value)$, pomimo znacznej ilości węzłów, które mogą często odłączać się od sieci.

Sieć Kademlia składa się z dużej ilości węzłów wymieniających dane i przechowujących informacje wzajemnie o sobie. Każdy węzeł posiada następujące elementy

- identyfikator: $nodeID$
- adres IP: $IPaddress$
- port UDP: $UDPport$

Identyfikatory $nodeID$ i klucze key zapisane są na B bitach. Wartość $B = 160$ odpowiada 20 bajtowym identyfikatorom plików, które używane są przez BitTorrent.

Parametrami Kademlii są:

- α : stała określająca ilość równoległych połączeń w sieci, standardowo 3,
- k : maksymalna liczba kontaktów przechowywana w jednym pojemniku (*bucket*).

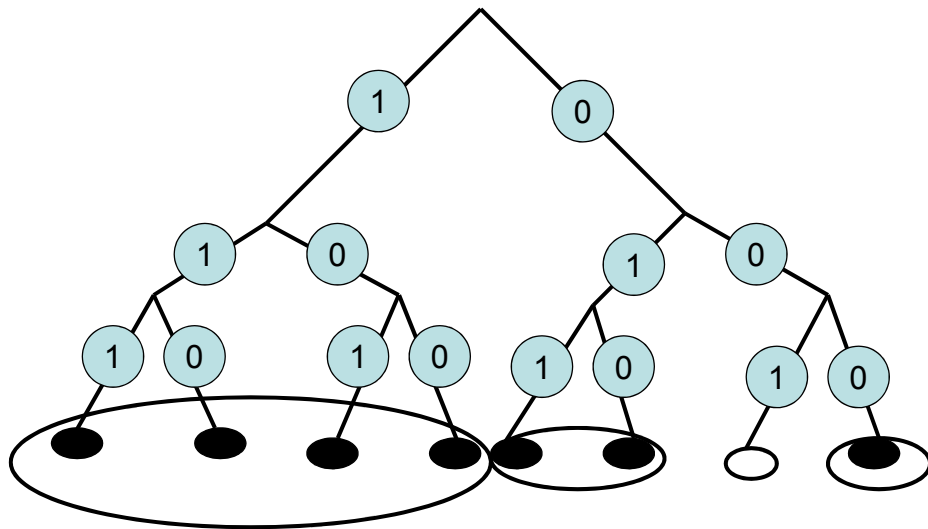
Para $(key, value)$ zapisywana jest na węzłach, których $nodeID$ jest najbliższy kluczowi key . Również poszukiwanie wartości $value$ polega na znalezieniu węzłów bliskich kluczowi. Odległość XOR pomiędzy dwoma identyfikatorami x i y (lub pomiędzy kluczem a identyfikatorem) zdefiniowana jest jako:

$$x \oplus y = x_i \oplus y_i, \quad (4.2)$$

gdzie $\oplus$ jest operacją XOR (exclusive OR), a rezultat interpretowany jest jako wartość całkowita. Na przykład dla $B = 7$, jeżeli $x = (1011011)$

a $y = (1010001)$, odległość $x \oplus y = 10$. Odległość posiada również następującą własność: odległość pomiędzy x i y jest co najmniej 2 do potęgi odwrotnej do rozmiaru wspólnego prefiksu bitowego. Dla poprzedniego przykładu, wspólnym prefiksem bitowym jest $x = (101)$ o rozmiarze 3, więc odległość jest równa $x \oplus y \geq 2^{7-1-3} = 8$.

Kademlia traktuje każdy węzeł jako liść w drzewie binarnym o pozycji wskazanej przez unikatowy prefiks identyfikatora. Rysunek 4.5 pokazuje węzeł 001 w przykładowym drzewie dla $B = 3$. Dany węzeł dzieli drzewo na sekwencje stopniowo niższych poddrzew, które go nie zawierają. Najwyższe poddrzewo zawiera połowę drzewa (lewe poddrzewo o prefiksie 1 na rysunku). Następne poddrzewo zawiera połowę pozostałego poddrzewa nie zawierającego węzła, i tak dalej. (poddrzewo o prefiksie 01 zawierające 011 i 010 na rysunku). Węzły w tym samym poddrzewie są sobie bliższe niż węzły znajdujące się w innych poddrzewach co wynika bezpośrednio z własności funkcji XOR.



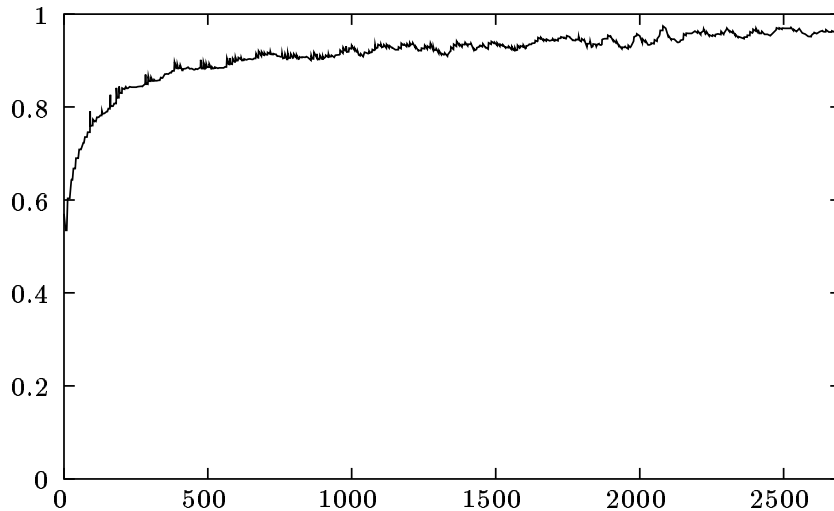
Rysunek 4.5: Struktura drzewiasta Kademlii.

Kademlia gwarantuje, że każdy węzeł zna co najmniej 1 węzeł w każdym poddrzewie, jeżeli to poddrzewo zawiera węzły. Własność ta służy do lokalizacji każdego węzła na podstawie jego *nodeID*.

Każdy węzeł zapamiętuje informacje o innych węzłach: dla każdego $0 \leq i \leq 160$ węzeł posiada k -pojemnik (k – *buckets*) zawierający k kontaktów – trójek

$$\langle IPaddress, nodeID, UDPport \rangle$$

reprezentujących węzły w odległości od 2^i do 2^{i+1} . Wewnątrz k -pojemnika kontakty są sortowane według czasu: najstarszy kontakt na samym początku, ostatni na końcu. Przyjęcie stosunkowo dużej wartości dla k daje małe prawdopodobieństwo, że wszystkie kontakty znikną. Kademlia przyjmuje wartość $k = 20$ na podstawie obserwacji czasu pozostania aktywnym w sieci Gnutella (cf. Rys. 4.6).



Rysunek 4.6: Prawdopodobieństwo pozostania kontaktu w sieci, przez kolejne 60 minut, w funkcji czasu przebywania w sieci w minutach. Źródło: [34]

Protokół Kademlii definiuje cztery procedury połączeniowe pomiędzy węzłami (RPCs – Remote Procedure Calls). Wysyłając wywołanie RPC,

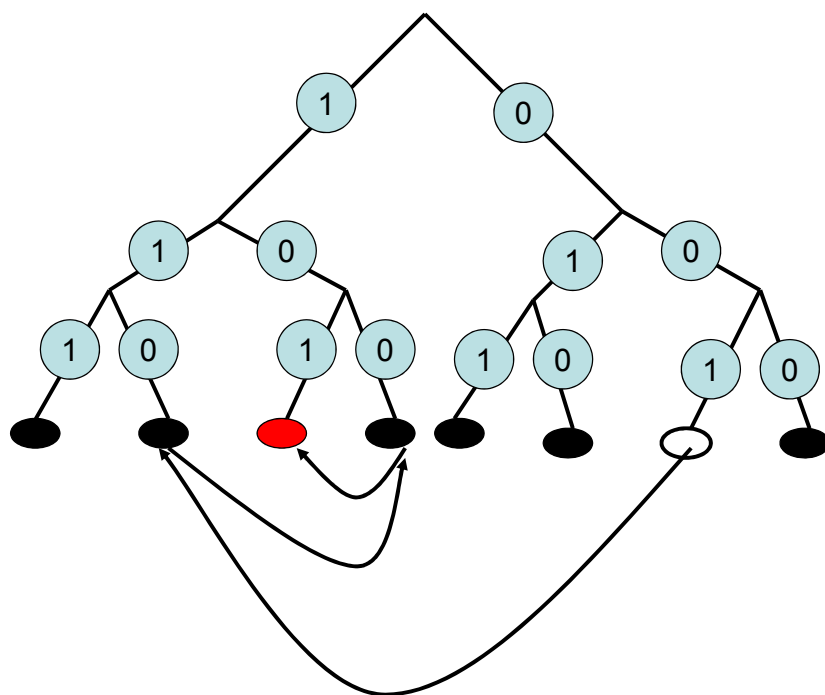
każdy węzeł dostarcza odbiorcy swoje wyżej wymienione trzy elementy charakterystyczne dla kontaktu $IPaddress$, $nodeID$, $UDPport$.

1. PING: test połączenia do potencjalnego odbiorcy. Jeżeli odbiorca komunikatu jest aktywny to odpowie komunikatem PONG. Nadawca i odbiorca umieszczają w swoich k -pojemnikach trójkę informacji o kontakcie $\langle IPaddress, nodeID, UDPport \rangle$.
2. STORE: zadanie zapamiętania pary $\langle key, value \rangle$ do późniejszego odczytu. Nadawca przesyła klucz key i blok danych $value$ (rozmiar wartości to 1024 bity). Para $\langle key, value \rangle$ przechowywana jest na $k = 20$ węzłach, których $nodeID$ jest najbliższy kluczowi.
3. FIND_NODE: poszukiwanie pary $\langle key, value \rangle$ o znanym kluczu. Odbiorca komunikatu zwraca znane mu trójki $\langle IPaddress, nodeID, UDPport \rangle$ k kontaktów najbliższych kluczowi.
4. FIND_VALUE: poszukiwanie wartości przyporządkowanej do klucza w ramach pary $\langle key, value \rangle$. Ta funkcja zachowuje się tak samo jak FIND_NODE z tą różnicą, że węzeł, który ją otrzymuje zwraca blok danych $value$, jeżeli go posiada.

Najważniejszą operacją Kademlii jest znalezienie węzłów bliskich danemu kluczowi X , co pozwala następnie na odczytanie wartości przyporządkowanej do danego klucza poprzez funkcję FIND_VALUE. Polega ona na rekurencyjnym wywoływaniu funkcji FIND_NODE(X), gdzie X jest kluczem. Inicjator poszukiwania wybiera α węzłów z pierwszego niepułstego k -pojemnika, który jest najbliższy X i wysyła do nich wywołanie FIND_NODE(X). Po otrzymaniu odpowiedzi, inicjator wysyła następnie to wywołanie do węzłów zwróconych w poprzednim kroku. Każda iteracja zbliża poszukiwanie do węzłów bliższych kluczowi i kończy się gdy

Aby wartości zapisane do DHT były dostępne w sposób ciągły, węzły muszą cyklicznie ponawiać ich zapis. W przeciwnym wypadku, procedura poszukiwania może zakończyć się niepowodzeniem z dwóch powodów:

- Strategia wybrana przez Kademię polega na ponawianiu zapisu klucza co godzinę.



Rysunek 4.7: Wyszukiwanie klucza w Kademlii.

Rysunek 4.7 ilustruje proces wyszukiwania klucza $X = 101$ dla $B = 3$ i $\alpha = 1$. Inicjator poszukiwania ma identyfikator 001 i posiada kontakt 110 do poddrzewa z prefiksem 1. Wyszukiwanie przebiega w sposób następujący:

- N_1 : 110
- N_2 : 100 = FIND_NODE(X)
- N_3 : 101 = FIND_NODE(X)

Kademlia jest protokołem komunikacyjnym powstałym głównie z myślą o wykorzystaniu w sieciach P2P. Rozwiązanie służyło uniezależnieniu się od centralnych serwerów. Podłączone do systemu stacje posiadają możliwość wymiany informacji. Do zalet Kademlii należą między innymi:

- Efektywność. Maksymalna liczba wywołań FIND_NODE wynosi $\log(n)$ w sieci z n węzłami.
- Odporność na awarię węzłów.
- Możliwość równoległego wyszukiwania.

Ponieważ wydajność Kademlii zależy od szybkiego wywoływania funkcji RPC na węzłach, dlatego RPC wykorzystuje zamiast TCP prostszy protokół bezpołączeniowy UDP, który posiada wystarczające do tego właściwości.

Rozdział 5

Wykorzystanie sieci P2P do propagacji obiektów nauczania i tworzenia kooperacyjnych grup zainteresowań

Jak stwierdzono na wstępie, celem niniejszej pracy jest przebadanie w jaki sposób można wykorzystać sieci P2P do rozpowszechniania obiektów nauczania i tworzenia kooperacyjnych grup zainteresowań. Wymaga to dostarczenia systemowych narzędzi do efektywnej indeksacji oraz poszukiwania obiektów przy użyciu ontologii. Obiekty znalezione w precyzyjny sposób powinny być następnie efektywnie przesyłane do uczących się, co pozwala na odnalezienie innych użytkowników zainteresowanych daną dziedziną. Proponowane obiekty nauczania mogą również dostarczyć informacji do bezpośredniej komunikacji pomiędzy członkami grupy za pomocą systemów VoIP, co prowadzi do zastąpienia tradycyjnej pracy samodzielnej poprzez kooperację w grupie.

Chcąc wykorzystać aktualnie działające sieci P2P do rozpowszechniania obiektów nauczania należało wziąć pod uwagę ich niedogodności:

- Specjalizacja do przesyłania plików multimedialnych. Sieci P2P głównie zajmują się transferem plików audio i video.
- Zbyt proste funkcje indeksacji i odszukiwania. Poszukiwanie plików odbywa się jedynie na podstawie znajomości ich nazwy.
- Brak komunikacji między systemami poza bezpośrednim przesyłem upublicznionych plików. Osoba pobierająca plik nie ma żadnej informacji o innych użytkownikach, które pobrały ten sam plik, ani nie ma możliwości wzbogacania zawartości pliku.
- Problemy własności intelektualnej. Nie zawsze wymiana repozytoriów odbywa się za zgodą ich twórców. Często dokonywana jest nawet wbrew zastrzeżonym przez ich autorów prawom. Powoduje to częste spory, a nawet procesy z właścicielami sieci, które upowszechniają takie materiały. Powszechna skala tego procederu zmusza twórców do czynienia ustępstw i sformalizowania zasad wymiany celem poszanowania ich praw autorskich¹.

W niniejszym rozdziale przedstawiono propozycje uzupełnienia sieci P2P tak aby osiągnąć zamierzone cele pracy.

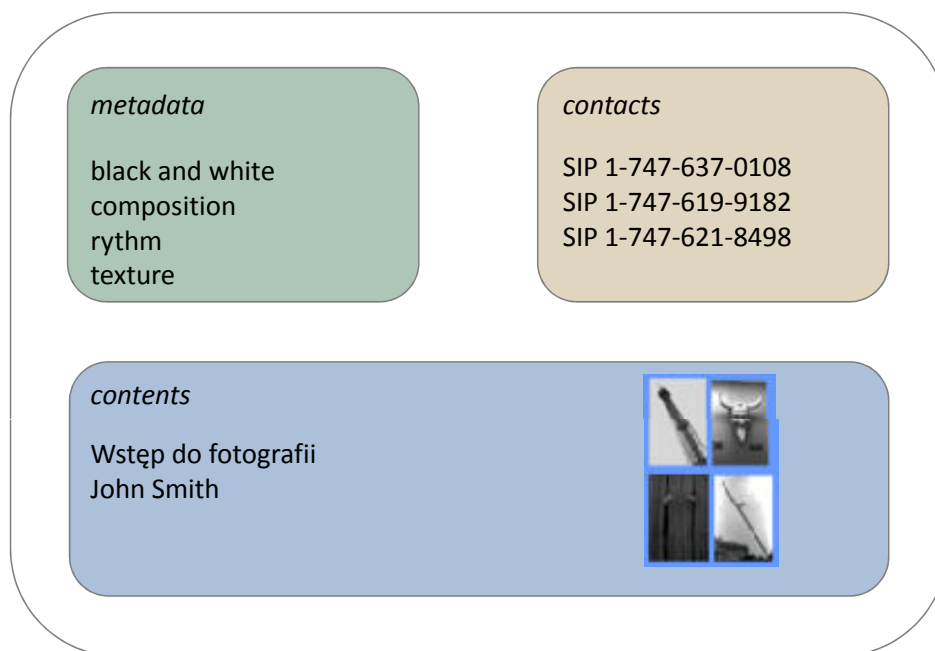
5.1 Koncepcja obiektu nauczania

Aby wykorzystać sieci P2P do rozpowszechniania obiektów nauczania i tworzenia kooperacyjnych grup zainteresowania, obiekty te powinny zawierać niezbędne informacje i posiadać pewną, ściśle określoną strukturę. Rysunek 5.1 ilustruje ideę konstrukcji złożonego obiektu nauczania.

Obiekt nauczania powinien zawierać trzy następujące typy informacji:

- stosowną zawartość dydaktyczną,

¹Niniejsza praca nie rozpatruje problemu nielegalnego rozpowszechniania materiałów nauczania i nie ma ambicji jego rozwiązania, a jedynie ogranicza się do aspektów technicznych. Główną ideą jest wykorzystanie wymiany plików, których autorzy są zainteresowani jak najszerszym dostępem do nich i zgadzają się na dokonywanie zmian oraz ulepszeń w rozpowszechnianych obiektach.



Rysunek 5.1: Przykładowa zawartość obiektu nauczania.

- metadane do precyzyjnej indeksacji i odszukania obiektu,
- informacje o sposobie kontaktowania się grupy umożliwiające proste nawiązanie łączności VoIP.

Podstawową częścią obiektu jest jego dydaktyczna zawartość złożona z materiałów różnego typu, takich jak strony WWW, dokumenty Word, slajdy PPT, nagrania audio, filmy video, jak również dokumenty multimedialne typu SMIL lub Flash. Zawartość ta jest usystematyzowana zgodnie ze standardem SCORM, co umożliwia wykorzystanie istniejących aplikacji prezentacji, które interpretują ten format.

Aby umożliwić precyzyjne wyszukiwanie obiektu użyto metadanych zawartych w części informacyjnej obiektu. Metadane opisują zawartość obiektu nauczania. Należą do nich informacje o autorze, tytuł, pojęcia z danej dziedziny wiedzy, opisy oraz inne dane zdefiniowane według

standardu Dublin Core [25] (Rysunek 5.1 pokazuje jedynie kilka wybranych elementów metadanych specyficznych dla wykładu o fotografii).

Asocjacja pomiędzy zawartością i dziedziną wiedzy wykorzystuje zasady *ontologii dziedzinowej*. Ontologia dziedzinowa jest siecią pomiędzy pojęciami, która definiuje elementy i relacje semantyczne między nimi [38, 21]. Wykorzystując ontologię dziedzinową autor obiektu nauczania może przypisać pojęcia do zawartości obiektu dla odzwierciedlenia jego semantyki. Jej zastosowanie pozwala na precyzyjne indeksowanie zawartości, nawet gdy dany obiekt nie zawiera reprezentacji tekstowej danego pojęcia. Na przykład, gdy wykład dotyczy Kamer Cyfrowych, autor może przypisać pojęcie Fotografia, mimo że nie występuje ono w tytule wykładu.

Ostatnim elementem obiektu jest lista kontaktów umożliwiających późniejsze tworzenie grup kooperacyjnego nauczania [45]. Lista ta zawiera kontakty lub identyfikatory serwisu VoIP autora, nauczających lub osób, które pobrały dany obiekt.

Systemy P2P wykorzystywane obecnie do rozpowszechniania filmów i muzyki są obiektami nie ulegającymi zmianom w czasie. W przypadku zdefiniowanej struktury obiektów nauczania, ich zawartość może się zmieniać z dwóch powodów:

- każdy nowy użytkownik, który pobiera materiał dodaje swój identyfikator umożliwiający kontakt poprzez VoIP
- zawartość materiałów może być modyfikowana poprzez uczestników kursu.

Ta właściwość obiektów nauczania narzuca istotne wymagania na aktualne systemy P2P, które obecnie nie są przystosowane do rozpowszechniania obiektów zmieniających się w czasie. Wymaga to więc pewnej specyficznej struktury, tak aby kurs mógł modyfikować się w czasie.

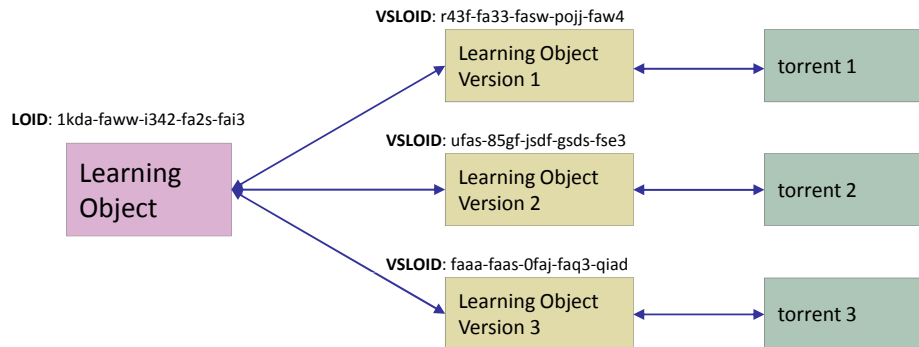
Proponowane rozwiązanie polega na zarządzaniu kolejnymi wersjami obiektu nauczania, tak że każda wersja jest niezmiennym plikiem z punktu widzenia systemu rozpowszechniania P2P. Jednocześnie kolekcja niezmiennych wersji jest widziana jako jeden modyfikowalny obiekt nauczania dzięki unikalnemu identyfikatorowi. Prowadzi to do definicji złożonego obiektu w postaci dwóch części:

- część niezmienna w czasie: obiekt nauczania i jego wersje. Każdy taki niezmienny obiekt może być rozpowszechniany poprzez standardową sieć P2P.
- część zmienna w czasie: informacja o związkach pomiędzy obiektem nauczania i jego wersjami, jak również lista kontaktów pozwalająca na tworzenie grup kooperacyjnego nauczania. Informacje te powinny być zapisywane w inny sposób niż standardowa sieć P2P.

Oznaczenia dotyczące tak zdefiniowanego obiektu i jego identyfikatorów są następujące:

- **LO** – *Learning Object*, złożony modyfikowalny obiekt nauczania,
- **LOID** – *Learning Object Identifier*, identyfikator złożonego modyfikowalnego obiektu nauczania LO,
- **VSLOID** – *Version Learning Object Identifier*, identyfikator kolejnej wersji niemodyfikowalnej obiektu LO.
- **torrent** – informacja niezbędna do rozpowszechniania danej wersji VSLOID niemodyfikowalnej obiektu.

Wszystkie identyfikatory są kompatybilne z identyfikatorami systemów P2P, tzn. są to hasze na 160 bitach (20 bajtów). Rysunek 5.2 przedstawia relacje pomiędzy obiektem, jego wersjami i identyfikatorami. Powyżej obiektu nauczania podane są identyfikatory LOID i VSLOID w zapisie alfanumerycznym i wzajemne relacje między obiektami.



Rysunek 5.2: Obiekt L0, jego wersje VSL0ID oraz relacje między nimi.

Tak zdefiniowany obiekt nauczania w ramach sieci P2P stwarza możliwość jego modyfikacji przez kooperujących uczestników, poprzez identyfikatory rozróżniające utworzony obiekt i jego zmodyfikowane wersje pod warunkiem, że relacje przedstawione na Rys. 5.2 przechowywane są w systemie. Koniecznym było stworzenie metody zapamiętywania relacji pomiędzy wersjami i obiektem L0, tak aby umożliwić zidentyfikowanie kolejnych wersji VSL0ID i otrzymać je z sieci P2P w standardowy sposób, na podstawie informacji w *torrent*. Ponieważ rozwiązanie przewiduje wykorzystanie sieci P2P w celu udostępniania wersji VSL0ID, naturalnym podejściem jest wykorzystanie struktury DHT sieci P2P do zapamiętania relacji pomiędzy L0, VSL0ID i informacją w *torrent*.

Struktura DHT oferuje interfejs operacji na kluczach *key* i wartościach *value* w postaci dwóch funkcji: *Store(key, value)* zapamiętuje parę $(key, value)$, a *Get(key)* zwraca *value* zapamiętaną z *key* (lub wielokrotne wartości jeżeli takie były zapisane w DHT). Wszystkie relacje w obiekcie nauczania L0 są reprezentowane jako zestaw par $(key, value)$.

Poniżej, przedstawione są informacje o obiekcie nauczania LO zapisywane do DHT jako następująca reprezentacja: po lewej stronie klucz *key* po prawej wartość *value* [12]:

$$\text{LOID} \rightarrow \text{VSLOID} \quad (5.1)$$

Do zapisania tej informacji należy wykonać operację:

$$\text{Store}(\text{LOID}, \text{VSLOID})$$

a gdy zna się LOID, można odszukać VSLOID poprzez operację:

$$\text{Get}(\text{LOID}) \rightarrow \text{VSLOID}$$

Następne relacje zapisywane są w podobny sposób – relacja odwrotna do wzoru 5.1:

$$\text{VSLOID} \rightarrow \text{LOID}$$

oraz relacja pomiędzy wersją i **torrent** – obiektowi niemodyfikowalnemu VSLOID przyporządkowano **torrent**, który zawiera informacje niezbędne do jego dostępu w sieci P2P:

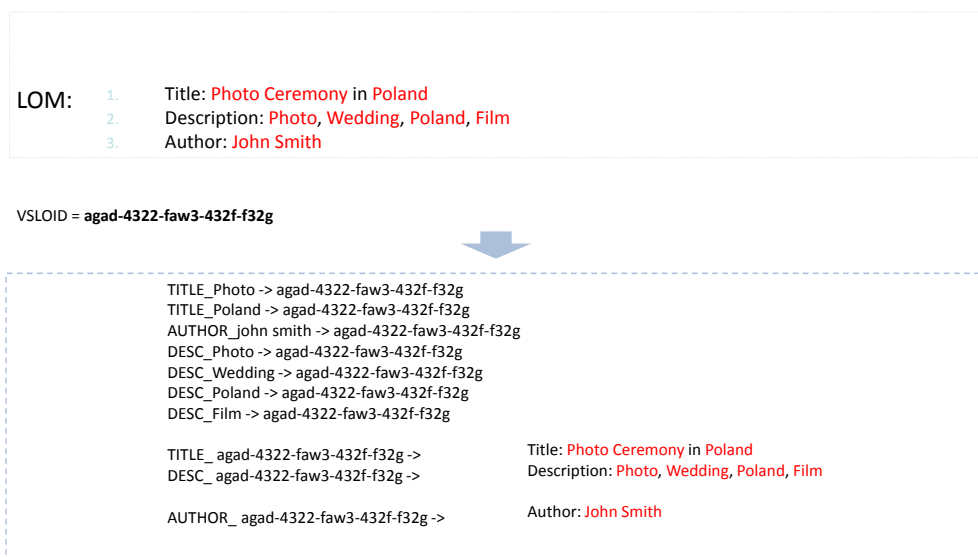
$$\text{VSLOID} \rightarrow \text{torrent}$$

Znajomość tych relacji i identyfikatora obiektu pozwalają użytkownikowi na pobranie go z sieci P2P.

5.2 Indeksacja obiektów nauczania

Systemy P2P takie jak BitTorrent, wykorzystujące DHT, indeksują identyfikatory obiektów łącząc je z ich adresami fizycznymi, a klucze obiektów łączą z danymi do nich przypisanymi. Relacja zachodzi tylko w jedną stronę. Podczas prac nad obiektem nauczania zdawano sobie sprawę z konieczności powiązania wielu terminów ze sobą. Dodatkowo naturalnym jest powiązanie autora z jego dziełem jak i przypisanie dziełu jego autora.

Mając na uwadze fakt, że obiekt nauczania musi posiadać złożoną strukturę oraz fakt, że kooperujący uczestnicy mogą go zmieniać i wzbogacać, opracowano koncepcję złożonej indeksacji obiektu nauczania. Indeksacja oprócz wcześniejszej opisanej trywialnej relacji między dziełem a twórcą musi obejmować atrybuty przypisane obu stronom relacji jak i wzajemne między nimi zależności. Indeksacja, obiektu nauczania jest jednym z wielu problemów indeksacji obiektów w sieciach P2P [20]. Rozpatrując złożoność problemu, koniecznym było uwzględnienie ontologii klasyfikującej opisywane zasoby. Rysunek 5.3 przedstawia indeksację przykładowego obiektu – kolekcji fotografii i filmów ślubnych.



Rysunek 5.3: Przykład indeksacji obiektu.

Autor obiektu– kolekcji przypisuje do niej kilka atrybutów (ślub, miejsce, zdjęcia, film, uroczystość), tak aby jak najszerzej opisać zawartość semantyczną tego obiektu. W kolejnym kroku, przez inwersję, łączy obiekt ze słowami kluczowymi.

W wypadku złożonych obiektów nauczania, zawierają one różnorodne metadane zawarte w części informacyjnej, oparte są na dwóch źródłach.

Pierwszym są standardowe atrybuty semantyczne pochodzące z analizy zawartości obiektu, na przykład informacje zawarte w opisie obiektu SCORM lub w części zarządzającej LOM, jak również atrybuty nadawane przez autora. Atrybuty opisu semantycznego są zdefiniowane według standardu Dublin Core. Opis obiektów jest skorelowany z metadanymi obiektu SCORM a te, jak już wspomniano, również są wzorowane na standardzie Dublin Core. Drugim źródłem metadanych są pojęcia z ontologii dziedzinowej używanej do opisu semantycznego obiektu.

Indeksacja danego obiektu L0 polega na utworzeniu relacji pomiędzy atrybutami metadanych i obiektem VSL0ID – każdy element składowy opisu obiektu zostaje przypisany do danego obiektu.

Podobnie do zapamiętania relacji rządzących strukturą obiektu nauczania L0, indeks zostaje zapamiętany w DHT jako para (*key, value*), gdzie *key* jest konstruowany z nazwy atrybutu i słowa kluczowego występującego w atrybucie, a *value* = VSL0ID. Mechanizm indeksacji pokazany jest na przykładzie wykładu opisanego przez następujące atrybuty:

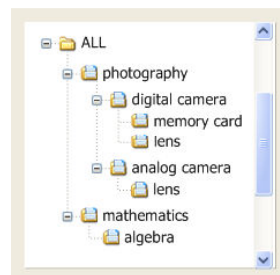
- *Title : Digital Photography Tips*
- *Autor : John Smith*

Indeks utworzony jest jako zbiór następujących par zapisanych w DHT:

Title_Digital → VSL0ID
Title_Photography → VSL0ID
Title_Tips → VSL0ID
Autor_John → VSL0ID
Autor_Smith → VSL0ID

Oprócz atrybutów indeksacja wykorzystuje pojęcia z ontologii dziedzinowej używanej do opisu semantycznego obiektu. Jak wspomniano poprzednio, ontologia dziedzinowa jest siecią pomiędzy pojęciami, która definiuje elementy i relacje semantyczne między nimi. Do najważniejszych relacji

w ontologii zaliczamy specjalizację i generalizację pojęć: ontologia tworzy drzewo pojęć, w którym pojęcia znajdujące się wyżej są bardziej ogólne od pojęć znajdujących się niżej. Im niższy poziom tym pojęcia są bardziej wyspecjalizowane. Rysunek 5.4 przedstawia przykład prostej ontologii reprezentującej relację specjalizacji pojęć użytą w prototypie GLEN na bazie klienta Azureus (cf. Rozdział 6). Na przykład: pojęcie **PHOTOGRAPHY** jest pojęciem generalnym w stosunku do **DIGITAL CAMERA**, jak również **ALGEBRA** jest pojęciem wyspecjalizowanym w stosunku do **MATHEMATICS**.



Rysunek 5.4: Przykład prostej ontologii dziedzicznej.

W tak zdefiniowanej ontologii istnieje możliwość dodawania pojęć ogólnych, nadrzędnych w stosunku do terminów szczegółowych i możliwość ich użycia w zapytaniach podczas poszukiwania właściwych elementów struktury drzewa. Dodatkowym efektem takiego podejścia jest możliwość rozróżnienia dwóch elementów, o tej samej nazwie, dzięki przyporządkowaniu ich do różnych poddrzew. Pozwala to na precyzyjniejsze wyszukiwanie pojęć.

W podobny sposób jak atrybuty, terminy z ontologii dziedzicznej są przypisane do obiektu. Ich indeksacja polega na zapisaniu następujących par w DHT przy założeniu, że termin **TERM** pochodzi z ontologii [15]:

$$\text{ONT_TERM} \rightarrow \text{VSLOID}$$

Na przykład, jeżeli autor indeksuje swój obiekt *Course on Digital Photography* (kurs fotografii cyfrowej) przyporządkowując do obiektu klucz *lens* (obiektyw), można również przypisać mu określenia nadrzędne z ontologii,

tak jak w przypadku jego określeń nadrzędnych *camera* i *photography*. Zapamiętane więc będą w DHT następujące pary:

$$\begin{aligned}\text{ONT_lens} &\rightarrow \text{VSL0ID} \\ \text{ONT_digital-camera} &\rightarrow \text{VSL0ID} \\ \text{ONT_photography} &\rightarrow \text{VSL0ID}\end{aligned}$$

Używając przykładowej ontologii z Rysunku 5.4 do indeksacji obiektów i poszukując kursu kojarzonego z terminem *digital photography* odnajdziemy go również podając klucz *lens*.

Wszystkie metadane są zapisywane jako zbiór następujących par w DHT:

$$\begin{aligned}\text{Title_VSL0ID} &\rightarrow \text{Title: Digital Photography Tips} \\ \text{Author_VSL0ID} &\rightarrow \text{Autor: John Smith} \\ \text{Desc_VSL0ID} &\rightarrow \text{Desc: photography, digital-camera, lens}\end{aligned}$$

Kiedy obiekt zostaje odszukany i pobrany, znajomość odwrotnych relacji pomiędzy danym obiektem a atrybutami jest również pożądana, aby dostarczyć metadane użytkownikowi. W tym celu następujące pary również zapisywane są do DHT:

$$\begin{aligned}\text{VSL0ID} &\rightarrow \text{Title_Digital} \\ \text{VSL0ID} &\rightarrow \text{Title_Photography} \\ \text{VSL0ID} &\rightarrow \text{Title_Tips} \\ \text{VSL0ID} &\rightarrow \text{Autor_John} \\ \text{VSL0ID} &\rightarrow \text{Autor_Smith} \\ \text{VSL0ID} &\rightarrow \text{ONT_lens} \\ \text{VSL0ID} &\rightarrow \text{ONT_digital-camera} \\ \text{VSL0ID} &\rightarrow \text{ONT_photography}\end{aligned}$$

Końcowym elementem indeksacji obiektu nauczania jest dodanie informacji o identyfikatorze VoIP uczestnika wymiany obiektu L0, który umożliwia nawiązanie kontaktu i tworzenie grup kooperacyjnych zainteresowań. Osoba pobierająca obiekt może podać swój adres SIP celem komunikacji ze sobą innym zainteresowanym poprzez komunikator VoIP. Identyfikator obiektu przechowany jest jako prefiks komunikacyjny COM do identyfikatora VSL0ID, łączący go w parę z adresem SIP. W efekcie przechowujemy w DHT parę:

$$\text{COM_}'\text{VSL0ID}' \rightarrow \text{SIP-ID}$$

Podsumowując, proces wprowadzania nowego obiektu do sieci P2P przebiega następująco:

1. Metadane otrzymywane są poprzez analizę zawartości obiektu nauczania w formacie SCORM.
2. Tworzony jest plik L0 do pobierania, zawierający obiekt SCORM, metadane i listę kontaktów.
3. Generowany jest identyfikator L0ID w postaci haszu na metadanych.
4. Generowany jest identyfikator VSL0ID dla początkowej wersji.
5. Tworzony jest **torrent** dla pliku L0.
6. Dodawane są następujące pary do DHT:
 $\text{L0ID} \rightarrow \text{VSL0ID}, \text{VSL0ID} \rightarrow \text{L0ID}, \text{VSL0ID} \rightarrow \text{torrent}.$
7. Tworzone są indeksy metadanych i dołączane do DHT.
8. Informacja o kontakcie z autorem dodana jest do listy kontaktów SIP.

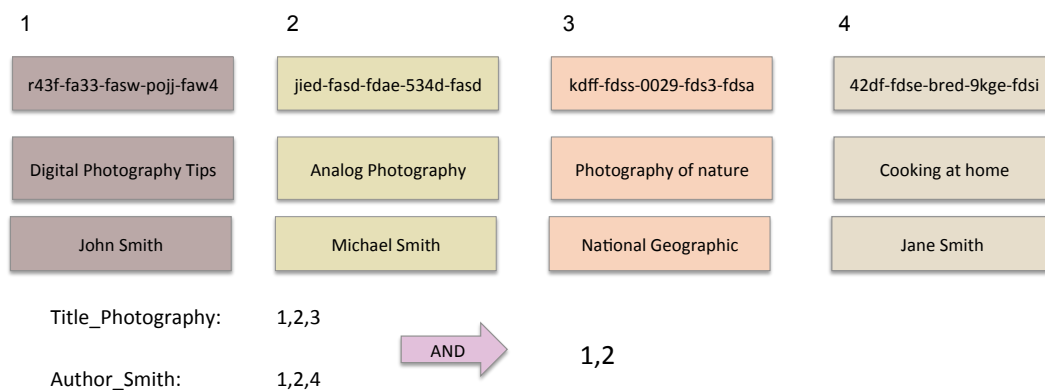
Powyższy proces realizowany jest pod kontrolą użytkownika, który może wzbogacić metadane poprzez specyficzne słowa kluczowe i pojęcia z danej ontologii dziedzinowej. Tak skonstruowany obiekt jest gotowy do procesu wyszukiwania i pobierania.

5.3 Proces wyszukiwania obiektów nauczania

Wyszukiwanie obiektów polega na sformułowaniu zapytania i przeszukiwaniu DHT, tak aby odnaleźć obiekty, których metadane zawierają słowa kluczowe lub terminy związane z poszukiwaną tematyką. Gdy zapytanie zawiera jedno słowo kluczowe, na przykład “Title: Photography”, wyszukiwanie sprowadza się do następującej operacji:

$$Get(\text{Title_Photography}) \rightarrow \text{VSL0ID}$$

Zapytania bardziej złożone wymagają odczytania indeksów dotyczących każdego słowa kluczowego i przetwarzania zbiorów rezultatów. Na przykład, poszukując obiektu zapisanego według dwóch zadanych kluczy trzeba zastosować do otrzymanych poszczególnych wyników, operator logiczny AND. Rysunek 5.5 przedstawia procedurę wyszukiwania dla zapytania “Title: Photography” i “Author: Smith”.



Rysunek 5.5: Procedura wyszukiwania dla przykładowego zapytania.

Wyszukując według klucza *Photography* odnajdujemy obiekty 1, 2 i 3 a według klucza *Smith* obiekty 1, 2 i 4. W wyniku zastosowania operatora AND wybrane zostają obiekty 1 i 2.

Podsumowując, proces wyszukiwania obiektu polega na realizacji następujących działań:

1. Użytkownik formułuje zapytanie dotyczące poszukiwanych obiektów nauczania.
2. Aplikacja P2P proponuje wzbogacenie zapytania o terminy z ontologii dziedzinowej.
3. Terminy zapytania pozwalają na wykonanie operacji *Get()* na DHT.
4. Znalezione obiekty i ich metadane prezentowane są użytkownikowi.
5. Użytkownik decyduje się na pobranie danego obiektu – jego **torrent** jest pobierany z DHT.
6. Informacja o kontakcie z użytkownikiem dodana jest do listy kontaktów SIP obiektu.

Ten proces wyszukiwania pozwala na utworzenie grupy użytkowników, którzy są zainteresowani daną tematyką.

5.4 Semantyczna indeksacja i wyszukiwanie obiektów nauczania

W częściach 5.2 i 5.3 przedstawiono sposób indeksacji i wyszukiwania obiektów nauczania polegający na zapamiętaniu słów kluczowych i terminów z ontologii w DHT, tak aby móc odszukać wszystkie obiekty. Można określić tę metodę jako jednopoziomową – wszystkie indeksy zapisane są w DHT co pozwala na odzyskanie odpowiednich obiektów po jednym dostępie do indeksów.

Felber *et al.* [20] zaproponowali inną metodę poszukiwania opartą o hierarchiczne wielopoziomowe indeksy i sukcesywną specjalizację złożonych zapytań. W przeciwieństwie do metody jednopoziomowej, indeksy nie zawierają wprost identyfikatorów obiektów, lecz inne coraz bardziej wyspecjalizowane zapytania. Użytkownik rozpoczyna poszukiwanie od sformułowania

zapytania ogólnego i stopniowo odkrywa zapytania bardziej specjalizowane, aż do uzyskania jednoznacznej informacji o poszukiwanych obiektach, która pozwala na dostęp do nich. W niniejszej pracy zaproponowano rozszerzenie hierarchicznej metody wielopoziomowej o użycie ontologii jako podstawy do definicji stopniowanych specyficznych zapytań.

5.4.1 Wielopoziomowa indeksacja semantyczna

Dla wyjaśnienia metody indeksacji wprowadzono poniższe przykłady obiektów nauczania opisanych przez następujące deskryptory:

Deskryptor d_1
Title: Photography Tips
Autor: John Smith

Deskryptor d_2
Title: Digital Cameras
Autor: John Smith

Deskryptor d_3
Title: Digital Cameras
Autor: John Doe

Dla danego obiektu O opisanego deskryptorem d , można łatwo skonstruować zapytanie q , takie że d jest jedynym deskryptorem odpowiadającym na zapytanie q (znajomość d pozwala na dostęp do obiektu). Takie zapytanie q jest zdefiniowane jako *zapytanie najbardziej szczególne*.

Dla danych dwóch zapytań q_1 i q_2 , zdefiniowano relacje *pokrycia* oznaczoną przez $q_1 \supseteq q_2$, jeżeli każdy deskryptor d_1 , który odpowiada na zapytanie q_1 również odpowiada na zapytanie q_2 . Relacja pokrycia również może dotyczyć deskryptorów: q_1 pokrywa d ($q_1 \supseteq d$), jeżeli $q_1 \supseteq q_2$ i q_2 jest najbardziej szczególnym zapytaniem dla d .

Relacja pokrycia będzie zilustrowana na następujących przykładach:

Zapytanie q_1
Title=Photography Tips and Autor=John Smith

Zapytanie q_2

Title=Digital Cameras and Autor=John Smith

Zapytanie q_3

Autor=John Smith

Zapytanie q_4

Title=Photography Tips

Zapytanie q_5

Title=Digital Cameras

Następujące relacje istnieją pomiędzy przykładowymi zapytaniami:

$$q_3 \supseteq q_1, q_3 \supseteq q_2, q_4 \supseteq q_1, q_5 \supseteq q_2$$

jak również zachodzą relacje pomiędzy zapytaniami i deskryptorami: q_1 i q_4 pokrywa d_1 , q_2 pokrywa d_2 , q_3 pokrywa d_1 , q_5 pokrywa d_2 i d_3 .

Schemat indeksacji polega na zapamiętywaniu następujących informacji w DHT. Dla danego obiektu O o deskryptorze d i o najbardziej szczególnym zapytaniu q , obiekt O zapisany jest na węźle odpowiadającym identyfikatorowi $k = h(q)$, co odpowiada zapisaniu obiektu tam, gdzie można go odszukać znając zapytanie q . Następnie generowana jest sekwencja zapytań

$$Q = \{q_1, q_2, \dots, q_n\} \text{ takich, ze } q_i \supseteq q. \quad (5.2)$$

Dla każdego q_i generowany jest identyfikator

$$k_i = h(q_i)$$

i para

$$(q_i, q)$$

która zapamiętywana jest w DHT w węźle odpowiadającym identyfikatorowi k_i .

Można zauważyć, że relacja pokrycia tworzy częściowy porządek pomiędzy wszystkimi zapytaniami, które odpowiadają danemu deskryptorowi. Zapytania te mogą być przedstawione hierarchicznie w formie skierowanego acyklicznego grafu. Najbardziej szczególne zapytanie jest korzeniem grafu, a każda krawędź reprezentuje informację indeksowania, która przyporządkowuje zapytanie wierzchołka dziecka do zapytania wierzchołka ojca. Istnieją wielorakie drogi do korzenia, a liczba operacji *GET* w DHT, aby znaleźć dany obiekt jest równa liczbie wierzchołków na najkrótszej drodze od początkowego zapytania do korzenia.

5.4.2 Rozszerzenie o hierarchię pojęć

Przedstawiona reprezentacja sugeruje, że indeksacja obiektów może być rozszerzona o hierarchię terminów z ontologii dziedzinowej. Załóżmy, że ontologia dziedzinowa zdefiniowana jest jako zbiór pojęć:

$$C = \{c_1, c_2, \dots, c_M\} \text{ takich, że } c_i \sqsupseteq c_{i+1},$$

gdzie relacja pokrycia $c_i \sqsupseteq c_{i+1}$ została rozszerzona w sposób następujący: c_i pokrywa c_{i+1} jeżeli każdy deskryptor d_i , który odpowiada pojęciu c_i również odpowiada pojęciu c_{i+1} .

Powyższą relację pokrycia można zilustrować na następujących przykładach:

Pojęcie c_1
Photography

Pojęcie c_2
Digital Camera

Pojęcie c_3
Lens

Następujące relacje istnieją między przykładowymi pojęciami:

$$c_1 \sqsupseteq c_2, c_1 \sqsupseteq c_3, c_2 \sqsupseteq c_3$$

Można stwierdzić, że w podobny sposób jak hierarchia zapytań pozwala na definiowanie coraz bardziej szczególnych zapytań dotyczących obiektów, hierarchia pojęć odzwierciedla coraz bardziej szczególne pojęcia charakteryzujące obiekty.

Pozwala to na rozszerzenie metody indeksowania na bazie sekwencji zapytań poprzez dodanie możliwości użycia pojęć z ontologii dziedzinowej. Na przykład, Deskryptor d_2 może mieć następującą formę:

```
Deskryptor  $d_2$ 
Title: Digital Cameras
Autor: John Smith
Concept: Photography
Concept: Digital Camera
```

W podobny sposób, generowana sekwencja zapytań (wzór. 5.2) do zapisania w DHT może zawierać zarówno atrybuty jak i pojęcia:

$$Q = \{Q_1, Q_2, \dots, Q_n\}, Q_i = q_i \text{ or } c_i. \quad (5.3)$$

5.4.3 Proces wyszukiwania

Proces wyszukiwania rozpoczyna się od sformułowania przez użytkownika zapytania Q_0 . Sformułowanie to zostanie wysłane do węzła odpowiadającego identyfikatorowi $k_0 = h(Q_0)$. Węzeł zwróci poszukiwany obiekt jeżeli Q_0 jest dla niego najbardziej szczególnym zapytaniem lub pojęciem. W przeciwnym wypadku, węzeł zwraca listę zapytań $Q = \{Q_1, Q_2, \dots, Q_n\}$ takich, że pary (Q_0, Q_i) , $Q_i \in Q$ są zapamiętane na węźle n . Użytkownik może wybrać z listy jedno lub kilka zapytań Q_i i rekursywnie powtórzyć proces poszukiwania z wybranym zapytaniem. Odszukiwanie kończy się, gdy zwrócone jest najbardziej szczególne zapytanie lub pojęcie, które pozwala na dostęp do obiektu. W ten sposób użytkownik podąża drogą indeksów zawierających coraz bardziej szczególne zapytania i pojęcia, aż do znalezienia obiektu.

Proces odszukiwania może być zilustrowany poniższym przykładem. Załóżmy, że indeksy zostały utworzone na jednym węźle:

$$q_3 \sqsupseteq q_2, q_3 \sqsupseteq d_1, q_4 \sqsupseteq d_1, q_5 \sqsupseteq q_2, q_5 \sqsupseteq d_3.$$

a na drugim

$$q_2 \sqsupseteq d_2.$$

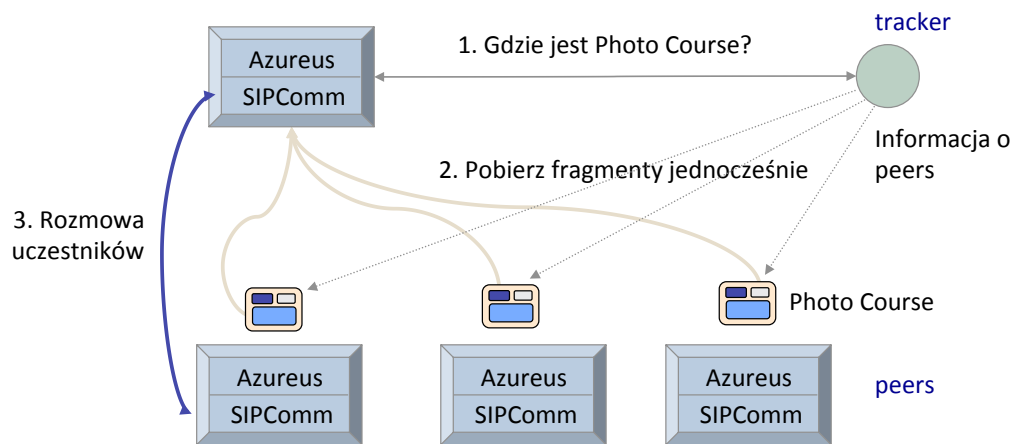
Użytkownik formułuje początkowe zapytanie q_3 i otrzymuje z pierwszego węzła zapytania q_2 i d_1 . Wysyła do drugiego węzła zapytanie q_2 i otrzymuje rezultat d_2 , co pozwala na dostęp do obiektów jednoznacznie zdefiniowanych poprzez deskryptory d_1 i d_2 . Podobny sposób dostępu jest możliwy gdy zapytania zawierają również pojęcia z ontologii.

Wybór indeksów dla danego obiektu jest arbitralny, jeżeli tylko zachowana jest relacja pokrycia pomiędzy zapytaniami. Można zauważyć, że odszukiwanie danego obiektu może się odbyć na podstawie różnych dróg indeksów. Aby zapewnić efektywne poszukiwanie, dany obiekt powinien być indeksowany wystarczającą ilość razy używając atrybutów jak najbardziej znanych użytkownikom, tak aby droga poszukiwania była jak najkrótsza.

5.5 Kooperacja pomiędzy uczestnikami grupy

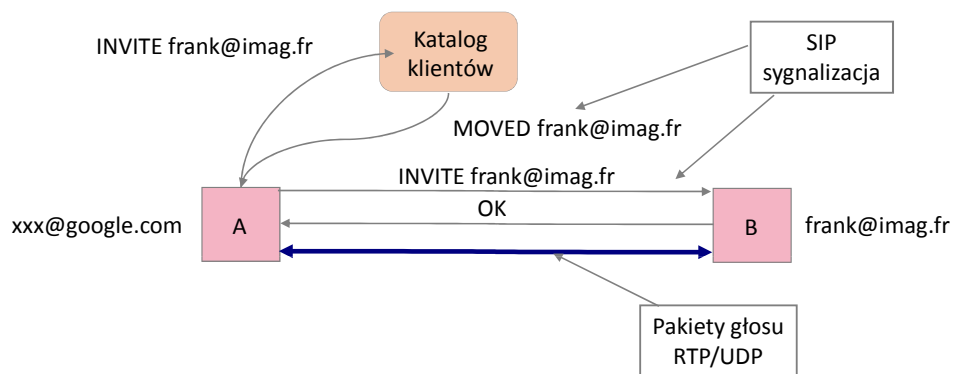
Jednym z celów tworzenia obiektu nauczania była integracja z systemami komunikacji pomiędzy uczestnikami procesu edukacyjnego. Celem było wzbogacenie obiektu nauczania o kontakty lub identyfikatory VoIP w taki sposób, aby uczący się mógł łatwo nawiązać kontakt z osobami, które wykład utworzyły lub już go przestudiowały. Istotną była również łatwość nawiązywania komunikacji przy użyciu popularnych środków technicznych.

Rysunek 5.6 obrazuje sposób funkcjonowania systemu. Podczas wymiany plików w sieci P2P uczestnicy otrzymują informacje o innych kooperatorach, z którymi mogą łatwo porozumiewać się poprzez VoIP. Możliwość



Rysunek 5.6: Funkcjonowanie systemu P2P z komunikacją VoIP.

wzajemnego porozumiewania się w czasie rzeczywistym pozwala na tworzenie grup kooperatywnego nauczania, w których „każdy może każdego czegoś nauczyć, o coś wzbogacić, coś wyjaśnić”.



Rysunek 5.7: Zasada komunikacji VoIP

Uzupełniając opis zagadnienia na Rysunku 5.7 przedstawiono zasadę komunikacji poprzez VoIP. Technologia VoIP umożliwia przesyłanie głosu poprzez łącza internetowe. Pierwszym etapem jest odnalezienie abonenta zarejestrowanego w katalogu klientów danego komunikatora lub systemu. Następnie nawiązane jest połączenie i inicjalizacja przesyłu strumienia danych z wykorzystaniem protokołu inicjalizacji sesji SIP (Session Initiation Protocol) [26] w przypadku komunikatorów używających otwartych standardów. Popularny komunikator Skype funkcjonuje podobnie, jednakże używa zamiast SIP własnego protokołu sygnalizacyjnego.

Działanie protokołu SIP polega głównie na inicjowaniu sesji pomiędzy klientami. Dostarcza on również rozmaitych funkcji umożliwiających operacje typu wybieranie numeru czy też generowanie sygnału zajętości. SIP jest odpowiedzialny jedynie za sygnalizację, a co za tym idzie współpracuje z innymi protokołami. SIP wspierany jest przez IETF i obecnie stał się dominującym protokołem sygnalizacyjnym dla VoIP. Po nawiązaniu łączności następuje przesył kodowanego głosu strumieniem danych RTP (Real-time Transport Protocol).

W przypadku komunikacji użytkowników danego komunikatora rozmowy są bezpłatne (jeżeli pominąć stały koszt abonamentu). Ponieważ transmisja VoIP odbywa się podobnie jak zwykły ruch sieciowy za pośrednictwem routerów i serwerów sieciowych, umożliwia ona zastosowanie wielu usług dodatkowych, niemożliwych lub kosztownych w realizacji w tradycyjnych sieciach telefonicznych [14].

W początkowej fazie raportowanych prac związanych z dysertacją, planowane było stworzenie rozwiązań na bazie technologii VoIP dla e-learningu. W trakcie badań, koncentrowano się bardziej na wykorzystaniu sieci P2P niż na aspektach związanych z VoIP. W proponowanym rozwiązaniu propagacji obiektu nauczania w sieci P2P, aspekt VoIP narzuca warunek modyfikowalności obiektu nauczania i pojawia się w formie listy identyfikatorów SIP

związanej z danym obiektem, tworzonej i utrzymywanej w sposób dynamiczny. Tym niemniej zaawansowane serwisy oferowane przez komunikację VoIP sugerują, że w przyszłości narzędzia informatyczne dla e-learningu powinny w pełni wykorzystać ten typ współpracy pomiędzy uczącymi. Pozostawiono tę problematykę do rozwinięcia w przyszłych pracach.

Rozdział 6

Walidacja proponowanych rozwiązań. Prototyp propagujący obiekt nauczania

Po fazie opracowania koncepcji złożonego obiektu nauczania i metody jego indeksacji, skoncentrowano się na eksperymentalnej walidacji proponowanych rozwiązań poprzez implementację prototypu, któremu nadano nazwę GLEN (Global Lecture Exchange Network), a jego zadaniem jest wykorzystanie popularnych sieci P2P i ich klientów do propagacji złożonych obiektów nauczania.

Pierwszą decyzją był wybór sieci P2P. Mając na uwadze właściwości różnych systemów rozpowszechniających pliki w sieci, dla propagacji obiektu nauczania zdecydowano się na BitTorrent. Najistotniejszym czynnikiem była szeroka decentralizacja struktury tej sieci. Kolejną zaletą BitTorrenta jest wysokoefektywny sposób pobierania danych oraz jego rosnąca popularność wśród użytkowników sieci P2P. Atutem tej technologii jest również możliwość pracy z trackerem jak i bez niego (wersja *trackerless* oparta na strukturze DHT Kademlia).

Kolejna decyzja dotyczyła problemu wyboru klienta tej sieci. Rozpatrzono następujące aplikacje:

- μ Torrent [37] charakteryzuje się znikomym obciążeniem systemu podczas pracy dostarczając przy tym wszystkie zaawansowane aplikacje cechujące systemy P2P. Umożliwia jednoczesne pobieranie i wysyłanie wielu plików. μ Torrent jest darmowo rozpowszechniany w sieci BitTorrent z możliwością instalacji w systemach operacyjnych Windows, Linux, Mac OS. Niestety twórcy programu pozostawili jego kod źródłowy zamkniętym co stanowi jego istotną wadę.
- Azureus jest aplikacją napisaną w języku Java. Wymaga zainstalowania środowiska Java Runtime Enviroment na komputerze klienta. Tak jak w innych programach sieci P2P, dostępne są wersje dla najbardziej popularnych systemów operacyjnych. Udostępniany jest w sieci na licencji GPL – wolnego i otwartego oprogramowania. Posiada rozbudowany system statystyk informujących o połączonych peerach (adres IP, używany port, prędkość wysyłania i ściągania, przewidywany czas ściągania pliku) liczbie seed’ow i leech’ow oraz danych dotyczących pliku `torrent`.

Azureus [4] posiada wbudowany tracker, ale umożliwia również pracę w wersji bez trackera (*trackerless*). Tworzenie własnego pliku `torrent` i umieszczenie go na swoim trackerze uniezależnia użytkownika od obcych trackerów i ich niedomagań.

Od wersji o numerze 4.3.0 wprowadzono zmianę GUI ułatwiającą korzystanie z programu oraz wyeliminowano mniej popularne funkcje. Postawiono na zmniejszenie objętości Azureusa i prostotę obsługi, przy równoczesnym zachowaniu jego funkcjonalności. Tak zmodyfikowana wersja otrzymała nazwę Vuze [51].

Wybierając sieć propagującą obiekt nauczania należało liczyć się z możliwością zmiany kodu źródłowego ogólnie dostępnych programów P2P. Z kolei ingerencja w kod źródłowy funkcjonujących programów jest zadaniem bardzo trudnym i rzadko stosowanym. Należało uwzględnić dołączenie wtyczek

plug in, na potrzeby przesyłania obiektów nauczania o strukturze dużo bardziej skomplikowanej niż wymiana plików audio lub video.

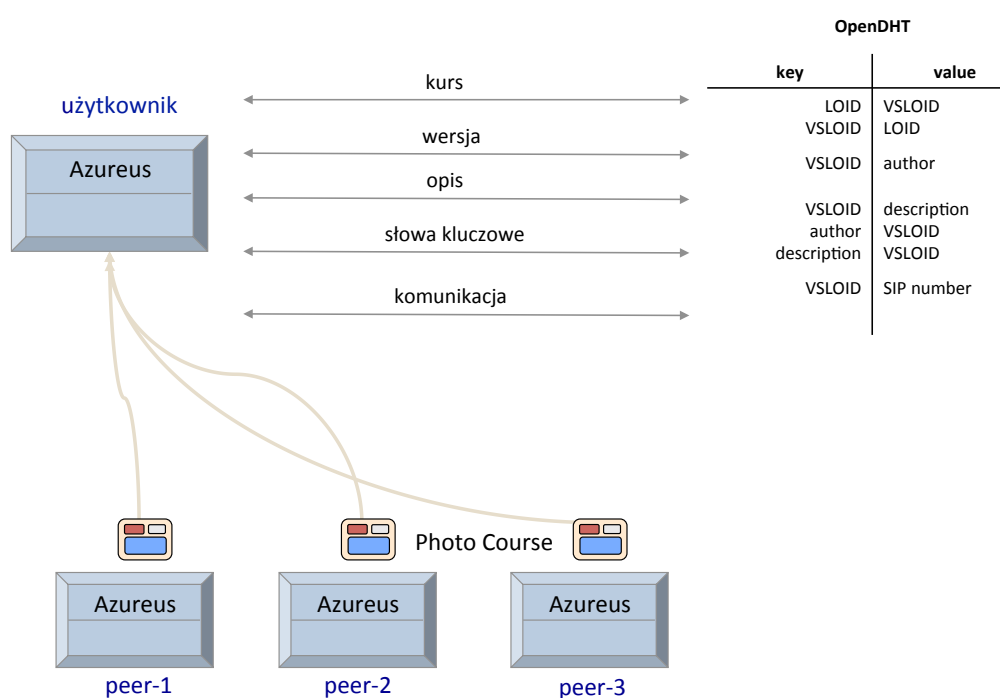
Przesyłając obiekty o takim przeznaczeniu zdecydowano się na standard SCORM mając również na uwadze konieczność pozyskania z niego pewnych informacji o samym obiekcie nauczania i poddanie ich ontologicznej klasyfikacji. Koniecznym było również dołączenie do obiektu modułu umożliwiającego wymianę treści edukacyjnych w sieci P2P. Zdecydowano się obdarzyć zaufaniem stosunkowo najpopularniejsze programy BitTorrent – Azureus i jego nowszą wersję Vuze.

6.1 Obiekt nauczania w prototypie GLEN

Dla obiektu i jego kolejnych wersji koniecznym było stworzenie metody poszukiwań pozwalającej na prosty i szybki dostęp do wszystkich wersji *LO*. Rozważano rozwiązanie z zastosowaniem DHT Azureusa. Jednak w tym przypadku DHT pozwala na poszukiwanie jednego obiektu – działa wykorzystując tylko jedną relację łączącą identyfikator upowszechniającego węzła (*nodeID*) z identyfikatorem danych (*key*) dającym dostęp do informacji zawartych w pliku *torrent*. W przypadku rozpatrywanym w dysertacji, struktura obiektu jest złożona oraz planowano również wykorzystanie DHT do zapamiętania i odszukiwania relacji indeksów, ponieważ obiekt posiada opis w postaci metadanych, które należy indeksować. Jednocześnie należało mieć na uwadze możliwość modyfikowania kolejnych wersji *LO*.

Konsekwencją zmian w obiekcie są zmiany w części informacyjnej, a w efekcie konieczność obliczenia nowego klucza *hash(descriptor)*, co nie jest przewidziane w DHT Azureusa. Użycie DHT Azureusa do zapisywania indeksów wiązało się z problemem modyfikacji aplikacji Azureus, tak aby relacje opisane w Rozdziale 5 były zapisywane w DHT. Wymagałoby to instalacji zmodyfikowanej wersji Azureusa na wielu komputerach użytkowników, co było bardzo trudnym przedsięwzięciem.

Z powyższych powodów w rozwiązaniu prototypowym GLEN zdecydowano się na umieszczanie informacji o obiektach nauczania w systemie DHT zewnętrznym do prototypu. Również indeksacja i wyszukiwanie obiektów realizowane były z użyciem zewnętrznego systemu DHT. Prototyp może pracować z dwoma zewnętrznymi systemami DHT: OpenDHT [40] i Bamboo [5]. Rysunek 6.1 przedstawia zasadę działania prototypu GLEN.



Rysunek 6.1: Zasada działania prototypu GLEN.

OpenDHT to projekt pozwalający na testowanie i eksperymenty na szeroką skalę poprzez wykorzystywanie mocy obliczeniowej oferowanej przez komputery w ramach systemu PlanetLab [42]. W PlanetLab od 2003 roku ponad tysiąc instytutów badawczych i uniwersytetów udostępnia swoje “oddelegowane” komputery na potrzeby projektów naukowych w zakresie sieci i systemów rozproszonych. OpenDHT jest eksperymentem w dziedzinie systemów P2P na dużą skalę, który rozwinął pierwotny projekt P2P

Bamboo. Korzystając z możliwości PlanetLab, Sean Rhea w 2005 roku zaprojektował i zaimplementował OpenDHT jako rozproszony system tablic DHT, oferujący użytkownikom rozproszoną bazę danych, którą można wypełnić zestawem par $\langle key, value \rangle$ [46]. Klucz to pole mieszczące 20 bajtów natomiast na wartość przeznaczono 1024 bajty.

GLEN może być również skonfigurowany do pracy z Bamboo. W tym wypadku trzeba mieć do dyspozycji serwer Bamboo i wprowadzić jego adres do konfiguracji GLEN. Niedogodnością tego rozwiązania jest centralny aspekt serwera Bamboo – wszystkie operacje *Store* i *Get* wykonywane są na jednym serwerze.

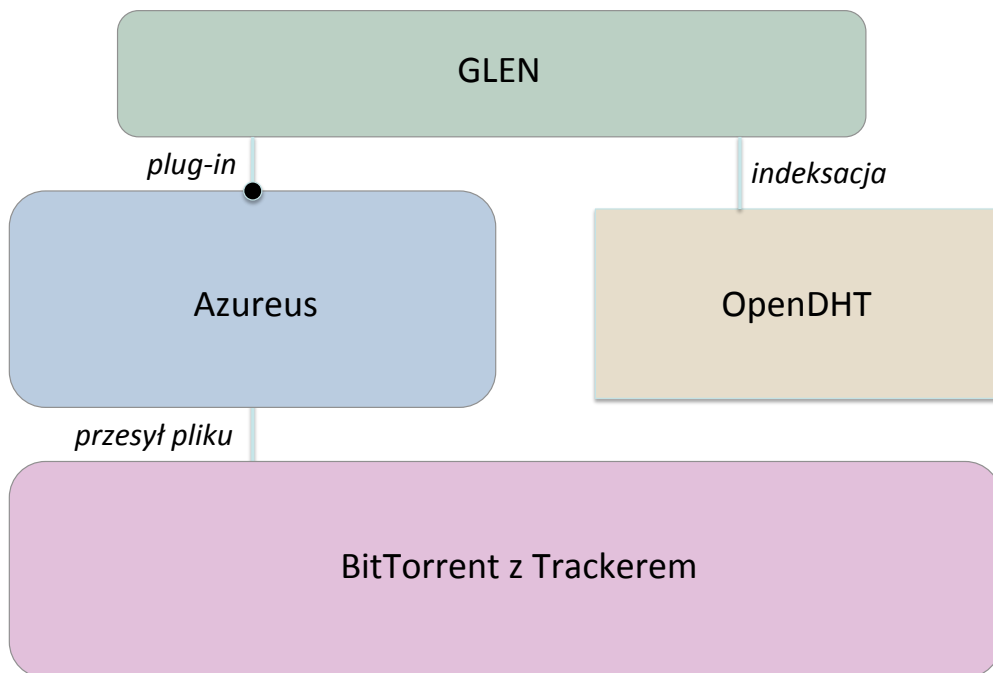
6.2 Prototyp GLEN

GLEN jest zmodyfikowaną aplikacją Azureusa, która oferuje użytkownikowi funkcje dodawania nowego obiektu nauczania do systemu propagacji i poszukiwania obiektów. Funkcje GLEN zrealizowane są jako wtyczka *plugin*, która pozwala na rozszerzanie funkcjonalności Azureusa w standardowy sposób. Rysunek 7.1 przedstawia schemat prototypu GLEN oraz jego relacje z Azureusem i OpenDHT.

GLEN operuje na obiektach nauczania przygotowanymi w formacie SCORM 1.2. Do ich przygotowania użyto edytora eXe, który pozwala na integrację różnych materiałów źródłowych w jeden złożony obiekt. Edytor pozwala również na dołączenie metadanych w formacie LOM zakodowanym przy pomocy składni XML. Obiekt nauczania ma formę jednego pliku archiwalnego ZIP.

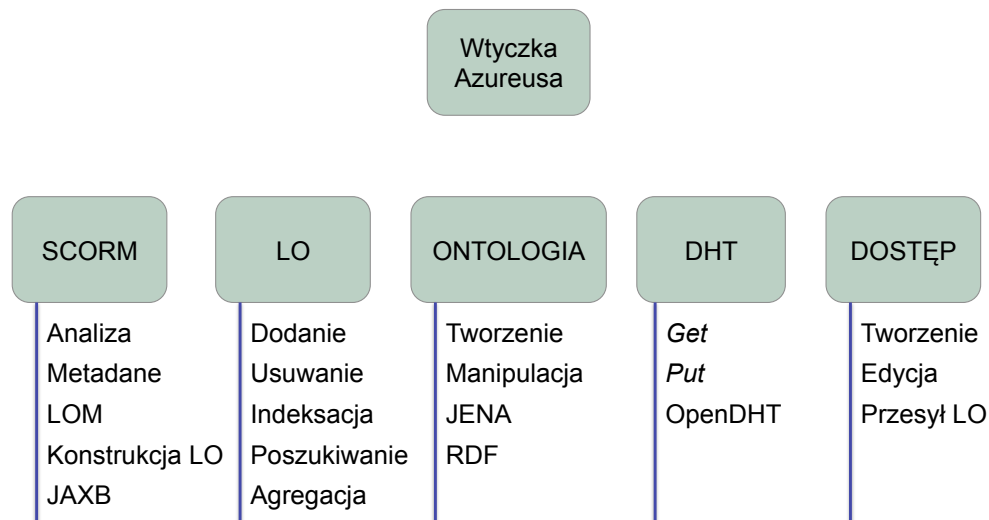
6.2.1 Struktura prototypu GLEN

Rysunek 6.3 przedstawia strukturę prototypu GLEN. Składa się on z następujących modułów:



Rysunek 6.2: Schemat prototypu GLEN.

- SCORM: moduł zawierający funkcje do analizy formatu SCORM. Wykorzystuje bibliotekę JAXB (Java Architecture for XML Binding) [27] do analizy syntaktycznej pakietu SCORM i utworzenia jego strukturalnej postaci drzewiastej. Pozwala również na analizę formatu LOM i ekstrakcję metadanych. Konstruuje obiekt LO na podstawie formatu SCORM.
- LO: moduł zarządzający obiektami nauczania, który oferuje funkcje dodawania obiektów do systemu rozpowszechniania P2P jak również



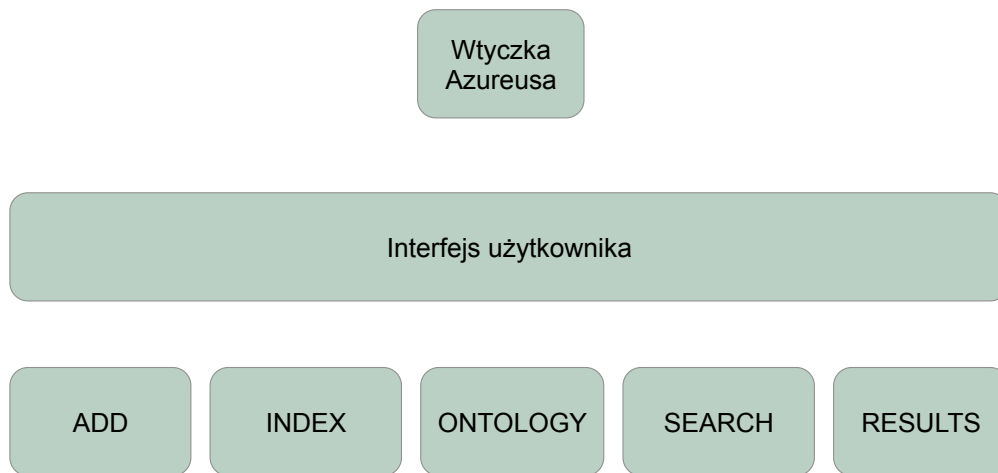
Rysunek 6.3: Struktura prototypu GLEN.

ich edycji i usuwania. Opiera się na funkcjach modułów SCORM (analiza dodawanego obiektu), ONTOLOGIA (wybór pojęć do opisu semantycznego), DOSTĘP (zapisanie informacji o dostępie do pliku torrenta). Używa modułu DHT do zapisu indeksów i wyszukiwania obiektów odpowiadających na zapytania wprowadzane przez użytkownika. Dokonuje również agregacji rezultatów poszukiwania.

- ONTOLOGIA: moduł zawierający procedury tworzenia i manipulacji ontologiami. Wykorzystuje bibliotekę JENA (Semantic Web Framework for Java) [28] do konstrukcji ontologii w formacie RDF. Dostarcza pojęć w fazie indeksacji obiektów i ich poszukiwania.
- DHT: moduł stanowi interfejs pomiędzy Azureusem a OpenDHT. Daje wizję rozproszonej bazy danych z operacjami *Put()* i *Get()*. Używa dostępu do sieci OpenDHT poprzez pakiet zdalnego wywoływania procedur Apache XML RPC [58, 3].

- DOSTĘP: moduł, który pozwala na tworzenie i edycję plików torrenta jak również transfer danego obiektu na podstawie informacji zawartych w pliku torrent. Łączy podstawowe funkcje dostępu do plików Azureusa z prototypem GLEN.

Rysunek 6.4 przedstawia strukturę interfejsu użytkownika. Opiera się on na podstawowym interfejsie Azureusa i rozszerza go poprzez mechanizm wtyczek. Dostarcza funkcje do zarządzania okienkami, wyświetlania informacji i współpracy z użytkownikiem.



Rysunek 6.4: Struktura interfejsu użytkownika.

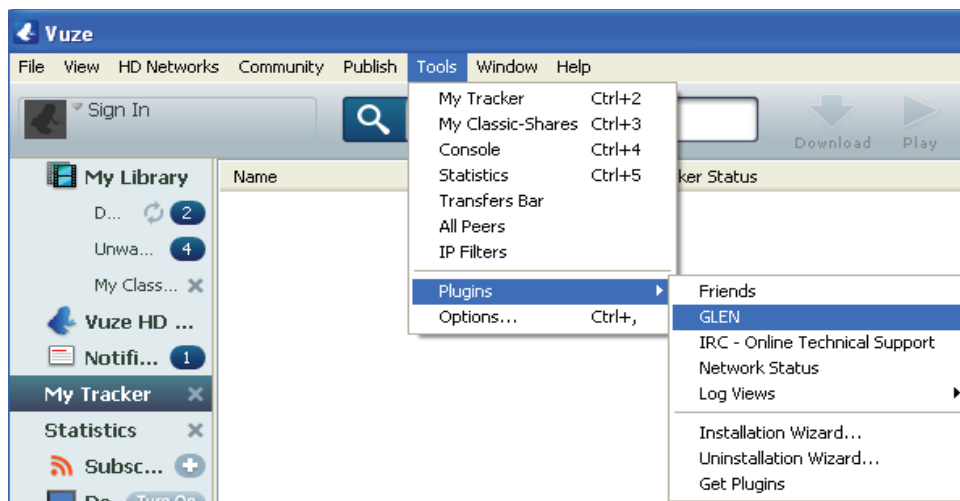
Interfejs użytkownika składa się z następujących modułów:

- ADD: pozwala na wprowadzanie złożonego obiektu nauczania do systemu, jego analizę i ekstrakcję metadanych.
- INDEX: dostarcza okienko dialogu z użytkownikiem do dodawania terminów indeksujących.
- ONTOLOGY: pozwala na wprowadzanie pojęć z ontologii przy indeksacji obiektu i jego wyszukiwaniu.

- SEARCH: zarządza okienkiem dialogu z użytkownikiem, który definiuje zapytania dotyczące obiektów nauczania.
- RESULTS: pozwala na agregację i wyświetlanie rezultatów poszukiwania.

6.2.2 Przykład użycia prototypu GLEN

Rozpoczynając pracę w GLEN należy zainstalować programy środowiska Java, samego Azureusa oraz nakładki do Vuze i wybrać wtyczkę GLEN, tak jak jest to przedstawione na Rysunku 6.5.

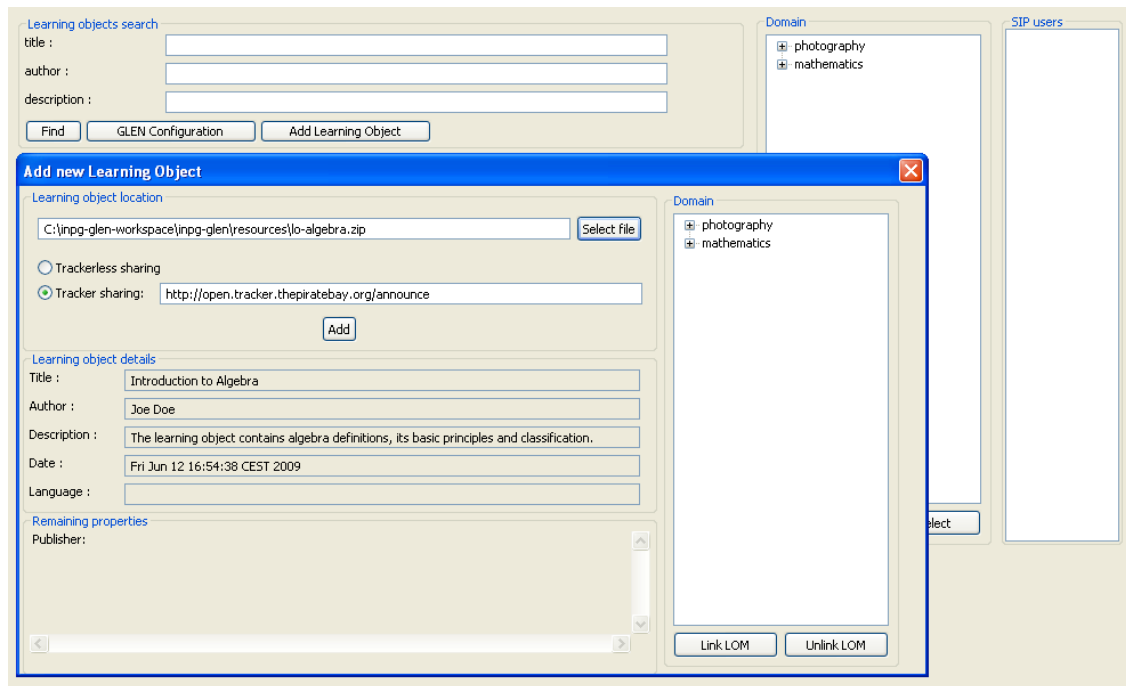


Rysunek 6.5: Konfiguracja Azureusa z wtyczką GLEN

Propagowanie obiektu w prototypie GLEN

Okienko dialogu pozwala na wskazanie pliku, z którego wprowadzany obiekt zostanie pobrany (cf. Rysunek 6.6). GLEN analizuje podstawowe atrybuty semantyczne i podaje najważniejsze z nich (*Title*, *Author*, *Description*, *Date*, i *Language*). Mogą one być również wprowadzone przez użytkownika.

Użytkownik może przyporządkować swój obiekt do istniejącej ontologii (pokazanej po prawej stronie okna jako *Domain*). Prototyp zintegrował przykładową ontologię utworzoną w formacie RDF przy użyciu biblioteki



Rysunek 6.6: Interfejs dodawania obiektu

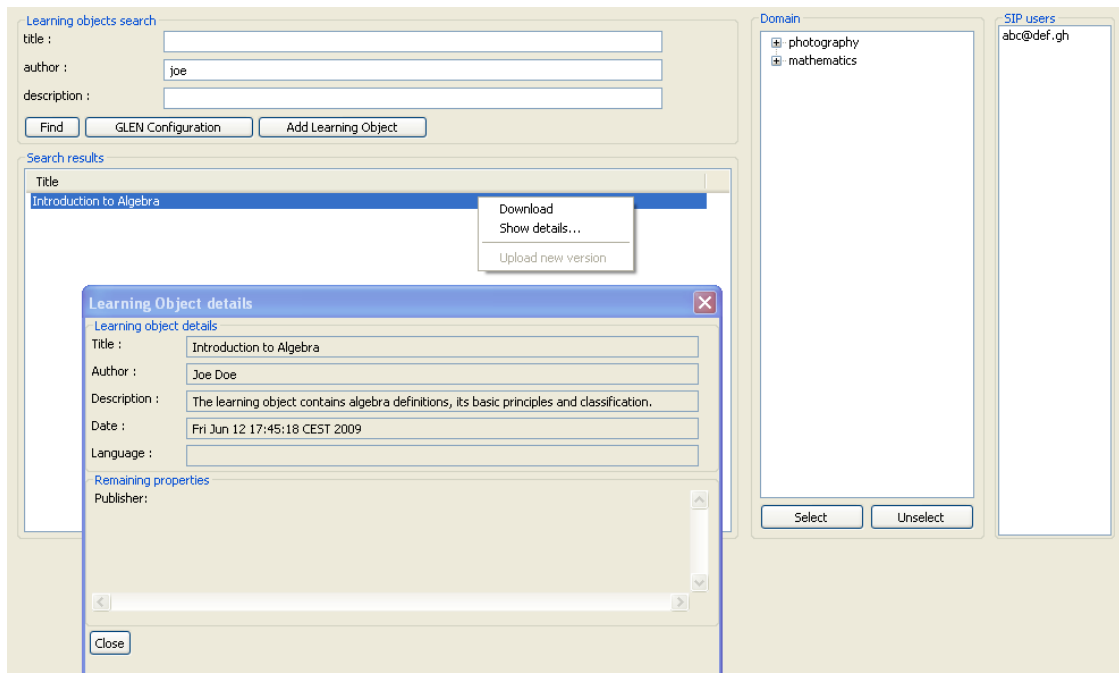
JENA. Nakłada to ograniczenie na uczestników używania tej samej ontologii. Tym niemniej, ontologia ta może być publiczna i podlegać ewolucji, tak aby dostosować się do potrzeb użytkowników.

W przykładzie pokazanym na Rysunku 6.6, użytkownik ma do wyboru dwa poddrzewa – *photography* i *mathematics*. Wybiera obiektywy (*lens*) w poddrzewie *photography*. GLEN przechowuje indeksowane informacje o obiekcie w OpenDHT, jak opisano to wcześniej – wskazanie miejsca obiektu w istniejącej ontologii powoduje automatyczne dodanie go do systemu zapytań w OpenDHT i przypisanie do obiektu VSLOID. Informacje te umożliwią późniejsze jego wyszukiwanie.

Interfejs pozwala również na dołączenie do indeksacji obiektu LOM zawierającego inne metadane (przycisk *Link LOM*). Użytkownik ma również wybór pomiędzy użyciem wersji z trackerem (*tracker sharing*) lub wersji trackerless do rozpowszechniania obiektu.

Wyszukiwanie obiektu w prototypie GLEN

Okienko dialogu wyszukiwania pozwala na wprowadzenie podstawowych atrybutów (*Title*, *Author*, *Description*) lub na wybranie terminów z ontologii dziedzinowej. Przykładowy proces wyszukiwania dla zapytania będącego atrybutem *Author: Joe* przedstawia Rysunek 6.7.



Rysunek 6.7: Interfejs wyszukiwania obiektu

GLEN wykonuje operacje *Get()* w DHT odpowiadające zapytaniu, tak jak zostało to przedstawione w Rozdziale 5. Rezultatem jest VSL0ID przypisany do tego atrybutu i przedstawiający listę obiektów spełniających to kryterium (w przedstawionym przykładzie zwracany jest tylko jeden obiekt o tytule *Introduction to Algebra*). Lista obiektów odpowiadających zapytaniu wyświetlana jest w okienku *Search results*. Klikając na jeden obiekt będący rezultatem wyszukiwania, użytkownik może poprosić o jego metadane, aby sprawdzić czy jest to obiekt, którego szuka lub od razu zainicjować jego pobranie. Po pobraniu obiektu przez użytkownika, jego kontakt wprowadzony jest do DHT (okienko *SIP users*).

Użycie ontologii dziedzinowej pozwala na przeglądanie (*browsing*) obiektów wprowadzonych do sieci, co jest interesujące, gdy użytkownik nie jest w stanie sprecyzować swego zapytania.

6.2.3 Dyskusja

Prototyp GLEN miał na celu walidację rozwiązań z punktu widzenia funkcjonalnego – w jaki sposób członek grupy kooperacyjnej może używać proponowanych funkcji oraz czy funkcje indeksacji i wyszukiwania działają poprawnie. Z tego punktu widzenia, prototyp zdał egzamin i pozwolił na otrzymanie skutecznie działającego narzędzia.

Niedogodnością prototypu jest użycie zewnętrznego systemu DHT. Wkrótce po realizacji GLEN i integracji z OpenDHT, ten światowy program – system OpenDHT został wstrzymany. Fakt, że GLEN używa zewnętrznego systemu DHT stanowi przeszkodę w popularyzacji prototypu i wprowadzeniu go na szerszą skalę do prób z rzeczywistymi użytkownikami. Tym niemniej, pozwolił on na wstępną funkcjonalną walidację proponowanych rozwiązań.

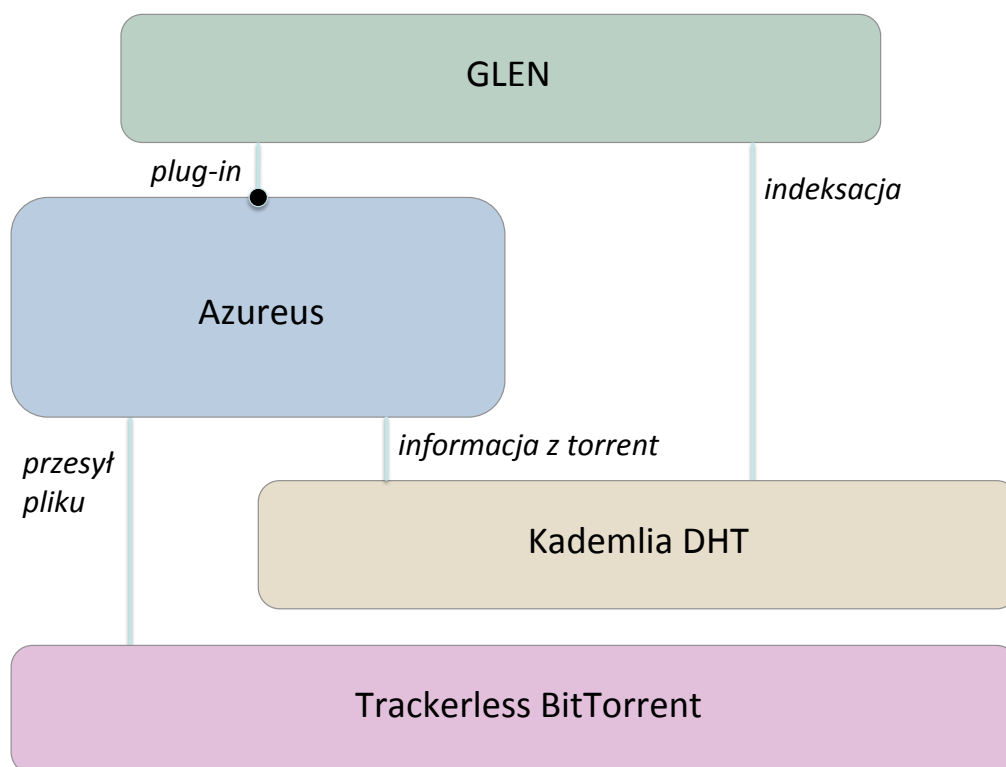
Rozdział 7

Ocena efektywności proponowanych rozwiązań

Prototyp GLEN pozwolił na walidację na małą skalę proponowanych rozwiązań z punktu widzenia funkcjonalnego, nie był jednak przystosowany do walidacji ich działania dla większej ilości obiektów i użytkowników, ani do oceny efektywności rozwiązań. Dodatkowym problemem było użycie struktury DHT zewnętrznej w stosunku do sieci P2P BitTorrent.

Proponowane rozwiązania mogłyby być zaimplementowane jako rozszerzenie Azureusa pracującego na bazie trackerless BitTorrent, który używa DHT Kedemlii. Rysunek 7.1 przedstawia schemat takiej implementacji.

Implementacja taka i eksperymenty z dużą ilością użytkowników wymagałyby integracji GLEN z rozprowadzanymi wersjami Azureusa oraz zaakceptowania takiej wersji przez użytkowników, co jest trudne do osiągnięcia. W tej sytuacji postanowiono użyć symulacji do reprezentacji zachowania GLEN z DHT Kademlia i trackerless BitTorrent. Zaletą symulacji jest możliwość przeprowadzenia eksperymentów z dużą ilością węzłów zachowując jednocześnie precyzję reprezentacji algorytmów DHT Kedemlia i BitTorrent.



Rysunek 7.1: Schemat rozwiązania na bazie DHT Kedemlia i trackerless BitTorrent.

Celem symulacji jest zbadanie efektywności proponowanych rozwiązań poprzez analizę czasu wykonywania operacji w sieci P2P. W wypadku proponowanej metody reprezentacji złożonego obiektu nauczania i jego indeksacji, znaczna ilość informacji zapisywana jest do struktury DHT. W podobny sposób, w procesie wyszukiwania, zapytanie użytkownika umożliwia dostęp do wielu par w DHT. Ilość zapisywanych i odczytywanych par zależy od wielu czynników: zakresu metadanych i pojęć ontologicznych, struktury zapytań użytkownika, etc. W przedstawianych symulacjach liczba par do zapisania lub do odczytania przyjęta jest jako parametr. Metryką efektywności jest czas operacji indeksacji i odszukiwania, który można porównać z czasem tych samych operacji w wypadku standardowego

funkcjonowania sieci P2P dla jednego pliku: jeden zapis do DHT i jedno odczytanie.

7.1 Wybór symulatora sieci P2P

Istnieje kilka wysokiej klasy symulatorów sieci (OPNET [41], ns2 [35], Qualnet [43], OMNet++ [49]), ale tylko nieliczne oferują możliwość łatwej symulacji systemów P2P. Uwagę skupiono na Peer Sim [36] i OverSim [7, 8], dwóch spełniających założone kryteria: otwarte oprogramowanie, gotowa symulacja BitTorrent i łatwość jej realizacji.

PeerSim posiada licencję wolnego i otwartego oprogramowania (GPL). Napisany w Java implementuje wiele popularnych algorytmów takich jak Pastry, Chord, Kademlia i BitTorrent.

OverSim jest symulatorem P2P działającym w środowisku symulatora sieci OMNet++. Oferuje standardową symulację BitTorrent w wersji z trackerem oraz trackerless na bazie DHT Kademlia. Po analizie symulatorów, OverSim został wybrany do oceny efektywności.

7.2 Symulator propagacji obiektów nauczania

OMNeT++ jest środowiskiem symulacji na bazie dyskretnych zdarzeń zorientowanym obiektowo. Program symulacji składa się z kolekcji modułów połączonych poprzez Bramki (Gates). Bramki są interfejsami wejścia/wyjścia modułów. Taka struktura umożliwia hierarchiczne konstruowanie modeli, w których złożone moduły tworzone są poprzez odpowiednie połączenia modułów prostszych. Komunikacja pomiędzy modułami odbywa się poprzez wymianę komunikatów reprezentujących ramki (warstwa MAC) lub pakiety (warstwa sieci). Przychodzące komunikaty odpowiadają dyskretnym zdarzeniom symulacji, które są przetwarzane przez odpowiedni moduł i powodują przesuwanie się symulowanego czasu. Komunikaty wysyłane przez

moduł do siebie samego są interpretowane jako zegary (Timers), pozwalające na ustawienie przetwarzania o określonym czasie. Klasa Kanał (Channel) reprezentuje zachowanie łącza i OMNeT++ dostarcza gotowe moduły do reprezentacji łączy przewodowych jak i bezprzewodowych.

Moduły programowane są w języku C++, a połączenia w specjalistycznym języku NED (NEtwork Description). Istnieją wersje dla trzech podstawowych systemów operacyjnych (Windows, Linux, MacOS).

Program symulacji OMNeT++ składa się z:

1. opisu topologii sieci w języku NED (pliki .ned),
2. definicji wymienianych komunikatów (pliki .msg),
3. kodu i nagłówków modułów (pliki .h i .cc).

Przykładowa sieć opisana w języku NED ma następującą postać:

```
//  
// Przykładowa sieć  
//  
network Network  
{  
    types:  
        channel C extends ned.DatarateChannel {  
            datarate = 100Mbps;  
        }  
    submodules:  
        node1: Node;  
        node2: Node;  
        node3: Node;  
        ...  
    connections:  
        node1.port++ <--> C <--> node2.port++;  
        node2.port++ <--> C <--> node4.port++;
```

```
node4.port++ <--> C <--> node6.port++;  
...  
}
```

Sieć *Network* składa się z kilku węzłów *node1*, *node2*, *node3*. . . . Niektóre z nich połączone są poprzez element *channel* reprezentujący łącze o przepustowości 100 Mbits/s. Typ *Node* nie jest tutaj zdefiniowany, jego definicja powinna być podana gdzie indziej, co ilustruje hierarchiczny sposób definiowania złożonych struktur sieciowych.

Komunikaty są centralnym elementem symulatora OMNeT++ reprezentowanym poprzez klasę *CMESSAGE* i podklasę *C PACKET*. Komunikaty posiadają nazwę, typ, priorytet, adresy źródłowy i docelowy, jak również czas dotarcia. Mogą być używane jako zegary – aby zaprogramować zdarzenie po danym okresie czasu, moduł może wysłać komunikat do siebie samego z czasem dotarcia odpowiadającym danemu przedziałowi czasu.

Każda zmiana w modułach wymaga ponownej kompilacji i budowy programu wykonywalnego.

Wykonanie symulacji kontrolowane jest poprzez plik inicjalizacji, który zawiera parametry i czas symulacji.

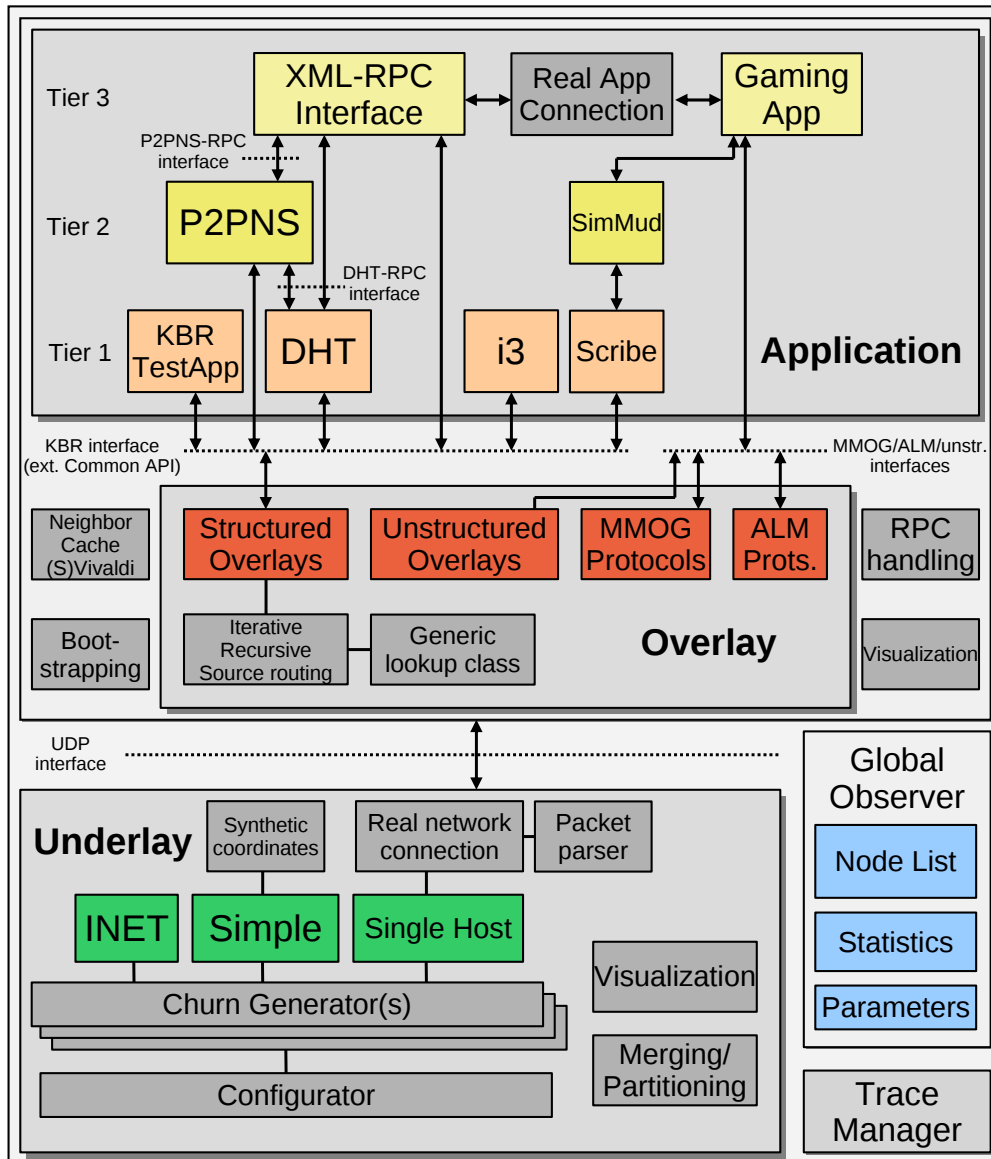
W czasie symulacji, symulator przechodzi poprzez trzy fazy:

1. inicjalizacja, w czasie której węzły dołączane są do sieci,
2. stabilizacja, w czasie której system zachowuje się w sposób zakodowany w modułach,
3. generacji rezultatów.

Po upływie zdefiniowanego czasu symulacji, wyniki są zapisywane do plików, które trzeba przetworzyć skryptami w języku Python i wygenerować wykresy.

OverSim stanowi bibliotekę modułów realizujących podstawowe funkcje komunikacyjne (INET, zachowanie węzłów) oraz zawiera główne systemy

P2P (Kademlia, Chord, Pastry, Gia, Scribe, etc.), interfejs DHT i aplikacje testów. Rysunek 7.2 przedstawia strukturę symulatora Oversim.



Rysunek 7.2: Struktura symulatora Oversim, źródło [8].

OverSim posiada modularną architekturę. Jego podstawę tworzą dwa główne moduły komunikacji: *Simple* i *INET*. W modelu *Simple* węzły znajdują się w n wymiarowej przestrzeni euklidesowej, w której opóźnienie pomiędzy dwoma węzłami zależy od ich wzajemnej odległości. Pozycje węzłów odpowiadają pomiarom opóźnień w sieci Internet dokonany w projekcie CAIDA/Skitter [16, 31]. Każdy węzeł posiada wirtualny interfejs dostępowy do sieci reprezentowany poprzez różne parametry przepustowości, opóźnienia i stopy strat pakietów. Taka reprezentacja pozwala realizować symulacje sieci P2P z dużą ilością węzłów (do 100000) i z tego powodu wykorzystano go w niniejszej pracy. *INET* integruje stos protokołów TCP/IP i pozwala na dokładną symulację niższych warstw protokolarnych włącznie z modelowaniem routerów. Większa dokładność w reprezentacji elementów sieci nie pozwala jednak na symulacje z dużą ilością węzłów. Moduł *SingleHost* daje możliwość podłączenia komputera do istniejącej sieci, testowania wybranego protokołu P2P i emulacji wykonywania aplikacji na węźle w realnych warunkach.

Moduł *Overlay* oferuje wszystkie najważniejsze protokoły P2P. Użytkownik wybiera dany protokół w pliku konfiguracji. Prezentowane symulacje używają protokołu Kademlii, który oferuje tak jak inne metody interfejs KBR (Key Based Routing). Warstwa wyższa reprezentuje aplikacje, jest ona podzielona na dwie części: *Application Tier 1* i *Application Tier 2*.

OverSim oferuje klasę **ChurnGenerator** do symulacji typowego zachowania węzłów w sieci P2P: podłączenie się, okres komunikacji i odłączenie się. Głównym parametrem klasy jest średni czas obecności w sieci. Istnieją różne typy generatorów:

- **No Churn**: pozwala symulować sieć statyczną ze stałą ilością węzłów.
- **LifetimeChurn**: kiedy tworzony jest węzeł, jego czas podłączenia jest wybierany losowo z obciążonego rozkładu normalnego o danej średniej

i wariancji (parametry te mogą być ustawione przez użytkownika w pliku konfiguracji).

- **ParetoChurn**: podobny do poprzedniego, lecz czas podłączenia jest wybierany losowo z rozkładu Pareto.
- **RandomChurn**: wybiera losowo liczbę co *targetMobilityDelay* sekund i na podstawie rezultatu, węzeł jest dołączany do sieci, odłączany lub przemieszcza się według prawdopodobieństw zdefiniowanych przez użytkownika.

Tablica 7.1 podaje parametry używane w symulacjach i zakres ich zmian.

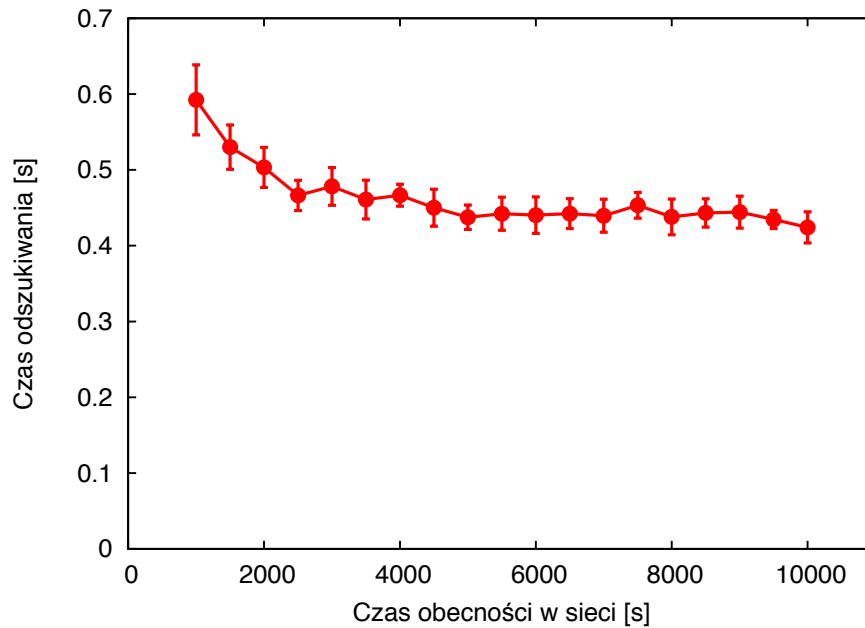
Tabela 7.1: Parametry symulacji

P2P	Kademlia
Generator zachowania węzła w sieci	LifetimeChurn
Liczba węzłów w sieci	$N = 100 \dots 1000$
Rozmiar pojemników w DHT Kademlia (<i>k-buckets</i>)	$k = 8$
Liczba równoległych RPC w poszukiwaniu DHT	$p = 3 \dots 10$
Średni czas obecności węzła w sieci (<i>lifetime</i>)	$c = 500 \dots 10000$
Liczba par do zapisu w DHT	$M = 1 \dots 500$

Wykresy przedstawione poniżej podawane są z przedziałami ufności dla poziomu 0.95.

7.3 Rezultaty symulacji

W tej części przedstawiono wyniki symulacji działania systemu DHT Kademlia z propagacją obiektów nauczania. Główną różnicą pomiędzy standardową siecią DHT Kademlia, taką jaka używana jest przez Azureusa w wersji trackerless, a proponowanymi rozwiązaniami, jest generowana ilość zapisów lub zapytań do DHT. W przypadku Azureusa, wprowadzenie obiektu wymaga jednego zapisu do DHT, tak jak jego odszukanie (parametr $M = 1$).



Rysunek 7.3: Średni czas odszukiwania obiektu, $N = 100$, $k = 8$, $M = 1$, słabe natężenie ruchu

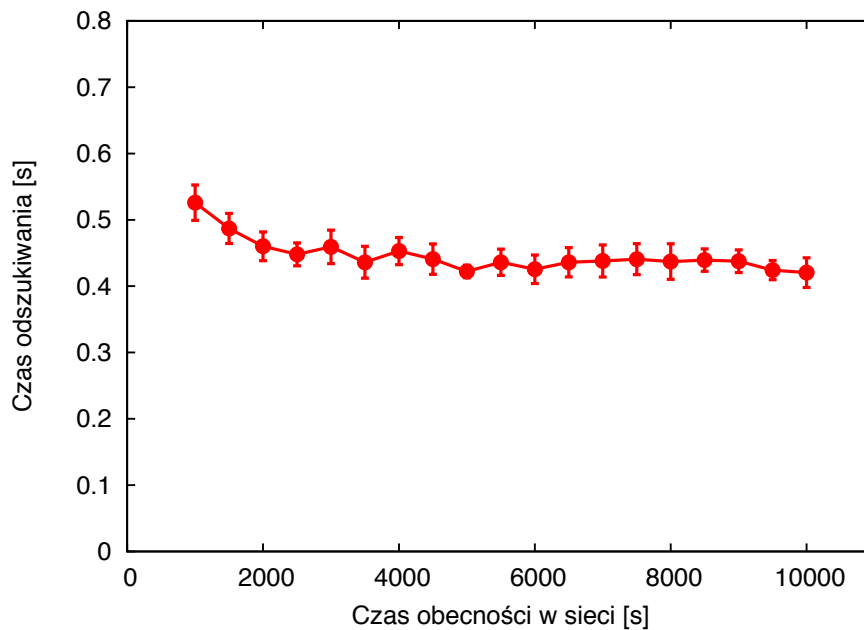
Gdy obiekt nauczania jest indeksowany, prawie w tym samym momencie generowana jest znaczna liczba zapisów do DHT, które trzeba w niej umieścić. W tym wypadku parametr M może przyjmować wartości zależne od ilości słów kluczowych lub terminów używanych do indeksacji. Sytuacja jest podobna w procesie wyszukiwania, który może generować znaczną ilość terminów w zapytaniach użytkownika. Przyjęto do symulacji zakres parametru $M = 1...500$.

Symulacja składa się z pliku konfiguracji wybierającego metodę Kademlii w Overlay i definiującego parametry symulacji, modułu definiującego strukturę DHT (*Application Tier 1*) i modułu aplikacyjnego DHTApp, który jest modyfikacją standartowego modułu Oversim DHTTestApp (*Application Tier 2*). DHTApp używa funkcji *Put()*, *Get()*, *Modify()*, aby odczytać, zapisać lub zmodyfikować pary $\langle key, value \rangle$. Zachowuje się jak aplikacja użytkownika wykonująca publikacje i wyszukiwanie plików. Węzeł posiada wszystkie warstwy protokołu Kademlii, komunikuje się z innymi

węzłami poprzez protokół UDP i realizuje funkcje DHT wyszukiwania par $\langle key, value \rangle$ na podstawie czterech procedur RPC Kademlii (PING, STORE, FIND_NODE, FIND_VALUE), takich jak zostało to opisane w Rozdziale 4.2.2.

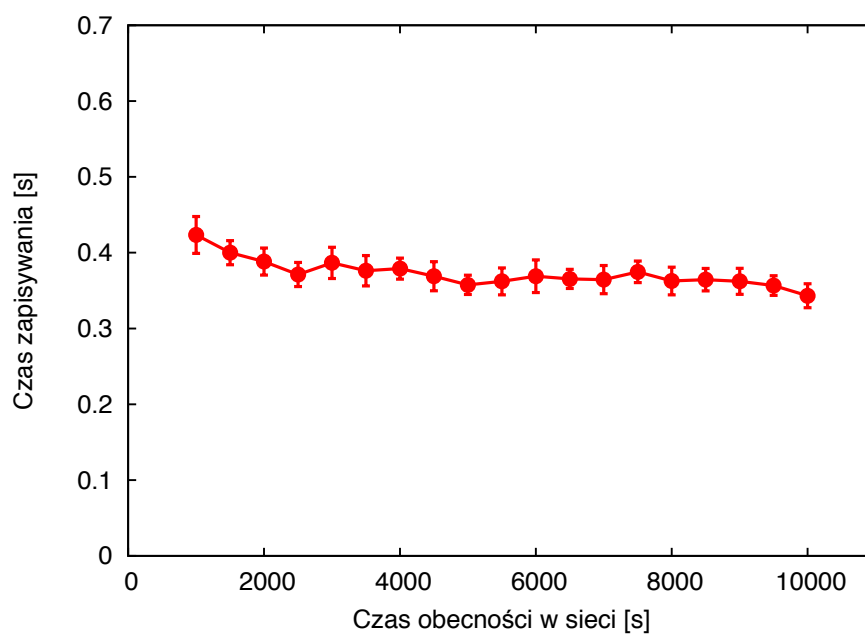
7.3.1 Efektywność standardowej sieci DHT Kademlia

W symulacjach badano zależność średniego czasu wyszukiwania od innych parametrów – głównie średniego czasu obecności węzła w sieci (churn rate lub lifetime) i natężenia ruchu.

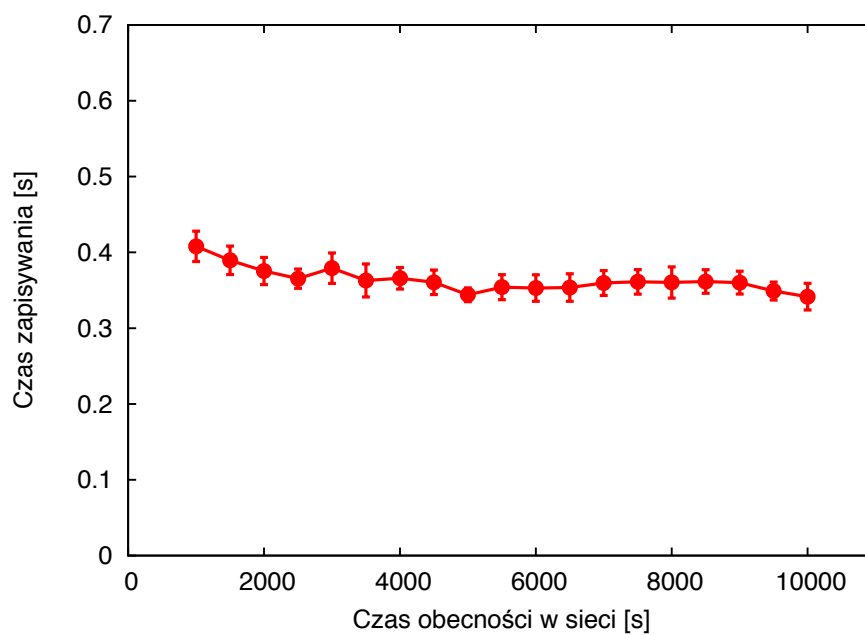


Rysunek 7.4: Średni czas odszukiwania obiektu, $N = 100$, $k = 8$, $M = 1$, silne natężenie ruchu

Rysunek 7.3 przedstawia czas odszukiwania obiektu (operacja *Get()*) w funkcji średniego czasu obecności węzła w sieci. Można zauważyć wydłużenie czasu odszukiwania dla krótkiego czasu obecności. Jest to charakterystyczne dla sieci DHT, w których czas poszukiwania jest praktycznie stały w funkcji ilości węzłów, jednocześnie wrażliwy jest na częste podłączenia i odłączenia węzłów.



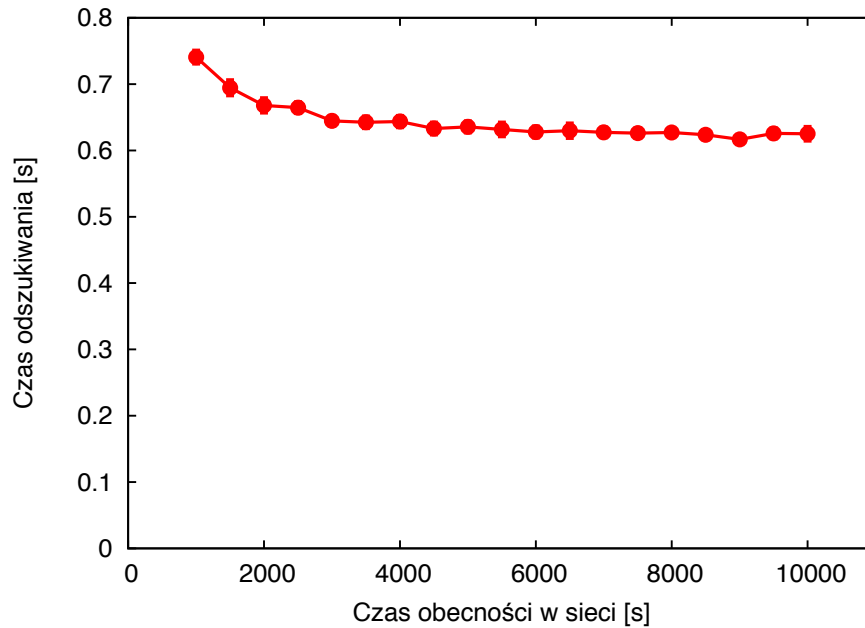
Rysunek 7.5: Średni czas zapisywania pary danych do DHT, $N = 100$, $k = 8$, $M = 1$, słabe natężenie ruchu



Rysunek 7.6: Średni czas zapisywania pary danych do DHT, $N = 100$, $k = 8$, $M = 1$, silne natężenie ruchu

Rysunek 7.4 przedstawia podobny wykres czasu odszukiwania obiektu w funkcji średniego czasu obecności węzła w sieci dla zwiększonego ruchu w sieci – zapytania generowane są z większą częstotliwością.

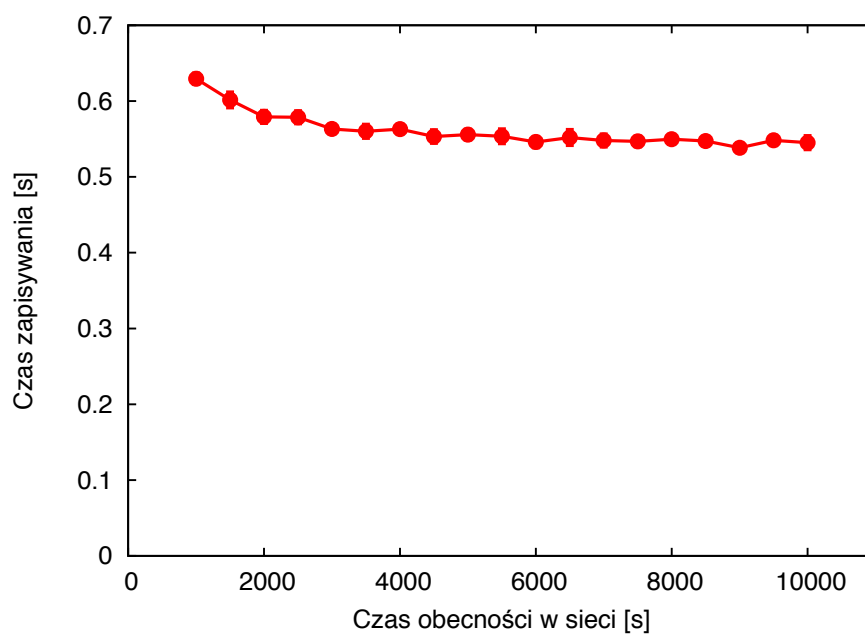
Rysunek 7.5 przedstawia wykres czasu zapisywania jednej pary danych (operacja *Store()*) w funkcji średniego czasu obecności węzła w sieci.



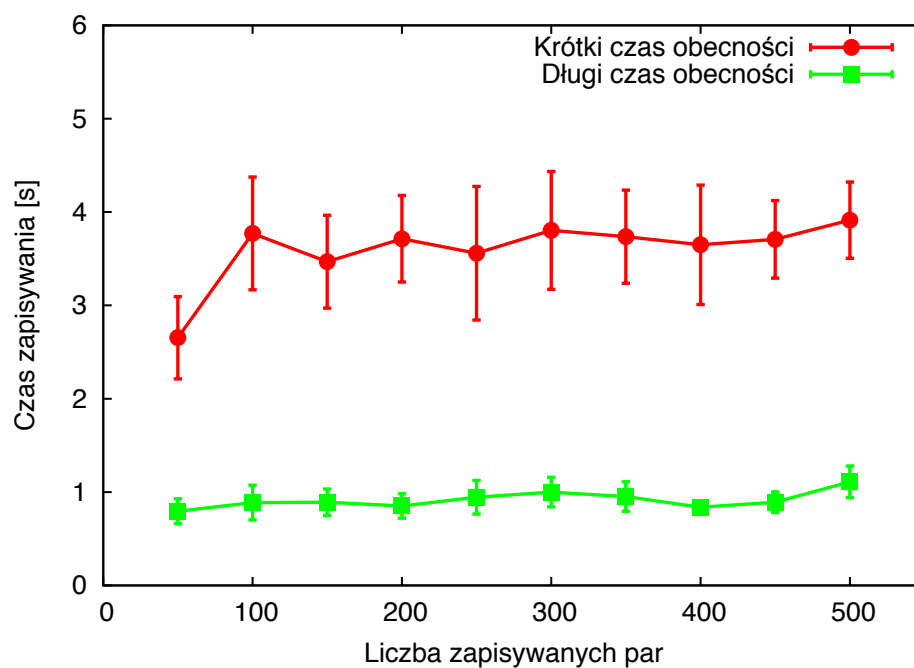
Rysunek 7.7: Średni czas odszukiwania obiektu, $N = 1000$, $k = 8$, $M = 1$, słabe natężenie ruchu

Rysunek 7.6 jest analogiczny do Rysunku 7.5 dla zwiększonego ruchu w sieci.

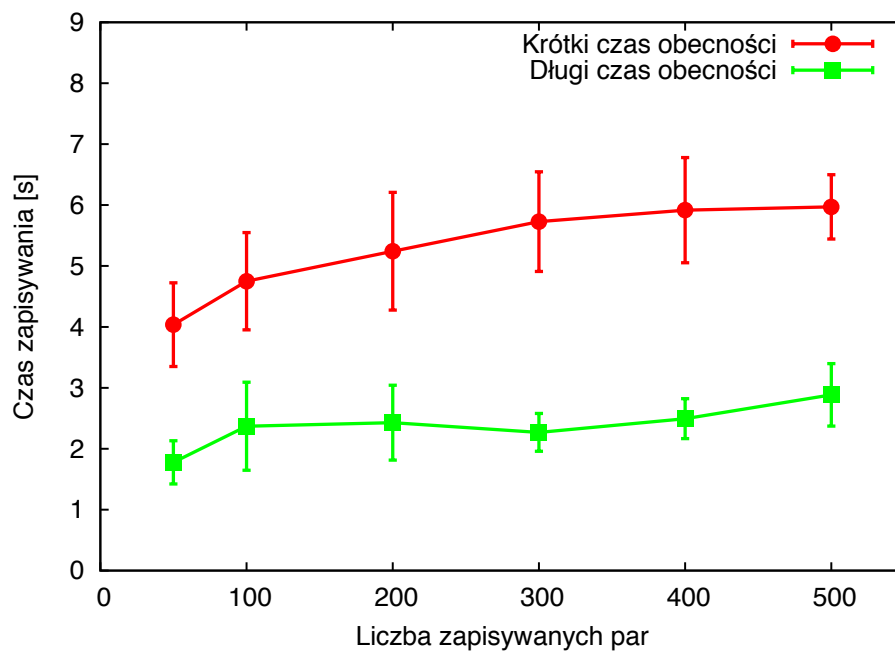
Rysunki 7.7 i 7.8 przedstawiają wykresy analogiczne do Rysunku 7.3 i 7.5 dla większej ilości węzłów ($N = 1000$). Istotną właściwością tych wykresów jest fakt, że czasy odszukiwania i zapisu są tylko nieznacznie powiększone mimo znacznego zwiększenia skali sieci.



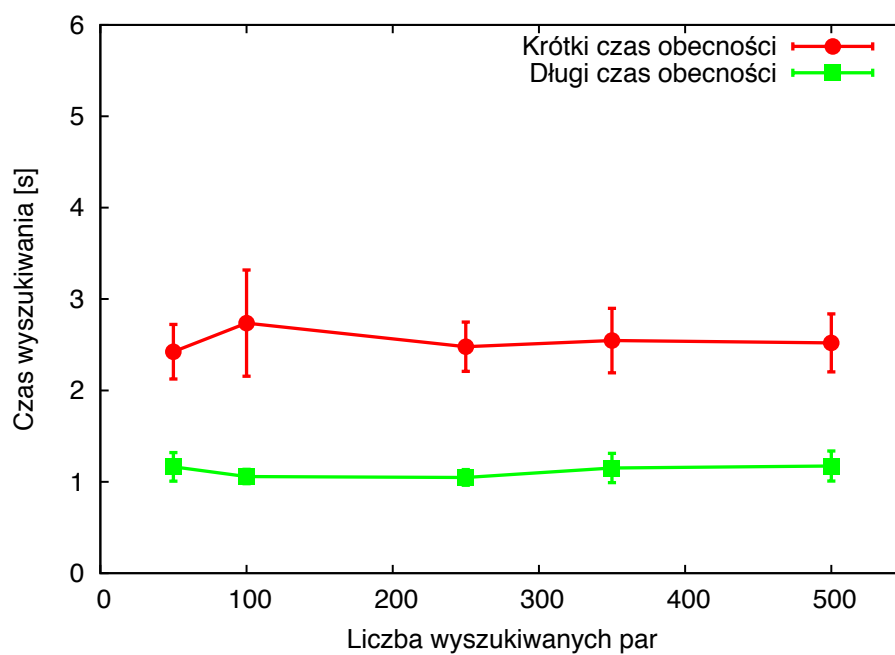
Rysunek 7.8: Średni czas zapisywania pary danych do DHT, $N = 1000$, $k = 8$, $M = 1$, słabe natężenie ruchu



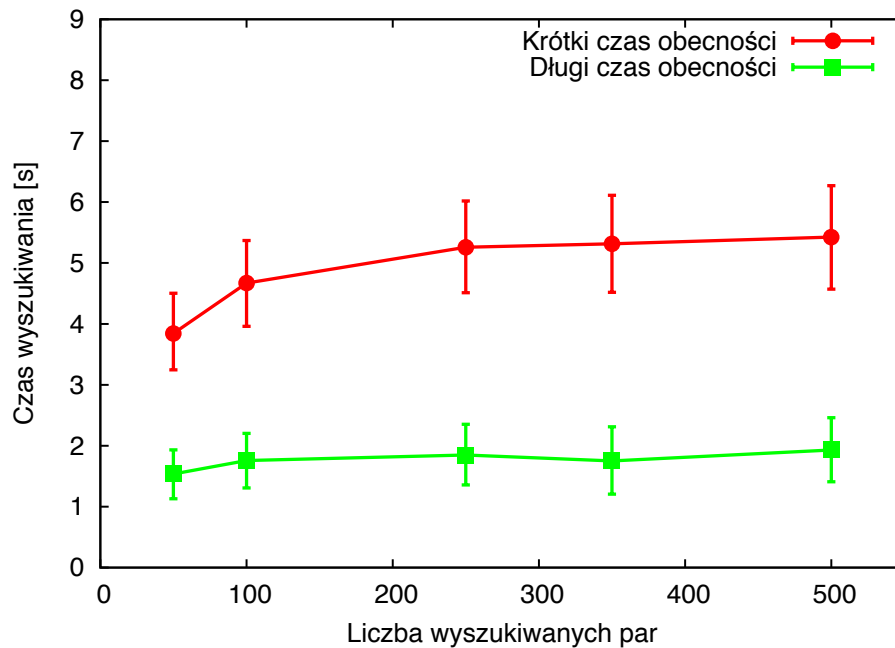
Rysunek 7.9: Średni czas zapisywania M par danych do DHT, $N = 100$, $k = 8$



Rysunek 7.10: Średni czas zapisywania M par danych do DHT, $N = 1000$, $k = 8$



Rysunek 7.11: Średni czas wyszukiwania M par danych w DHT, $N = 100$, $k = 8$



Rysunek 7.12: Średni czas wyszukiwania M par danych w DHT, $N = 1000$, $k = 8$

7.3.2 Efektywność sieci DHT Kademlia z propagacją obiektów

Symulacja sieci DHT Kademlia z propagacją obiektów polega na generacji znacznej ilości zapisów lub zapytań do DHT (parametr $M = 1...500$). DHTAppGLEN symuluje ruch sieciowy aplikacji GLEN, gdy użytkownik publikuje złożony obiekt nauczania. Indeksacja obiektu generuje zapis do DHT dużej ilości par $\langle key, value \rangle$. Może to obciążyć system i spowodować pogorszenie się jego efektywności. DHTAppGLEN generuje fałę M zapisów i mierzy czas niezbędny do wykonania tej operacji w funkcji ilości zapisów. W podobny sposób oceniany jest czas wyszukiwania M par.

Rysunek 7.9 przedstawia średni czas zapisywania M par danych w wypadku krótkiego czasu obecności węzła w sieci (krzywa czerwona) i dłuższego czasu obecności (krzywa zielona). Wykres pokazuje, że efektywność

zapisywania prawie nie zależy od ilości par, a decydujący wpływ ma czas obecności.

Rysunek 7.10 pokazuje analogiczne rezultaty dla większej ilości węzłów ($N = 1000$). Można zaobserwować na nim podobne zachowanie.

Podczas symulacji zbadano również zachowanie systemu podczas procesu wyszukiwania złożonych obiektów nauczania. Rysunki 7.11 i 7.12 przedstawiają czas wyszukiwania M par dla $N = 100, 1000$.

Na podstawie powyższych rezultatów można stwierdzić, że system P2P na bazie Kademlii wykazuje bardzo dobrą wydajność – czas zapisu i wyszukiwania w niewielkim tylko stopniu zależy od ilości węzłów w sieci. Ponieważ krótki pobyt węzła w sieci eliminuje zapisaną informację i zaburza funkcjonowanie routingu pomiędzy węzłami, krytycznym parametrem, od którego zależy wydajność, jest czas podłączenia węzła do sieci.

Rozdział 8

Podsumowanie i wnioski końcowe

Celem niniejszej pracy było rozpatrzenie możliwości wykorzystania systemów P2P do tworzenia kooperacyjnych grup nauczania. Aktualne systemy wspomagania nauczania oparte są głównie na technologiach WWW, a bogate narzędzia P2P używane są jedynie do rozpowszechniania popularnych plików muzycznych i filmowych. Ideą przewodnią pracy była analiza systemów P2P, zakres ich funkcji, możliwych rozszerzeń, które mogłyby przyczynić się do łatwego i szybkiego rozpowszechniania materiałów nauczania, a także ich wykorzystania do tworzenia kooperacyjnych grup uczących się. Jednocześnie proponowane rozszerzenia powinny ułatwiać komunikację pomiędzy uczestnikami procesu nauczania.

Warto wspomnieć, że obecne technologie P2P są jednymi z najbardziej skomplikowanych systemów informatycznych. Działają one na wielką skalę umożliwiając poszukiwania milionów plików na milionach podłączonych węzłów milionom użytkowników. Jest to jedna z różnic pomiędzy systemami P2P a prostym modelem klient/serwer WWW, która wprowadza dodatkowe utrudnienie w realizacji celów określonych w dysertacji.

Punktem wyjściowym pracy był system P2P BitTorrent, który pozwala na efektywny przesył plików w sposób równoległy i dysponuje implementacjami typu Open Source, co daje możliwość realizacji prototypów i ich walidacji.

Praca zawiera następujące oryginalne wyniki:

1. Pierwszym elementem proponowanych rozwiązań była struktura obiektu nauczania umożliwiająca wykorzystanie sieci P2P do rozpowszechniania obiektów nauczania i tworzenia kooperacyjnych grup zainteresowania. Obiekty te powinny zawierać niezbędne informacje i posiadać ściśle określoną strukturę. Ważną decyzją było oparcie się o standard SCORM, co umożliwia wykorzystanie istniejących aplikacji, które interpretują ten format. Należało go wzbogacić o metadane opisujące zawartość obiektów nauczania. W odróżnieniu do istniejących podejść do opisu semantycznego zaproponowano asocjacje pomiędzy zawartością i dziedziną wiedzy na podstawie ontologii dziedzinowej. Ostatnim elementem struktury jest lista kontaktów umożliwiająca późniejsze tworzenie grup kooperacyjnego nauczania. Tak zdefiniowana struktura obiektów nauczania powoduje, że ich zawartość może się zmieniać, co stanowi przeszkodę w użyciu systemów P2P do ich rozpowszechniania. Obecnie rozpowszechniane pliki P2P nie ulegają zmianom w czasie.

Proponowana w niniejszej pracy struktura obiektu nauczania składa się z kolejnych wersji. Każda z nich jest niezmiennym plikiem z punktu widzenia systemu rozpowszechniania P2P. Jednocześnie kolekcja niezmiennych wersji jest widziana jako jeden modyfikowalny obiekt nauczania dzięki unikalnemu identyfikatorowi.

Tak zdefiniowana reprezentacja obiektu wymagała stworzenia metody zapamiętywania relacji pomiędzy wersjami i obiektem w celu zidentyfikowania kolejnych wersji i pobrania ich z sieci P2P. Naturalnym

podejściem było wykorzystanie struktury DHT sieci P2P do zapamiętania relacji pomiędzy identyfikatorami wersji i identyfikatorem obiektu.

2. Następnym elementem proponowanych rozwiązań była koncepcja indeksacji obiektu nauczania. Indeksacja wykorzystuje metadane składające się ze standardowych atrybutów semantycznych pochodzących z analizy zawartości obiektu, jak również z pojęć zdefiniowanych w ontologiach dziedzinowych używanych do opisu semantycznego obiektu. Podobnie do zapamiętania relacji rządzących strukturą obiektu nauczania, indeksy zostają zapamiętywane w DHT jako pary kluczy i wartości. Kończącym elementem indeksacji obiektu nauczania jest dodanie informacji o identyfikatorze VoIP uczestnika wymiany obiektu, który umożliwia nawiązanie kontaktu i tworzenie kooperacyjnych grup zainteresowań.

Dane indeksacji pozwalają na efektywne wyszukiwanie obiektów. Użytkownik formułuje zapytanie i wysyła do DHT, tak aby odnaleźć obiekty, których metadane zawierają słowa kluczowe lub terminy związane z poszukiwaną tematyką. Ten sposób jednopoziomowego poszukiwania wymaga odczytania z DHT wszystkich par związanych z danym zapytaniem i ich redukcji w zależności od relacji logicznej zdefiniowanej w zapytaniu.

3. W pracy zaproponowano również rozszerzenie hierarchicznej metody wielopoziomowego poszukiwania obiektów w DHT i użycie ontologii jako podstawy do definicji zapytań o różnym poziomie szczegółowości. W przeciwieństwie do metody jednopoziomowej, indeksy nie zawierają identyfikatorów obiektów, lecz inne coraz bardziej wyspecjalizowane zapytania. Użytkownik rozpoczyna poszukiwanie od sformułowania

zapytania ogólnego i stopniowo odkrywa zapytania bardziej specjalizowane, aż do uzyskania jednoznacznej informacji o poszukiwanych obiektach, która pozwala na dostęp do nich.

4. Ostatnie części pracy dotyczą eksperymentalnej walidacji proponowanych rozwiązań poprzez implementacje prototypu i symulacje jego działania na szeroką skalę. Zadaniem prototypu pod nazwą GLEN (Global Lecture Exchange Network) było wykorzystanie popularnej sieci P2P BitTorrent i jego klientów do propagacji złożonych obiektów nauczania. GLEN rozszerza działanie klienta BitTorrent Azureus i używa zewnętrznego systemu OpenDHT do zapamiętywania atrybutów indeksacji i relacji pomiędzy identyfikatorami. GLEN oferuje użytkownikowi funkcje dodawania nowego obiektu nauczania w formacie SCORM 1.2 do systemu propagacji. Obiekt jest indeksowany słowami kluczowymi a użytkownik może wprowadzić dodatkowe indeksujące pojęcia z dziedzinowej ontologii. Wszystkie potrzebne atrybuty zapisywane są do OpenDHT. Użytkownik ma dostęp do funkcji poszukiwania obiektów za pomocą wprowadzania słów kluczowych lub wyboru pojęć z dziedzinowej ontologii. Ta metoda pozwala na dokładne sprecyzowanie zapytań dotyczących obiektów, jak również na progresywne odkrywanie obiektów począwszy od najbardziej ogólnych.

Prototyp GLEN pozwolił na walidację proponowanych rozwiązań z punktu widzenia funkcjonalnego. Implementacja klienta wykazała poprawność działania poprzez sprawdzenie w jaki sposób członek grupy kooperacyjnej może używać proponowanych funkcji oraz walidację funkcji indeksacji i wyszukiwania. Z tego punktu widzenia, prototyp zdał egzamin i pozwolił na otrzymanie skutecznie działającego narzędzia. Jego celem nie była jednak walidacja dla większej ilości obiektów i użytkowników ani szczegółowa ocena efektywności rozwiązań.

5. Dla szczegółowej oceny efektywności rozwiązań postanowiono użyć symulacji przy użyciu symulatora sieci OMNet++ i symulatora systemów P2P OverSim. Symulacje dla dużej ilości węzłów pokazały, że generacja znacznej ilości zapisów w DHT nie ma wpływu na średni czas zapisywania par danych do DHT. Podobnie poszukiwanie rezultatów wielu zapytań w DHT nie pociąga za sobą zwiększenia czasu oczekiwania. Rezultaty symulacji potwierdzają krytyczny wpływ czasu podczas którego węzły pozostają podłączone do sieci. Jeżeli ten czas jest za krótki, co oznacza, że sieć ma za duży stopień nowych podłączeń i odłączeń a efektywność indeksacji i wyszukiwania w DHT znacznie spada.

Warto dodać, że proponowane rozwiązania mogą służyć nie tylko do tworzenia kooperacyjnych grup nauczania, ale mogą również być wykorzystywane do powstawania grup społecznych na podstawie zainteresowania wspólną treścią lub zawartością. Można wspomnieć o grupach krytyki filmowej, muzyki lub literatury, grupach zainteresowanych przepisami kulinarnymi, grupach twórców pracujących nad wspólnym zagadnieniem (“crowdsourcing”) i wielu innych. Idea użycia propagacji danego obiektu do zapoczątkowania tworzenia grupy pozwala na późniejsze wykorzystanie innych narzędzi kooperacyjnych takich jak svn, trac, etc., które są już specyficzne dla działalności grupy.

Przyszłe badania skupią się na dalszych eksperymentach z prototypem i rozpatrzeniu możliwości tworzenia semantycznej sieci P2P.

Pierwszy aspekt dotyczy dopracowania prototypu GLEN, tak aby indeksacja i struktury danych niezbędne dla złożonego obiektu nauczania były zapisywane w DHT Azureusa. Uzyskany w ten sposób klient Azureusa byłby niezależny od systemów zewnętrznych i prowadziłby do możliwości przeprowadzenia testów z prawdziwymi użytkownikami. Oznacza to jednak, że należałoby używać specyficznej wersji Azureusa, co może nie być zaakceptowane

przez użytkowników, którzy chcą dysponować ostatnią wersją oryginalnego Azureusa.

Drugi aspekt polega na wykorzystaniu informacji o zawartościach obiektów nauczania do tworzenia semantycznych systemów P2P. W obecnych systemach P2P, działanie tablic DHT pociąga za sobą zapisywanie obiektów w sposób rozproszony – dwa obiekty o podobnej zawartości mogą znajdować się na dwóch różnych węzłach. W semantycznym systemie P2P, obiekty o podobnych zawartościach znalazłyby się na jednym węźle. W ten sposób użytkownik mógłby odkryć daną dziedzinę trafiając na węzeł, który jest wyspecjalizowany w danej tematyce. Problematyka ta związana jest również z wielopoziomową indeksacją semantyczną. Zapisywanie stopniowo specyficznych zapytań prowadzi do grupowania informacji o obiektach i możliwe, że pozwoli również na grupowanie obiektów o podobnej zawartości.

Bibliografia

- [1] K. Aberer and M. Hauswirth. An Overview on Peer-to-Peer Information Systems. In *Proceedings of the WDAS*, 2002. [cytowanie na str. 7]
- [2] Advanced Distributed Learning (ADL). Sharable Content Object Reference Model (SCORM). <http://www.adlnet.org>, 2004. [cytowanie na str. 5]
- [3] Apache XML RPC. <http://ws.apache.org/xmlrpc>. [cytowanie na str. 79, 115]
- [4] Azureus. An Open Source BitTorrent Client. <http://azureus.sourceforge.net/>. [cytowanie na str. 74]
- [5] Bamboo DHT. <http://www.bamboo-dht.org/>. [cytowanie na str. 76]
- [6] S. A. Baset and H. Schulzrinne. An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol. In *Proceedings of the INFOCOM*, 2006. [cytowanie na str. 10]
- [7] Ingmar Baumgart, Bernhard Heep, and Stephan Krause. OverSim: A Flexible Overlay Network Simulation Framework. In *Proceedings of 10th IEEE Global Internet Symposium (GI '07) in conjunction with IEEE INFOCOM 2007, Anchorage, AK, USA*, pages 79–84, May 2007. [cytowanie na str. 87]
- [8] Ingmar Baumgart, Bernhard Heep, and Stephan Krause. OverSim: A Scalable and Flexible Overlay Framework for Simulation And Real

- Network Applications. In *Proceedings of 9th International Conference on Peer-to-Peer Computing (IEEE P2P'09)*, 2009. [cytowanie na str. 87, 90, 119]
- [9] K.A. Berman and F.S. Annexstein. An Educational Tool for the 21st Century: Peer-to-peer Computing. In *Proceedings of the Ohio Learning Network Conference, Windows on the Future*, 2003. [cytowanie na str. 9, 30]
- [10] BitTorrent. <http://www.bittorrent.com>. [cytowanie na str. 37]
- [11] A. Bułkowski. Zastosowanie technologii peer-to-peer do kooperatywnego e-learningu. *E-mentor*, 17(5):22–25, 2006. [cytowanie na str. 11]
- [12] A. Bułkowski, E. Nawarecki, and A. Duda. Peer-to-Peer: an Enabling Technology for Next-Generation E-learning. In *Proceedings of the Fourth EDEN Research Workshop - Research Into Online Distance Education and E-Learning*, Barcelona, October 2006. [cytowanie na str. 57]
- [13] A. Bułkowski, E. Nawarecki, and A. Duda. Collaborative Learning for Everyone Based on Peer-to-Peer and Voice-over-IP Systems. In *Proceedings of Work With Computing Systems*, Stockholm, 2007. [cytowanie na str. 11]
- [14] A. Bułkowski, E. Nawarecki, and A. Duda. Peer-to-Peer Dissemination of Learning Objects for Creating Collaborative Learning Communities. In *Proceedings of ED-MEDIA*, Vienna, October 2008. [cytowanie na str. 71]
- [15] A. Bułkowski, E. Nawarecki, and A. Duda. Indexing and Searching Learning Objects in a Peer-to-Peer Network. In *Proceedings of IEEE EDUCON – Engineering Education 2010, The Future of Global Learning in Engineering Education*, Madrid, October 2010. [cytowanie na str. 60]

- [16] k. claffy, T. Monk, and D. McRobb. Internet Tomography. *Nature*, Jan 1999. [cytowanie na str. 91]
- [17] K. Curran. Peer-to-Peer Networking Collaboration within Education. *Journal of Educational Multimedia and Hypermedia*, 2002. [cytowanie na str. 32]
- [18] M. Dąbrowski. E-edukacja w SGH. *E-mentor*, 9(3), 2004. [cytowanie na str. 5, 18]
- [19] eXe. The eLearning XHTML Editor. <http://exelearning.org>, 2011. [cytowanie na str. 25]
- [20] P. Felber, E. Biersack, L. Garcés-Erice, K. Ross, and G. Urvoy-Keller. Data indexing and Querying in P2P DHT Networks. In *Proceedings of the International Conference on Distributed Computing Systems (ICDCS)*, Tokyo, 2004. [cytowanie na str. 58, 64]
- [21] H. Garcia-Molina and A. Crespo. Semantic Overlay Networks for P2P Systems. In *Stanford InfoLab, Technical Report*, 2003. [cytowanie na str. 54]
- [22] S. Guha, N. Daswani, and R. Jain. An Experimental Study of the Skype Peer-to-Peer VoIP System. In *Proceedings of the 5th International Workshop on Peer-to-Peer Systems (IPTPS)*, 2006. [cytowanie na str. 10]
- [23] IEEE TLSC. Draft Standard for Learning Object Metadata. http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf, 2002. [cytowanie na str. 5]
- [24] IMS Global Learning Consortium. IMS Learning Design Specification. <http://www.imsproject.org>, 3 2003. [cytowanie na str. 5]
- [25] Dublin Core Metadata Initiative. Dublin Core Metadata Element Set (DCMES) Version 1.1. Recommendation. <http://dublincore.org/>, 1999. [cytowanie na str. 25, 54]

- [26] Internet Engineering Task Force (IETF). SIP: Session Initiation Protocol. <http://tools.ietf.org/html/rfc3261>, 2011. [cytowanie na str. 71]
- [27] JAXB. Java Architecture for XML Binding. <http://jaxb.java.net>, 2011. [cytowanie na str. 78, 114]
- [28] JENA. A Semantic Web Framework for Java. <http://jena.sourceforge.net>, 2011. [cytowanie na str. 79, 115]
- [29] JXTA. <http://www.jxta.org/>. [cytowanie na str. 9, 33]
- [30] R. Katrys and K. Kempieński. *Zastosowanie systemu LMS w dydaktyce*. Instytut Elektroniki i Telekomunikacji Politechniki Poznańskiej, Poznań, 2005. [cytowanie na str. 21]
- [31] P. Mahadevan, D. Krioukov, M. Fomenkov, B. Huffaker, X. Dimitropoulos, k. claffy, and A. Vahdat. Lessons from Three Views of the Internet Topology: Technical Report. Technical report, Cooperative Association for Internet Data Analysis (CAIDA), Jun 2005. [cytowanie na str. 91]
- [32] S. Markusik and A. Bułkowski. Wykorzystanie możliwości e-learningu w kształceniu i doszkąłcaniu inżynierów. *E-mentor*, 10(3):29–33, 2005. [cytowanie na str. 17]
- [33] R.T. Mason and T.J. Ellis. Extending SCORM LOM. *Informing Science and Information Technology*, 6, 2005. [cytowanie na str. 22]
- [34] P. Maymounkov and D. Mazières. A Peer-to-peer Information System Based on the XOR Metric. In *Proceedings of the 1st International Workshop on Peer-to-Peer Systems (IPTPS '02)*, 2002. [cytowanie na str. 38, 45, 47, 118]
- [35] Steven McCanne and Sally Floyd. *The NS2 Network Simulator, Version 2.28*, 2005. [cytowanie na str. 87]

- [36] Alberto Montresor and Márk Jelasity. PeerSim: A scalable P2P simulator. In *Proc. of the 9th Int. Conference on Peer-to-Peer (P2P'09)*, pages 99–100, Seattle, WA, Sep 2009. [cytowanie na str. 87]
- [37] μ Torrent. <http://www.utorrent.com/>. [cytowanie na str. 74]
- [38] E. Nawarecki, G. Dobrowolski, S. Ciszewski, and M. Kisiel-Dorohinicki. Ontology of Cooperating Agents by Means of Knowledge Components. In *LNCS, Multi-Agent Systems and Applications III*, Prague, Czech Republic, 2003. [cytowanie na str. 11, 54]
- [39] W. Nejdl *et al.* Edutella: a P2P Networking Infrastructure Based on RDF. In *Proceedings of the 11th World Wide Web Conference*, 2002. [cytowanie na str. 9, 32]
- [40] OpenDHT. <http://www.opendht.org/faq.html>. [cytowanie na str. 76]
- [41] OPNET Technologies, Inc. <http://www.opnet.com>. [cytowanie na str. 87]
- [42] Planet Lab. <http://planet-lab.org/l>. [cytowanie na str. 76]
- [43] Qualnet 5.0.2. Scalable Networks. [cytowanie na str. 87]
- [44] RealNetworks Inc. Real Video Technical White Paper. <http://www.-realnetworks.com/>, 2011. [cytowanie na str. 4]
- [45] D.R. Rehak, P. Dodds, and L. Lannom. A Model and Infrastructure for Federal Repositories. In *Proceedings of the Interoperability Web-Based Educational System Workshop*, Chiba, Japan, 2003. [cytowanie na str. 54]
- [46] S. Rhea, B. Godfrey, B. Karp, J. Kubiawicz, S. Ratnasamy, S. Shenker, I. Stoica, and H. Yu. OpenDHT: A public DHT Service and Its Uses. In *Proceedings of SIGCOMM*, 2005. [cytowanie na str. 77]
- [47] M. Szeliga, S. Nieszwiec, R. Bachman, M. Kura, and T. Michalski. *Sieci P2P. Wymiana plików w internecie*. Helion, Gliwice, 2004. [cytowanie na str. 35]

- [48] J. Tane, C. Schmitz, G. Stumme, S. Staab, and R. Studer. The Courseware Watchdog: an Ontology-Based Tool for Finding and Organizing Learning Material. In *Mobiles Lernen und Forschen - Beiträge der Fachtagung an der Universität*, pages 93–104, November 2003. [cytowanie na str. 34]
- [49] Andras Varga. OMNeT++ User Manual – version 4.1. <http://www.omnetpp.org>, 2010. [cytowanie na str. 87]
- [50] Voip-Info.org. <http://www.voip-info.org/>. [cytowanie na str. 10]
- [51] Vuze. <http://www.vuze.com/>. [cytowanie na str. 74]
- [52] W. Nejd. Personalized Access to Distributed Learning Repositories–PADLR. In *WGLN Symposium*, 2002. [cytowanie na str. 34]
- [53] W3C. Resource Description Framework (RDF). <http://www.w3.org/RDF/>, 2004. [cytowanie na str. 9, 33]
- [54] W3C. Synchronized Multimedia Integration Language (SMIL 3.0). W3C Recommendation 01, <http://www.w3.org/TR/2008/REC-SMIL3-20081201>, 2011. [cytowanie na str. 4]
- [55] W3C. World Wide Web Consortium. <http://www.w3.org>, 2011. [cytowanie na str. 4]
- [56] K. Waćkowski and J. Chmielewski. Rola standaryzacji platform w e-learningu. *E-mentor*, 17(2), 2007. [cytowanie na str. 20]
- [57] D. Wiley. Learning Objects: Difficulties and Opportunities. http://www.opencontent.org/docs/lo_do.pdf, 2003. [cytowanie na str. 25]
- [58] XML RPC. <http://www.xmlrpc.com/spec>. [cytowanie na str. 79, 115]

Dodatki

Dodatek A

Opis prototypu GLEN

Niniejszy dodatek przedstawia szczegóły implementacji prototypu GLEN. Składa się on z następujących modułów, które wykonują następujące operacje.

A.1 Moduły przetwarzania

A.1.1 SCORM

Moduł zawierający funkcje do analizy formatu SCORM.

1. analiza syntaktyczna pakietu SCORM przy użyciu biblioteki JAXB (Java Architecture for XML Binding) [27],
2. tworzenie strukturalnej postaci drzewiastej pakietu SCORM,
3. analiza formatu LOM,
4. ekstrakcja metadanych,
5. konstrukcja obiektu LO na podstawie formatu SCORM.

A.1.2 LO

Moduł zarządzający obiektami nauczania. Opiera się na funkcjach modułów SCORM (analiza dodawanego obiektu), ONTOLOGIA (wybór pojęć do opisu semantycznego) i DOSTĘP (zapisanie informacji o dostępie do pliku torrenta).

1. dodawanie obiektów do systemu rozpowszechniania P2P,
2. edycja obiektów,
3. usuwanie obiektów
4. zapis indeksów do DHT,
5. wyszukiwanie obiektów odpowiadających na zapytania wprowadzane przez użytkownika,
6. agregacja rezultatów wyszukiwania.

A.1.3 ONTOLOGIA

Moduł zawierający procedury tworzenia i manipulacji ontologiami na podstawie biblioteki JENA (Semantic Web Framework for Java) [28].

1. konstrukcja ontologii w formacie RDF,
2. dostarczanie pojęć w fazie indeksacji obiektów i ich poszukiwania.

A.1.4 DHT

Moduł stanowiący interfejs pomiędzy Azureusem a OpenDHT.

1. dostarczenie operacji *Put()* i *Get()* na rozproszonej bazy danych,
2. dostęp do sieci OpenDHT poprzez pakiet zdalnego wywoływania procedur Apache XML RPC [58, 3].

A.1.5 DOSTĘP

Moduł związany z plikami torrent, które pozwalają na transfer danego obiektu. Łączy podstawowe funkcje dostępu do plików Azureusa z prototypem GLEN.

1. tworzenie plików torrenta,
2. edycja plików torrenta,
3. przesył obiektu na podstawie informacji zawartych w pliku torrent.

A.2 Interfejs użytkownika

Interfejs użytkownika rozszerza podstawowy interfejs Azureusa poprzez mechanizm wtyczek. Dostarcza funkcje do zarządzania okienkami, wyświetlania informacji i współpracy z użytkownikiem. Składa się z następujących modułów:

A.2.1 ADD

1. wprowadzanie złożonego obiektu nauczania do systemu,
2. analiza obiektu nauczania i ekstrakcja metadanych.

A.2.2 INDEX

1. wyświetlanie okienka indeksacji,
2. implementacja okienka dialogu z użytkownikiem do dodawania terminów indeksujących.

A.2.3 ONTOLOGY

1. wyświetlanie okienka ontologii,

-
2. implementacja okienka dialogu z użytkownikiem do wprowadzania pojęć z ontologii przy indeksacji obiektu i jego wyszukiwaniu.

A.2.4 SEARCH

1. wyświetlanie okienka odszukiwania,
2. implementacja okienka, który definiuje zapytania dotyczące obiektów nauczania.

A.2.5 RESULTS

1. agregacja rezultatów poszukiwania,
2. wyświetlanie rezultatów poszukiwania.

Spis rysunków

2.1	Struktura treści dydaktycznych na platformie www.e-sgh.pl . . .	19
2.2	LMS jako aplikacja bazodanowa działająca w oparciu o przeglądarkę WWW.	20
2.3	Schemat pakietu IMS/SCORM.	24
3.1	Zasada działania systemu Napster.	27
3.2	Zasada działania sieci Gnutella.	28
3.3	Zasada działania systemu FastTrack (Kazaa)	29
4.1	Zasada działania sieci BitTorrent z trackerem.	38
4.2	Pobieranie pliku w sieci BitTorrent.	39
4.3	Przykład pobierania w sieci BitTorrent.	42
4.4	Działanie sieci BitTorrent bez trackera.	43
4.5	Struktura drzewiasta Kademlii.	46
4.6	Prawdopodobieństwo pozostania kontaktu w sieci, przez kolejne 60 minut, w funkcji czasu przebywania w sieci w minutach. Źródło: [34]	47
4.7	Wyszukiwanie klucza w Kademlii.	49
5.1	Przykładowa zawartość obiektu nauczania.	53
5.2	Obiekt LO, jego wersje VSL0ID oraz relacje między nimi.	56
5.3	Przykład indeksacji obiektu.	58

5.4	Przykład prostej ontologii dziedzinowej.	60
5.5	Procedura wyszukiwania dla przykładowego zapytania.	63
5.6	Funkcjonowanie systemu P2P z komunikacją VoIP.	70
5.7	Zasada komunikacji VoIP	70
6.1	Zasada działania prototypu GLEN.	76
6.2	Schemat prototypu GLEN.	78
6.3	Struktura prototypu GLEN.	79
6.4	Struktura interfejsu użytkownika.	80
6.5	Konfiguracja Azureusa z wtyczką GLEN	81
6.6	Interfejs dodawania obiektu	82
6.7	Interfejs wyszukiwania obiektu	83
7.1	Schemat rozwiązania na bazie DHT Kedemlia i trackerless BitTorrent.	86
7.2	Struktura symulatora Oversim, źródło [8].	90
7.3	Średni czas odszukiwania obiektu, $N = 100$, $k = 8$, $M = 1$, słabe natężenie ruchu	93
7.4	Średni czas odszukiwania obiektu, $N = 100$, $k = 8$, $M = 1$, silne natężenie ruchu	94
7.5	Średni czas zapisywania pary danych do DHT, $N = 100$, $k = 8$, $M = 1$, słabe natężenie ruchu	95
7.6	Średni czas zapisywania pary danych do DHT, $N = 100$, $k = 8$, $M = 1$, silne natężenie ruchu	95
7.7	Średni czas odszukiwania obiektu, $N = 1000$, $k = 8$, $M = 1$, słabe natężenie ruchu	96
7.8	Średni czas zapisywania pary danych do DHT, $N = 1000$, $k = 8$, $M = 1$, słabe natężenie ruchu	97
7.9	Średni czas zapisywania M par danych do DHT, $N = 100$, $k = 8$	97
7.10	Średni czas zapisywania M par danych do DHT, $N = 1000$, $k = 8$	98
7.11	Średni czas wyszukiwania M par danych w DHT, $N = 100$, $k = 8$	98

7.12 Średni czas wyszukiwania M par danych w DHT, $N = 1000$, $k = 8$ 99

Wykaz terminów i skrótów

ADL, Advanced Distributed Learning, 5	JXTA, 9
Azureus, 15	Kademlia, 38
DHT, Distributed Hash Tables, 7	KBR, Key Based Routing, 91
Dublin Core Metadata Element Set, 25	LCMS, Learning Content Management System, 20
e-learning, 1	LMS, Learning Management System, 20
Edutella, 9, 32	LO, Learning Object, 55
GLEN, Global Lecture Exchange Network, 73	LOID, Learning Object Identifier, 55
GLEN, Global Lecture Exchange Network, 15	LOM, Learning Object Metadata, 5
Gnutella, 28	Napster, 27
Helpmate, 32	OMNet++, 15
IMS, Instructional Management Systems, 5	OpenDHT, 15, 76
JAXB, Java Architecture for XML Binding, 78	OverSim, 15
JENA, Semantic Web Framework for Java, 79	P2P, Peer-to-Peer, 2
	PADLR, Personalized Access to Distributed Learning Repositories, 34

PKM, Personal Knowledge Management, 30

RDF, Resource Description Framework, 9

RPC, Remote Procedure Call, 47

RTP, Real-time Transport Protocol, 71

SCO, Shareable Content Object, 23

SCORM, Sharable Courseware Object Reference Model, 5

SHA1, Secure Hash Algorithm, 40

SIP, Session Initiation Protocol, 71

Skype, 10

SMIL, Synchronized Multimedia Integration Language, 4

VoIP, Voice over IP, 3

VSLOID, Version Learning Object Identifier, 55

WBT, Web Based Training, 21

XML RPC, Remote Procedure Call z kodowaniem XML, 79