„System analizowania logów wyszukiwań dla bibliotek cyfrowych opartych na systemie dLibra”

Łukasz Nozdrzykowski

Szczecin, 2014

System analizowania logów wyszukiwań dla bibliotek cyfrowych opartych na systemie dLibra

Łukasz Nozdrzykowski[1]

1. Wprowadzenie

Budując repozytorium biblioteki cyfrowej należy zwrócić uwagę na możliwości rozwoju, w tym pozyskiwania informacji na temat poszukiwanych przez użytkowników końcowych, a zatem czytelników, potrzebnych im publikacji. W artykule przedstawiono propozycję systemu analizowania wyszukiwań publikacji w bibliotekach cyfrowych opartych na systemie dLibra w celu pozyskania informacji o potrzebach tych użytkowników. Pozyskane w ten sposób informacje mogą służyć do poszerzania oferty repozytorium biblioteki cyfrowej o nowe pozycje na które istnieje aktualne zapotrzebowanie.

2. Budowa systemu bibliotek cyfrowych dLibra

Oprogramowanie dLibra rozwijane jest przez Poznańskie Centrum Superkomputerowo-Sieciowe PCSS [1]. Oprogramowanie to służy do budowy bibliotek cyfrowych będących zorganizowanymi zasobami informacji oraz wiedzy z dodatkowymi funkcjonalnościami dostępnymi dzięki zastosowaniu technologii cyfrowej [2]. System dLibra pozwala na zautomatyzowane gromadzenie, opracowywanie i udostępniane danych w sieci komputerowej [3].

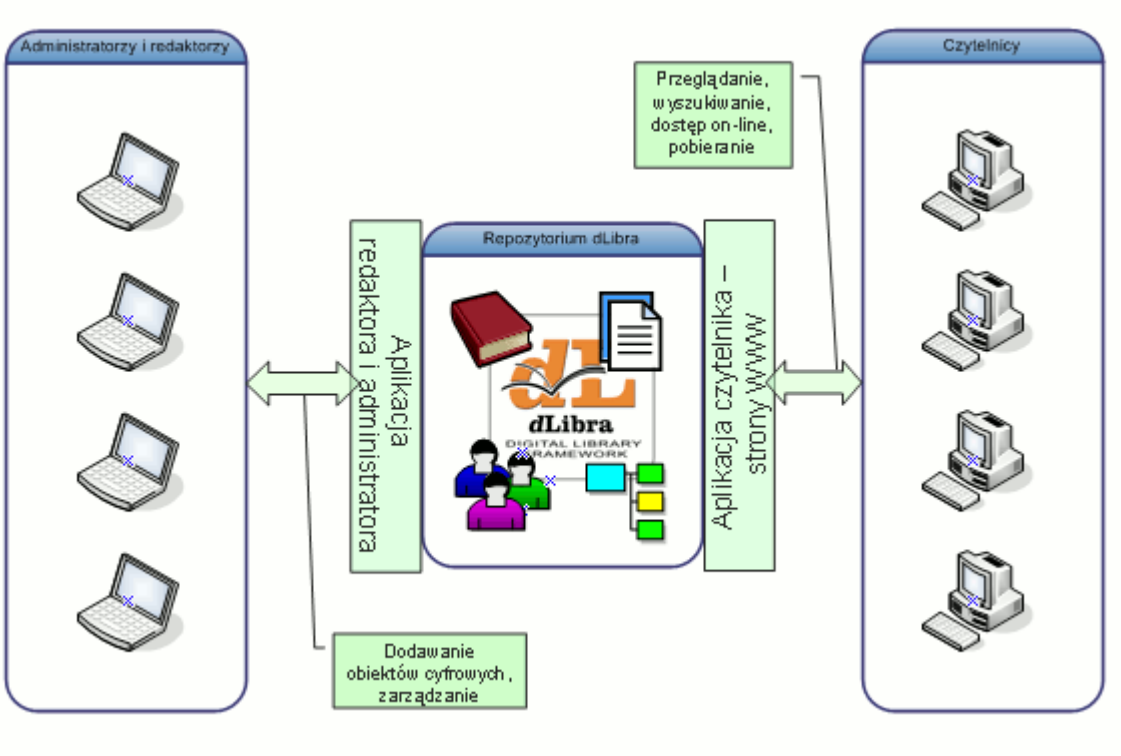
Głównymi możliwościami systemu bibliotecznego dLibra są [3]:

- kontrola dostępności zasobów na dwóch poziomach: użytkownika i zasobu,
- obsługa metadanych i danych przechowywanych w różnych formatach,
- rozbudowane mechanizmy wyszukiwania.

Sam system biblioteki cyfrowej dLibra opracowano z wykorzystaniem architektury sieciowej typu klient-serwer. Dodatkowe wykorzystanie budowy modularnej umożliwia realizowanie dostępu dla różnych grup użytkowników, którymi są administratorzy, redaktorzy oraz czytelnicy. Dla każdej z grup przygotowano odpowiednie aplikacje do ich obsługi, tj aplikacje administratora, redaktora oraz aplikacje czytelnika.

Nadrzędną grupą w systemie dLibra jest grupa administracyjna odpowiedzialna za konfigurację systemu biblioteki do potrzeb pozostałych grup użytkowników, w tym grupa ta odpowiedzialna jest za zarządzanie repozytorium. Za wprowadzanie obiektów do repozytorium biblioteki odpowiedzialna jest grupa redaktorów. Użytkownikami końcowymi, a zatem korzystającymi z umieszczonych w repozytorium obiektów, jest grupa czytelników mających do swej dyspozycji aplikację dostępną poprzez stronę WWW biblioteki. W centralnej części znajduje się tutaj serwer, do którego dostęp realizowany jest tylko poprzez aplikacje użytkowników. Serwer ten jest odpowiedzialny za realizację wszystkich funkcjonalności biblioteki.

[1] *dr inż. Łukasz Nozdrzykowski, l.nozdrzykowski@am.szczecin.pl, Instytut Technologii Morskich, Wydział Nawigacyjny, Akademia Morska w Szczecinie*



Rys. 1. Schemat budowy systemu dLibra [1]

Repozytorium systemu dLibra zbudowane jest w sposób hierarchiczny poprzez strukturę katalogów i danych w nich umieszczanych. Same zbiory w repozytorium są dzielone w sposób tematyczny, gdzie obiekty cyfrowe grupowane są względem posiadanych wspólnych cech. Takie podejście ułatwia finalnie zarządzanie umieszczanymi zasobami oraz prezentację powiązań pomiędzy nimi.

3. Biblioteka cyfrowa Akademii Morskiej w Szczecinie

Prezentowana w niniejszym artykule aplikacja przeznaczona do analizowania zapytań użytkowników do bazy systemu dLibra została przygotowana z myślą o rozwoju biblioteki cyfrowej Akademii Morskiej w Szczecinie wdrożonej w ramach projektu „Świat Morskich Publikacji" realizowanego w ramach Programu Operacyjnego Innowacyjna Gospodarka, Działanie 2.3 „Inwestycje związane z rozwojem infrastruktury informatycznej nauki" [4].

Głównym celem powstania tego repozytorium było opracowanie i udostępnienie największego zbioru publikacji pełnotekstowych z zakresu gospodarki morskiej dla użytkowników indywidualnych oraz instytucji związanych z tą branżą. Użytkownikami tymi są studenci, pracownicy naukowi i dydaktyczni, urzędy morskie, przedstawiciele branży stoczniowej, armatorzy, a także instytucje i firmy związane z eksploatacją portów i żeglugi [4].

Poprzez wdrożenie biblioteki cyfrowej w Akademii Morskiej umożliwiono wymianę wyników prac naukowo-badawczych, wymianę doświadczeń pomiędzy kadrą naukową i podmiotami gospodarczymi, a także podniesienie konkurencyjności i atrakcyjności Akademii Morskiej w Szczecinie na rynku jako partnera w projektach badawczych [4]. Zasoby Biblioteki Cyfrowej Świat Morskich Publikacji zostały podzielone na pięć kolekcji tematycznych [4]:

- Nauka i dydaktyka,

- Bazy, Portale, Linki,
- Marynistyka i Żeglarstwo,
- Z życia Akademii Morskiej w Szczecinie,
- Bibliografie.

W kolekcji biblioteki cyfrowej znajdują się teksty ciągłe, skrypty, podręczniki i materiały dydaktyczne, dorobek naukowy pracowników Akademii Morskiej w Szczecinie i innych uczelni związanych z gospodarską morską, materiały konferencyjne, doktoraty, artykuły z czasopism i artykuły zamawiane, adresy portali i stron internetowych związanych z branżą morską, bazy morskie, fotografie oraz odsyłacze internetowe do baz IMO i EMSA [5]. Dostęp do materiałów jest otwarty w ramach sieci uczelnianej lub sieci Internet zgodnie z ograniczeniami autorów publikacji lub właścicieli praw wydawniczych.

Od momentu powstania w 2010 roku do początku 2013 roku umieszczono w bibliotece łącznie 1564 publikacji, otworzono 34.555 publikacji, z czego poprzez wyszukiwanie wczytano 17.916 publikacji generując łącznie 2.830.545 stron.

4. System analizowania zapytań biblioteki cyfrowej dLibra

W celu zwiększenia konkurencyjności i atrakcyjności posiadanej biblioteki cyfrowej należy opracować system analizowania zachowań użytkowników w celu poznania tego czego użytkownicy poszukują. Poprzez pozyskanie informacji na temat publikacji poszukiwanych możliwe staje się dostarczenie do repozytorium nowych obiektów cieszącym się zainteresowaniem wśród czytelników.

Biblioteka cyfrowa Akademii Morskiej w Szczecinie wykorzystuje system dLibra w wersji 4.0. System ten został napisany z wykorzystaniem języka Java. Platformę serwerową stanowi tutaj serwer Apache Tomcat w połączeniu z systemem bazodanowym PostreSQL.

Twórcy systemu dLibra umożliwiają zapisywanie informacji na temat tego co użytkownicy poszukują. Dane zapisywane są w plikach logów dziennych w katalogu głównym serwera Apache Tomcat. Pliki te przechowywane są pod nazwami *query.log.xxxx-xx-xx*, przy czym pod znakami „x” ukrywana jest data pliku logu w postaci *rok-miesiąc-dzień*. W niniejszym artykule proponuje się wykorzystanie plików logów zapytań w celu budowy systemu analizy wyszukiwań użytkowników.

Sam system analizowania wyszukiwań został opracowany z użyciem języka PHP działającego po stronie serwera oraz w celu prezentacji wyników języków JavaScript z jQuery, a także HTML. System ten podzielono na dwie części. Pierwszą częścią jest podsystem przetwarzania plików logów w celu budowy bazy danych zapytań użytkowników. Drugi podsystem zajmuje się prezentacją wyników dla celów analitycznych.

4a. Podsystem przetwarzania plików logów

W celu zbudowania podsystemu przetwarzania plików logów należy dokonać analizy struktury tych plików w celu poznania zależności tej struktury z wyszukiwarką systemu dLibra.

System dLibra pozwala na wyszukiwanie zwykłe oraz wyszukiwanie zaawansowane. W wyszukiwaniu zwykłym możliwe jest wyszukiwanie we wszystkich możliwych treściach publikacji, wyszukiwanie tylko w tekście publikacji oraz wyszukiwanie w wybranej części publikacji, w tym w jej opisie czy polu autora. Wyszukiwanie zaawansowane umożliwia składanie zapytań logicznych AND/OR (oraz/lub), a także negacji NOT, domyślnie w trzech polach dotyczących publikacji, przy czym zapytanie można dynamicznie rozbudowywać o kolejne pole publikacji.

W plikach logów trzymane są informacje na temat daty i godziny wyszukiwania, a także rodzaju wyszukiwania oraz danych na temat poszukiwanych słów. Przykładowe zapytanie wygląda następująco:

2013-10-28 07:18:25,939 B034CC61C5C4ADB3B752E4410EF0C6D3-11
-3||english||||true||false||||

Na początku każdego wpisu znajduje się data oraz godzina wyszukiwania. Najważniejszym elementem jest numer przechowywany w drugiej linii. Numer ten oznacza rodzaj wyszukiwania. Następnym elementem jest wyszukiwana fraza. Numer -3 oznacza w prezentowanym przykładzie wyszukiwanie proste we wszystkich polach publikacji.

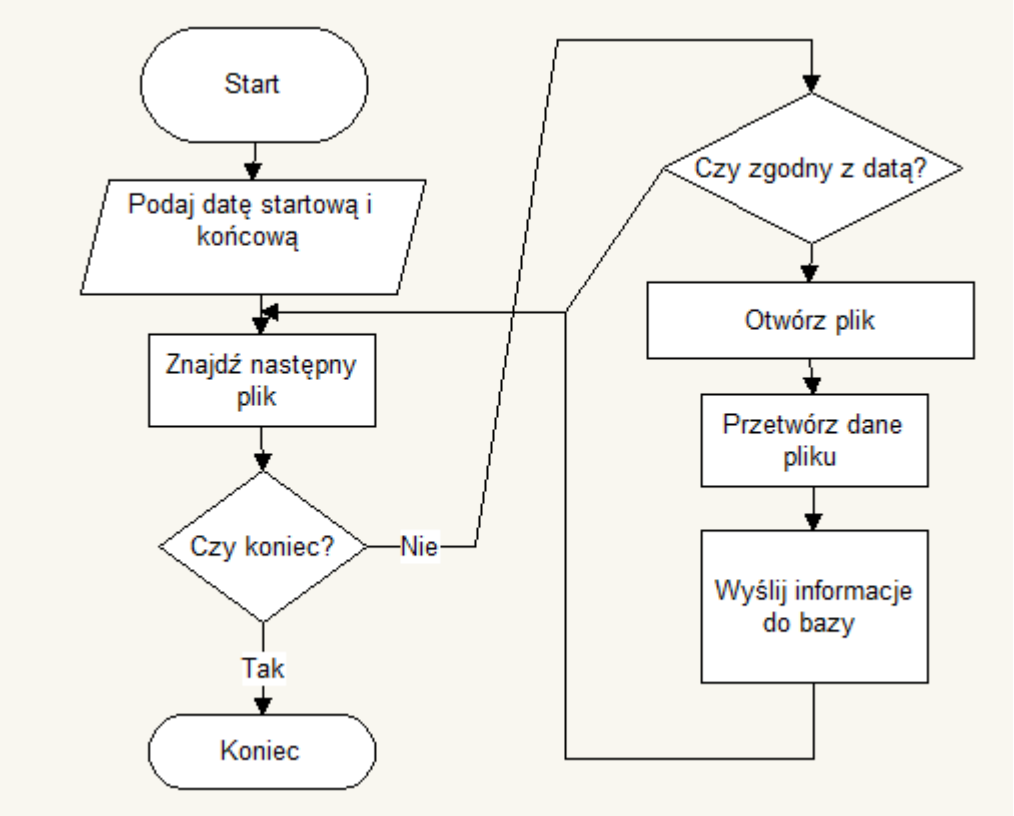
Proponowany w artykule system stosuje analizę plików dla wyszukiwania prostego. Przetwarzanie wyszukiwania zaawansowanego będzie realizowane w następnych etapach prac nad systemem.

W celu opracowania systemu analizy wyszukiwań należy przedstawić oznaczenia numeracji sposobów wyszukiwań stosowanych w systemie dLibra. Prezentuje to tabela 2.

Tabela 2. Kodowanie rodzaju wyszukiwań w plikach logów

Numer	**Znaczenie**
-1	Wyszukiwanie proste w tekście publikacji
-2	Wyszukiwanie proste w opisie publikacji
-3	Wyszukiwanie proste we wszystkich polach publikacji
-6	Wyszukiwanie proste w wybranym polu: Title – tytuł Creator - twórca Description – opis Publisher – wydawca Contributor – współtwórca Date – data Type – typ Format – format Identifier – identyfikator zasobu Source - źródło Language – język Relation – powiązania Coverage – zakres Rights – prawa Tags – tagi
-6	Wyszukiwanie zaawansowane za pomocą wyrażeń logicznych, przykład: *-4||||||true||false||||publikacja||AND||3||autor||AND||9||data||* gdzie w zapytaniu pojawia się informacja o wyrażeniu logicznym łączenia oraz o frazach szukanych w wybranym numerze pola dla wyszukiwania zaawansowanego

Podsystem przetwarzania plików logów został opracowany z wykorzystaniem języka PHP oraz frameworka jQuery. Całość przetwarzania danych z plików logów bazuje na algorytmie przedstawionym na rysunku 2.



Rysunek 2. Algorytm generowania bazy danych

Algorytm ten polega na wyszukiwaniu kolejnych plików logów, sprawdzaniu czy plik znajduje się w zakresie dat z podanego przez użytkownika preferencji oraz przetwarzaniu w pętli programowej danych z wybranych plików. Do wyszukiwania odpowiednich plików wykorzystuje się wyrażenie regularne:

'/query.log.[0-9]+\-[0-9]+\-[0-9]+/i'

Po znalezieniu pasującego do danego zakresu pliku następuje cykliczne przetwarzanie w pętli danych w nim zawartych w celu znalezienia daty zapytania oraz numeru operacji rodzaju zapytania i poszukiwanej frazy. Do tego celu można wykorzystywać wyrażenia regularne lub funkcję substr języka PHP. Uzyskując powyższe informacje należy sprawdzić czy w bazie danych przechowywane są już informacje o danej frazie i jaka była dotychczasowa liczba jej wyszukiwań. Zastosowano tutaj następujące zapytanie do tabeli *words* przechowującej informacje o wszystkich szukanych frazach:

SELECT count, date FROM words WHERE name='szukana fraza' AND date='$yf-$mf-$df'

Jeżeli szukana fraza (o nazwie wskazanej przez *name*) znajdowała się już w bazie pod wskazaną datą (*date*), wtedy licznik (*count*) jest zwiększany o jeden. W przeciwnym wypadku do bazy trafia informacja o tym, że szukana fraza była wyszukiwana jednokrotnie. Stosuje się tutaj zapytanie:

INSERT INTO words (id, name, count, date) VALUES (NULL, '$name', $count, '$yf-$mf-$df')

Jeżeli rekord już istnieje, wtedy należy zwiększyć jego liczbę (*count*) wystąpień poprzez zapytanie z wykorzystaniem identyfikatora *id* tego rekordu:

UPDATE words SET count=$count WHERE id=$id;

Dla zabezpieczenia podsystemu przetwarzania plików logów, podsystem ten został zabezpieczony systemem logowania z utrzymaniem sesji. Logowanie odbywa się z użyciem podania nazwy użytkownika i hasła, przy czym w systemie, a konkretnie w bazie danych, trzymana jest informacja o skrócie z hasła. Do wyliczania skrótu zastosowano funkcję skrótu MD5. Porównanie skrótu przechowywanego w bazie danych oraz skrótu wyliczonego na podstawie podanego przez użytkownika hasła umożliwia przeprowadzenie prawidłowego logowania. Zalogowanie z wynikiem pozytywnym umożliwia przeprowadzanie aktualizacji na bazie danych.

W celu utworzenia, odpowiedniej dla proponowanego systemu, bazy danych, należy w systemie zarządzania bazą danych wydać następujące zapytanie SQL:

```
CREATE TABLE IF NOT EXISTS `users` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `login` varchar(20) NOT NULL,
  `password` varchar(40) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;

CREATE TABLE IF NOT EXISTS `words` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` text COLLATE utf8_polish_ci NOT NULL,
  `count` int(11) NOT NULL,
  `date` date NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_polish_ci
AUTO_INCREMENT=1 ;
```

4b. Podsystem prezentowania wyników

Podsystem prezentowania wyników służy do opracowywania raportów dla celów analitycznych. Poprzez uzyskanie danych na temat poszukiwanych fraz odpowiadającym publikacjom otrzymuje się informacje na temat tego co użytkownicy wyszukują. W momencie nie znalezienia w systemie biblioteki cyfrowej wybranej publikacji, redaktorzy otrzymują dane na temat publikacji wartych pozyskania. To w efekcie powinno pozwolić na podniesienie atrakcyjności budowanego repozytorium biblioteki, szczególnie w przypadku częstych wyszukiwań.

Podsystem prezentacji wyników (podsystem raportowania) został opracowany z użyciem języka PHP oraz frameworku jQuery. Poprzez ten podsystem użytkownik redaktora otrzymuje wygodny interfejs w formie tabelarycznej z możliwością sortowania i przeszukiwania wyników. Stało się to możliwe dzięki przystosowaniu API jQuery-dataTables[6] do potrzeb podsystemu.

Podsystem ten pozwala na otrzymywanie raportu na temat liczby wszystkich odnotowanych wyszukiwań w systemie (*mysum*) w zadanym okresie czasu (*date*). Dane te są wyliczane na podstawie informacji przechowywanych w bazie danych w polu krotności (*count*). W celu pozyskania tych informacji zbudowano następujące zapytanie:

SELECT sum(count) as mysum FROM words WHERE date >= '$ys-$ms-$ds' AND date <= '$ye-$me-$de'

Tabela prezentacji wyników na temat wyszukiwań i ich krotności budowana jest na podstawie zapytania SQL z użyciem grupowania po szukanych frazach (*name*) i sortowana malejąco po liczbie wystąpień danej frazy (*mysum*). Służy do tego następujące zapytanie:

SELECT distinct(name) as myname, sum(count) as mysum FROM words WHERE date >= '$ys-$ms-$ds' AND date <= '$ye-$me-$de' GROUP BY name ORDER BY mysum DESC

Na rysunku 3 przedstawiono interfejs prezentacji wyników dla przykładowego okresu czasowego.

Rysunek 3. Interfejs główny proponowanego systemu

Poprzez zastosowaniu API jQuery.dataTables możliwe stało się szybkie sortowanie danych po wybranych kolumnach, wyszukiwanie po nazwie w sposób dynamiczny, a także zarządzanie liczbą wyświetlanych wyników.

5. Test proponowanego systemu w warunkach rzeczywistych

Proponowany system analizowania wyszukiwań został przetestowany z użyciem Biblioteki Cyfrowej Akademii Morskiej w Szczecinie. Okres testowy ustanowiono od początku działania biblioteki tj 09.11.2009 do 30.11.2013 roku. Analizowano tylko wyszukiwanie proste. Do testów posłużył serwer z procesorem Intel Core i7 z czterema rdzeniami taktowanymi zegarem 3.4 GHz.

Najbardziej złożonym obliczeniowo procesem jest generowanie bazy danych dla potrzeb systemu. Czas generowania bazy oraz dane o liczbie przetworzonych plików przedstawia tabela 2.

Tabela 2. Raport wydajności systemu dla przykładowej biblioteki

Liczba plików	Łączna wielkość plików	Wszystkich wyszukiwań	Wszystkich fraz	Unikalnych fraz	Czas generowania	Czas wyświetlania danych
1221	8.71 MB	40941	27.680	14 070	2 min 9 sek	3.81 sekundy

Ponieważ generowanie początkowe bazy danych może powodować zwiększony czas generowania zapytań, dlatego sugeruje się zwiększenie maksymalnego czasu wykonywania poleceń w pliku php.ini serwera (parametr max_execution_time), który w standardowym ustawieniu wynosi 30 sekund. W badanej bibliotece cyfrowej wygenerowano 1221 plików z logami wyszukiwań, które następnie poddano analizie. Ich łączna wielkość wynosi niecałe 9 MB co powoduje, że czas generowania bazy dla prezentowanego systemu wynosi ponad 2 minuty. Z tego też powodu czas wykonywania poleceń serwera powinien wynosić nawet około 300 sekund.

Prezentowana biblioteka była przeszukiwania z wyszukiwaniem prostym ponad 40 tysięcy razy generując ponad 27 tysięcy unikalnych fraz jakie były wyszukiwane. Pomimo tej liczby czas generowania wyników analizy wynosi niecałe 4 sekundy co pokazuje, że system ten działa w sposób bardzo szybki.

6. Podsumowanie

W artykule przedstawiono propozycję systemu analizowania plików logów wyszukiwań dla bibliotek cyfrowych opartych o system dLibra. Proponowany system dostarcza szybkich narzędzi analiz z sortowaniem fraz wyszukiwań wysyłanych do biblioteki cyfrowej. Dodatkową funkcją jest możliwość wyboru zakresu czasowego dla budowania baz danych do analiz jak i okresów samej analizy. Prezentowany system dostępny jest do pobrania pod adresem (adres do ustalenia).

Literatura:

[1] http://dlibra.psnc.pl/index.php?option=com_frontpage&Itemid=1 *(dostęp online: 2012)*
[2] Elżbieta Skubała, Anna Kazan: Polskie biblioteki cyfrowe na platformie dLibra – zasób w kontekście tworzenia nowoczesnych kolekcji źródeł informacji dla nauk technicznych. http://www.library.put.poznan.pl/konf_idn/art/1_3.pdf *(dostęp online: 2012)*
[3] Władysław Marek Kolasa: „dLibra" Digital Library Framework platforma do budowy bibliotek cyfrowych, http://*bbc.uw.edu.pl/Content/4/05.pdf (dostęp online: 2012*
[4] http://bc.am.szczecin.pl/biblioteka-cyfrowa-swiat-morskich-publikacji *(dostęp online: 2012)*
[5] http://www.smp.am.szczecin.pl/dlibra/text?id=example-page *(dostęp online: 2012)*
[6]. http://datatables.net/ *(dostęp online: 2013)*