



**УНИВЕРЗИТЕТ „ГОЦЕ ДЕЛЧЕВ“ - ШТИП  
ФАКУЛТЕТ ЗА ИНФОРМАТИКА**

---

---

ISSN:1857-8691

**ГОДИШЕН ЗБОРНИК  
2013  
YEARBOOK  
2013**

**ГОДИНА 2**

**VOLUME II**

---

---

**GOCE DELCEV UNIVERSITY - STIP  
FACULTY OF COMPUTER SCIENCE**

УНИВЕРЗИТЕТ „ГОЦЕ ДЕЛЧЕВ“ – ШТИП  
ФАКУЛТЕТ ЗА ИНФОРМАТИКА

---



**ГОДИШЕН ЗБОРНИК**  
**2013**  
**YEARBOOK**  
**2013**

ГОДИНА 2

МАРТ, 2014

VOLUME II

---

**GOCE DELCEV UNIVERSITY – STIP**  
**FACULTY OF COMPUTER SCIENCE**

**ГОДИШЕН ЗБОРНИК  
ФАКУЛТЕТ ЗА ИНФОРМАТИКА  
YEARBOOK  
FACULTY OF COMPUTER SCIENCE**

За издавачот:

**Проф д-р Владо Гичев**

**Издавачки совет**

Проф. д-р Саша Митрев  
Проф. д-р Лилјана Колева - Гудева  
Проф. д-р Владо Гичев  
Проф. д-р Цвета Мартиновска  
Проф. д-р Татајана Атанасова - Пачемска  
Доц. д-р Зоран Здравев  
Доц. д-р Александра Милева  
Доц. д-р Сашо Коцески  
Доц. д-р Наташа Коцеска  
Доц. д-р Зоран Утковски  
Доц. д-р Игор Стојановиќ  
Доц. д-р Благој Делипетров

**Редакциски одбор**

Проф. д-р Цвета Мартиновска  
Проф. д-р Татајана Атанасова - Пачемска  
Доц. д-р Наташа Коцеска  
Доц. д-р Зоран Утковски  
Доц. д-р Игор Стојановиќ  
Доц. д-р Александра Милева  
Доц. д-р Зоран Здравев

**Главен и одговорен уредник**

Доц. д-р Зоран Здравев

**Јазично уредување**

Даница Гавриловска - Атанасовска  
(македонски јазик)  
Павлинка Павлова-Митева  
(англиски јазик)

**Техничко уредување**

Славе Димитров  
Благој Михов

**Редакција и администрација**  
Универзитет „Гоце Делчев“ - Штип  
Факултет за информатика  
ул. „Крсте Мисирков“ 10-А  
п. фах 201, 2000 Штип  
Р. Македонија

**Editorial board**

Prof. Saša Mitrev, Ph.D.  
Prof. Liljana Koleva - Gudeva, Ph.D.  
Prof. Vlado Gicev, Ph.D.  
Prof. Cveta Martinovska, Ph.D.  
Prof. Tatjana Atanasova - Pacemska, Ph.D.  
Ass. Prof. Zoran Zdravev, Ph.D.  
Ass. Prof. Aleksandra Mileva, Ph.D.  
Ass. Prof. Saso Koceski, Ph.D.  
Ass. Prof. Natasa Koceska, Ph.D.  
Ass. Prof. Zoran Utkovski, Ph.D.  
Ass. Prof. Igor Stojanovik, Ph.D.  
Ass. Prof. Blagoj Delipetrov, Ph.D.

**Editorial staff**

Prof. Cveta Martinovska, Ph.D.  
Prof. Tatjana Atanasova - Pacemska, Ph.D.  
Ass. Prof. Natasa Koceska, Ph.D.  
Ass. Prof. Zoran Utkovski, Ph.D.  
Ass. Prof. Igor Stojanovik, Ph.D.  
Ass. Prof. Aleksandra Mileva, Ph.D.  
Ass. Prof. Zoran Zdravev, Ph.D.

**Managing/ Editor in chief**

Ass. Prof. Zoran Zdravev, Ph.D.

**Language editor**

Danica Gavrilovska-Atanasovska  
(macedonian language)  
Pavlinka Pavlova-Miteva  
(english language)

**Technical editor**

Slave Dimitrov  
Blagoj Mihov

**Address of the editorial office**

Goce Delcev University – Stip  
Faculty of Computer Science  
Krstе Misirkov 10-A  
PO box 201, 2000 Stip,  
R. of Macedonia

**СОДРЖИНА  
CONTENT**

<b>CALCULATION OF MULTI-STATE TWO TERMINAL RELIABILITY</b> Natasha Stojkovic, Limonka Lazarova and Marija Miteva .....	5
<b>INCREASING THE FLEXIBILITY AND APPLICATION OF THE B- SPLINE CURVE</b> Julijana Citkuseva, Aleksandra Stojanova, Elena Gelova .....	11
<b>WAVELET APPLICATION IN SOLVING ORDINARY DIFFERENTIAL EQUATIONS USING GALERKIN METHOD</b> Jasmina Veta Buralieva, Sanja Kostadinova and Katerina Hadzi-Velkova Saneva .....	17
<b>ПРОИЗВОДИ НА ДИСТРИБУЦИИ ВО КОЛОМБООВА АЛГЕБРА</b> Марија Митева, Билјана Јолевска-Тунеска, Лимонка Лазарова .....	27
<b>ПРИМЕНА НА МЕТОДОТ CRANK-NICOLSON ЗА РЕШАВАЊЕ НА ТОПЛИНСКИ РАВЕНКИ</b> Мирјана Коцалева, Владо Гичев .....	35
<b>S-BOXES – PARAMETERS, CHARACTERISTICS AND CLASSIFICATIONS</b> Dusan Bikov, Stefka Vouyuklieva and Aleksandra Stojanova .....	47
<b>ПРЕБАРУВАЊЕ ИНФОРМАЦИИ ВО ЕРП СИСТЕМИ: АРТАИИС СТУДИЈА НА СЛУЧАЈ</b> Ѓорѓи Гичев, Ана Паневска, Ивана Атанасова, Зоран Здравев, Цвета Маргиновска-Банде, Јован Пехчевски .....	53
<b>ЕДУКАТИВНО ПОДАТОЧНО РУДАРЕЊЕ СО MOODLE 2.4</b> Зоран Милевски, Зоран Здравев .....	65
<b>ПРЕГЛЕД НА ТЕХНИКИ ЗА ПРЕПОЗНАВАЊЕ НА ЛИК ОД ВИДЕО</b> Ана Љуботенска, Игор Стојановиќ .....	77
<b>ИНТЕРНЕТ АПЛИКАЦИЈА ЗА ОБРАБОТКА НА СЛИКИ СО МАТРИЧНИ ТРАНСФОРМАЦИИ</b> Иван Стојанов, Ана Љуботенска, Игор Стојановиќ, Зоран Здравев .....	85
<b>УТАУТ И НЕЈЗИНАТА ПРИМЕНА ВО ОБРАЗОВНА СРЕДИНА: ПРЕГЛЕД НА СОСТОЈБАТА</b> Мирјана Коцалева, Игор Стојановиќ, Зоран Здравев .....	95



## Пребарување информации во ЕРП системи: АртАИИС студија на случај

Горѓи Гичев<sup>1</sup>, Ана Паневска<sup>1</sup>, Ивана Атанасова<sup>1</sup>, Зоран Здравев<sup>2</sup>, Цвета Мартиновска-Банде<sup>2</sup>, Јован Пехчевски<sup>3</sup>

<sup>1</sup>Артисофт, Скопје, Македонија

(george.gicev, ana.panevska, ivana.atanasova)@artisoft.net

<sup>2</sup>Факултет за информатика, Универзитет „Гоце Делчев“ - Штип, Македонија  
(zoran.zdravev, cveta.martinovska)@ugd.edu.mk

<sup>3</sup>Факултет за информатика, Европски универзитет - Скопје, Македонија  
jovan.pehcevski@eurm.edu.mk

### Апстракт

Во процесот на раководење и водење на една компанија, богатството информации со кои располага компанијата, а кои укажуваат на нејзиното функционирање и развој и кои се употребуваат за подобрување на целосниот тек на работа, е од голема важност да биде постојано на допир и пристап до оној кому му е потребно. ЕРП системите кои денес претставуваат јадрото на информации на бизнисот на една компанија сè повеќе наоѓаат место во истражувачките кругови поради потребите за нивно подобрување и редефинирање на функционалниот аспект на достапност на податоците во системот.

Во овој труд е прикажана практична имплементација на пребарувачка машина за постоечки ЕРП систем наречен АртАИИС. Решението се обидува да воведо една нова насока во пребарувањето податоци во ЕРП системот АртАИИС, имено текст базирано пребарување, кое овозможува искористување на богатството на информации во компанискиот информациски систем, како за редовни така и за нередовни и необучени корисници за увид, бизнис аналитика и сл. Резултатите од нашата евалуација на стандардното филтер пребарување и пребарувачката машина за дадена тест колекција на информациски потреби покажуваат дека димензијата на текст ориентирано пребарување придонесува за полесна употреба на пребарувачка функционалност во ЕРП системот АртАИИС за сите типови на корисници.

**Клучни зборови:** пребарувачка машина, ЕРП системи, текст базирано пребарување, компаниско пребарување.

## 1 Вовед

Во светот на континуиран и зголемен проток на информации, постои евидентно зголемување на квантитетот на податоци потребни за анализа, бизнис увид и управување со информацијата во модерните интегрирани информациски системи. Денешниот свет веќе генерира околу 2.5 ексабајти (2.5x10<sup>8</sup>B) нови информации секој ден, што резултира со огромен импулс на решенија за Big Data (Halim, 2013).

Поради богатството на информации што ги има на веб, способноста да се пронајдат бараните информации со употреба на алатки за пребарување како Google заштедува многу време. Слични и поголеми предности, исто така, може да се добијат доколку пребарувањето на информации се ограничи во рамки на информациските системи на една компанија. Компаниска апликација, слично како и интернетот, поврзува компјутерски податоци низ целата компанија, обединувајќи различни типови информации во една единствена целина.

Како и во раните денови на интернетот, за да се пронајде информација во компаниската бизнис апликација потребно е да се знае каде истата се наоѓа. Овој метод

на пребарување претставува прифатлив начин за редовни корисници на системот за пребарување на информации за набавки, добавувачи или клиенти. Но, истиот не е применлив за повремени корисници на системот, па дури и за поредовен корисник кој пребарува информации низ апликацијата во област која не му е позната. Бидејќи компаниските апликации покриваат повеќе различни оддели и области на информации, тешко е корисниците на системот да имаат подетално познавање на функционалностите на целата апликација. Поради ова и апликациските вендори доаѓаат на пазарот со различни решенија за пребарување во нивните производи.

Постојат два различни приоди за овозможување на ваков вид на критично пребарувачка функција. Првиот приод се базира на компаниска апликациска алатка за пребарување која е цврсто интегрирана со компаниската апликација, и доставува пребарувачки резултати од апликацијата систем на знаење. Овој тип на приод е применет од бројни технолошки вендори, вклучувајќи го и Google, кој ја претстави својата Google Search Appliance и Google Mini Products (Gaudin, 2012). Дополнително, SAP и Oracle продолжуваат со маркетинг на своите пребарувачки алатки како засебни производи не само за своите апликации, туку и каде било на друго место во компанискиот интранет и бази на податоци.

Вториот приод на пребарување низ компанискиот информациски систем вклучува пребарувачка функција која е поцврсто интегрирана со специфична апликација, и истата претставува неопходна карактеристика која е потребно да се бара во една компаниска апликација. Само неколку апликациски вендори се фокусираат на доставување на функционалност на пребарување во рамки на нивната бизнис апликација која нуди пребарувачки способности слични на Google.

Денешните водечки светски софтверски вендори на ЕРП решенија сè повеќе започнаа со воведување на напредни пребарувачки машини во своите системи со цел подобрување на искуството на своите корисници. Дел од нив започнаа соработка со постоечки компании кои изработуваат вакви системи за интеграција на нивните апликации за поефикасно пребарување. Така Ziliak (2013), коосновач на xkzero, укажува дека според истражување спроведено од нив, корисниците на традиционалните ЕРП системи може да поминат и до 25% од своето време во пребарување на основни податоци низ своите системи. Нивниот систем GetX овозможува пребарување во стилот на Google за познатите Sage ЕРП системи преку пребарување на индексирани база на податоци (Ziliak, 2013).

Дополнително, сè повеќе компании ги препознаваат предностите од ваква надградба на својот ЕРП систем. Таков пример е компанијата EnPro (Frey, 2013). Компанијата решава да имплементира скалабилно и аналитичко решение кое ќе обезбеди преглед во корисничките интереси во реално време. Со помош на технологиите in-memory, овој приод овозможува податочно процесирање и аналитика во неколку секунди – искористувајќи ја машината на ЕРП системите од новите генерации.

Во овој труд детално е прикажана практичната имплементација на *артисофт пребарувачката машина*. Во продолжение на трудот е направен осврт на архитектурата, компонентите и нивната меѓусебна комуникација. Понатаму се опишани чекорите за имплементација на артисофт пребарувачка машина во ЕРП системот АртАИИС. На крај се прикажани резултатите од евалуацијата на двата типа на пребарување, филтер пребарување и артисофт пребарувачка машина и соодветно се дискутирани заклучоците и идните надградби и доработки на предложеното решение.

## 2. ЕРП пребарувачка машина во АртАИИС

Основната идеја на оваа практична имплементација беше да се развие сервисен и веб-ориентиран софтверски модул кој е интегриран, односно може да се користи како

пребарувачка машина во рамки на било која веб-базирана апликација и без разлика на документите т.е. содржините низ кои истата ќе пребарува.

Идејата се темели на два основни концепта:

- *Plug-n-play концепт на интегративност на корисничкиот интерфејс* – овој концепт подразбира дека пребарувачката машина како кориснички интерфејс за пребарување може да се имплементира врз која било веб-базирана апликација, независно од технологијата што се користи за развој на апликацијата во која се имплементира, како и од онаа за развој на пребарувачката машина.
- *Концепт на независност од колекција на документи* - овој концепт подразбира независност од видот на документи низ кои сакаме да пребаруваме, како и содржината на истите. Пребарувачката машина се полни, т.е. храни со податоци за овие документи преку јавно достапни веб-сервиси кои може да се повикаат од кој било софтвер независно од технологијата што се користи за развој на истиот. Од овој аспект, артисофт пребарувачката машина претставува сервисно-ориентиран продукт.

Артисофт пребарувачката машина претставува интегриран софтверски модул кој е комбинација од веб-апликација и множество на веб-сервиси за интерна комуникација во рамки на апликацијата, како и екстерни сервиси за комуникација со надворешни апликации. Токму екстерните сервиси се оние сегменти од решението кои овозможуваат имплементација на артисофт пребарувачката машина на која било веб-базирана апликација, т.е. систем. Апликацијата за потребите на индексирање и пребарување на документи го користи Lucene<sup>1</sup> - open-source множество од библиотеки кое овозможува поддршка за развој и имплементација на апликации за пребарување низ документи. Lucene, главно, се користи за пребарување низ текстуални содржини, иако има можност и за еден вид „имитација“ на релациона база на податоци. Целиот пакет е збир од Јава базирани библиотеки кои содржат моќни класи и функции за индексирање и пребарување на текстуални содржини.

### **2.1. Архитектура на артисофт пребарувачката машина**

Артисофт пребарувачката машина е продукт кој претставува комбинација од веб-апликација и множество од веб-сервиси за комуникација во рамки на истата, како и со надворешни системи врз кои истата би се интегрирала. Развојот на веб-апликацијата и веб-сервисите е доволно параметарски така што поддржува функционалност на различни платформи и технологии. Во продолжение ќе ги резимираме основните поими за пребарување информации и елементи на артисофт пребарувачката машина.

#### **Документ**

Документ е ентитет кој претставува логичка целина од содржина, автор, наслов, опис и сл. кој е од одредена вредност за дадена компанија, но истиот физички не мора да постои. Може да биде збир од записи во база на податоци, извештај добиен од некоја апликација, документ кој физички се чува во некој системи за менаџирање на документи или кој било електронски документ.

Секоја компанија може да чува повеќе документи од различни категории. Секој еден документ кој се индексира мора да се креира како Lucene Document објект или поточно објект од класата org.apache.lucene.document.Document. Полињата содржина, наслов, автор и краток опис се индексираат при градење на индексот. Сите полиња, освен полето

---

<sup>1</sup> <http://lucene.apache.org/>



содржина се чуваат во рамки на документ објектот, т.е. во индексот. Полето содржина не се чува поради фактот што индексот може да бара голема меморија за манипулација и да го забави процесот на пребарување. Сите останати информации се доволни за идентификација на документот и приказ на резултатите при пребарување. URL на документот може да биде мрежно достапна физичка локација на постоечки документ, URL до одреден документ во веб-апликација итн.

### **Стоп-зборови**

Стоп-зборови е множество од зборови за дадена компанија кои треба да се игнорираат при индексирање и пребарување. Секоја компанија го параметризира ова множество на зборови. Стоп-зборовите се најчесто сврзници, предлози и слични зборови кои не се многу значајни при пребарување. Од аспект на индексирање, ваквите зборови значително придонесуваат во намалување на големината на индексот.

### **Стоп-карактери**

Стоп-карактери е множество од специјални карактери кои се користат за поделба на даден текст на зборови. Иницијален стоп-карактер е празното место. Дополнително стоп-карактери може да бидат карактери како „.,\';:][!@#%&\*()\_+“ итн.

Мора да напоменеме дека колку и да изгледа едноставно мора да се посвети внимание при дефинирање на ваквите карактери, бидејќи истите може значајно да влијаат на пребарувањето. Пример за вакво сценарио е користење на “.” како стоп-карактер. На крајот од реченица таа е стоп-карактер, меѓутоа при индексирање на датумот 22.03.2013, термовите (терм, во понатамошниот текст збор) кои ќе се индексираат се “22”, “03” и “2013”.

### **Парсер**

Парсерот е ентитет кој во интерниот дизајн на решението се нарекува *тип на парсер*. Всушност, операцијата на парсирање се извршува преку Lucene објектот Analyzer, поточно објект од класата org.apache.lucene.analysis.standard.StandardAnalyzer. Тип на парсер како ентитет се користи заради дефинирање на различни комбинации на стоп-зборови и стоп-карактери при индексирање на различни видови документи.

### **Инвертиран индекс**

Инвертиран индекс претставува структура која чува статистики за множеството од зборови во документната колекција, како и листа од документи во кои даден збор се појавува. Целата комплексност на форматот на зачувување, заедно со сите информации и врски меѓу објектите кои се креираат при градење на индексот, е задача која ја извршува Lucene. Она што треба да се дефинира при генерирање на индексот е неговата физичка локација, односно директориумот во кој истиот ќе се чува. За Lucene индексот е збир од датотеки кои се запишуваат во еден директориум и се меѓусебно зависни. Оваа комплексност ја елиминираме така што индексот го нарекуваме директориум и ја знаеме локацијата на која треба да запишуваме и од која треба да читаме при пребарување.

## **2.2. Компоненти на артисофт пребарувачката машина**

Архитектурата на артисофт пребарувачката машина се состои од неколку компоненти: Document Crawler, Document Creator, Index Builder, Index и Searcher. Тие се објаснети како што следува.

*Document Crawler* – е компонента која комуницира со надворешни системи заради пронаоѓање на документи кои на одреден начин се означени дека треба да се

индексираат. Алгоритмот е креиран да ги помине рекурзивно сите директориуми на дадената локација и да ги собере сите пронајдени документи. Секој од овие документи се процесира и според форматот на истиот се повикува компонента која ја презема содржината на истиот во текст формат, како и други податоци од типот на наслов, автор и сл.

*Document Creator* - е компонента која директно може да се повика од кој било надворешен систем во вид на веб сервис. Се користи за обработка на документот во моментот на негово креирање, при што сервисот ја експортира содржината на истиот или пак ја добива во форма на текст како влезен аргумент. По процесирање на документот се повикува *Index Builder* компонентата која го индексира истиот.

Паралела на горниве две компоненти е веб crawler, робот компонента која го изминува вебот и ги индексира новите документи. Во случајот кога треба да се изминат веќе постоечки документи има потреба од дополнителни сетирања. Ова парче код го нарекуваме docProху и е прашање на имплементација на секој систем кој ќе ја користи пребарувачката машина.

*Index Builder* - оваа компонента игра една од клучните улоги во системот и се користи за индексирање на еден или група на документи. Се повикува од една од класите documentCrawler или documentCreator во Lucene кои како резултат даваат множество на документи. Всушност, задачата на оваа компонента е да запише еден или множество на документи во т.н. индекс на компанијата на која и припаѓаат документите.

*Searcher* - Searcher (Пребарувач) е компонента која се користи за пребарување. Оваа компонента како влезни аргументи прима идентификатор на компанијата и корисникот кој пребарува, како и клучните зборови. Нејзината задача е да пребарува низ директориумот кој се чува во RAM меморија, или пак низ главниот индекс, доколку не постои индекс во RAM меморија. Притоа се дефинира и по која од колоните кои се индексирани ќе се пребарува.

### **2.3. Практична имплементација на артисофт пребарувачката машина**

ЕРП системите на компаниите се централно место каде што се чуваат сите документи потребни за функционирањето на истите. Пребарувањето во ЕРП системите најчесто се врши со полиња филтри каде што корисникот е потребно да знае точна информација за идентификување на документот или пребарувањето резултира во голема листа на документи, подредени по нерелевантни фактори за оној кој пребарува.

АртАИИС е ЕРП системот на артисофт кој генерира 50-тина различни типови на документи. Пребарувањето се врши низ посебни функции за секој различен тип на документ со предефинирани филтри селектирани според проектантските анализи и одлуки при иницијалното проектирање на системот или пак при редицајн, секако со ограничувачки фактори. На овој начин не би можеле да се земат предвид сите можни сценарија, промени и потреби на корисниците, па во даден момент истиот е нефлексибилен и крут за корисниците.

Во продолжение се резимираани сите чекори потребни за имплементација на функционалност за индексирање и пребарување на документи од АртАИИС од тип *тикети* преку *артисофт пребарувачката машина*. Овие чекори се генерализирани и важат при имплементација на пребарувачката машина над која било колекција на документи од каква било категорија.

- 1) *Чекор 1: Постапување на компанија* – Првиот чекор е евиденција на компанијата која ќе ја користи *пребарувачката машина*.
- 2) *Чекор 2: Внес и мапирање на корисници* – За секоја компанија е потребно да се внесат

и мапираат корисниците на ЕРП системот кои би го користеле пребарувањето како функционалност. Секое пребарување се евидентира во историјат на пребарувања за корисникот кој пребарувал.

- 3) *Чекор 3: Конфигурација на системот* – Задолжителен чекор пред да се индексираат документите е конфигурација на начинот на процесирање на содржината на документите. Овој чекор опфаќа пополнување на стоп-карактери и стоп-зборови, како и креирање на потребните типови на парсери. Типот на парсер кој го креираме за целите на имплементација на Пребарувачката машина во АртАИИС е означен за default и ги вклучува стоп карактерите ^|,;:!"\$\*&., како и стоп зборовите *во, да, до, е, и, иако, итн, каде, како, на, но, од, по, покрај, пред, се, сл, т.е., што, итн.*
- 4) *Чекор 4: Полнење на индекс* – Веќе ги објаснивме компонентите *documentCrawler* и *documentCreator* кои се користат за обработка на документите кои треба да се индексираат и повикување на компонентата Index Builder која ќе ги индексира. Компонентата *documentCreator* е погодна доколку документите физички постојат на дадена мрежно достапна локација. Во нашиот случај документите (тикетите) се виртуелни, т.е. се комбинација од податоци кои се чуваат во релациона база на податоци. Во овој случај мора да креираме *docProху* компонента, т.е. парче код кое ќе ги креира документите кои сакаме да ги индексираме. Секој систем во кој се чуваат виртуелни документи треба да има можност за нивно печатење, при што документот се генерира во некаков отворен формат (најчесто pdf). Готовиот модул за печатење на документи се користи за нивно физичко креирање, по што пред да бидат избришани, се повикува компонентата која ќе ги индексира. Компонентата *documentCreator* може да се користи и за индексирање на сите новонастанати документи.
- 5) *Чекор 5: Интеграција на кориснички интерфејс* – Овој чекор подразбира воведување на можност за пребарување низ индексираните документи преку апликацијата на самиот корисник, во овој случај АртАИИС ЕРП системот.

### 3. Евалуација на ефективноста на артисофт пребарувачка машина

Во оваа секција се резимирани резултатите од евалуација и тестирање на практичната имплементација на артисофт пребарувачката машина во ЕРП системот АртАИИС. Методологијата за евалуација вклучува статистичка анализа и презентација на перформансите на двата системи преку добро познат сет од мерки за евалуација: Precision, Recall и MAP (Manning et al., 2009). За пресметка на овие мерки користен е TREC\_EVAL 8.1 софтверскиот пакет.

Најважните метрики за евалуација на ефективноста на еден пребарувачки систем се:

- RECALL – враќање на сите релевантни документи;
- PRECISION – враќање на најрелевантните документи на почеток од листата;
- КОМБИНАЦИЈА
  - враќање на што помалку нерелевантни документи;
  - враќање на релевантните документи пред нерелевантните.

Precision (P) е делот на вратени документи кои се релевантни

$$Precision = \frac{\#(\text{вратени релевантни документи})}{\#(\text{вратени документи})}$$

Recall (R) е делот на релевантни документи кои се вратени

$$Recall = \frac{\#(\text{вратени релевантни документи})}{\#(\text{релевантни документи})}$$

Mean Average Recall (MAP) единствена мерка за квалитетот на сите нивоа на recall:

$$MAP(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} Precision(R_{jk})$$

Дополнително, пребарувањето на информации низ компаниските информациски системи има различни потреби и специфики. Компаниските ЕРП системи се користат од различни типови на корисници, корисници кои се во секојдневна интеракција со системот и функционалности кои се блиски со обврските кои ги имаат на работното место и корисници кои не се доволно обучени да го користат системот или одреден дел од системот кој не е поврзан со нивните секојдневни обврски. Во евалуацијата што следува ќе ги земеме предвид двата типа на корисници.

### 3.1. Опис на тест колекција

Секоја тест колекција што се користи за евалуација на ефективност на информациски системи се состои од три главни компоненти:

- Колекција на документи, која е множество од документи што ќе се индексираат и врз кои ќе се извршува пребарување и мерење;
- Колекција на упити или пребарувања, т.е. тест множество од информациски потреби (пребарувања, query-ja);
- Множество на рачно идентификувани релевантни документи за секој упит или пребарување.

#### 3.1.1. Колекција на документи

Колекцијата на документи е множество од сите документи од АртАИИС од тип *тикети*. Тикет претставува документ кој се генерира за секоја активност во рамки на компанијата која подразбира интеракција со клиент на истата. Секоја пријава на проблем, барање за модификација, оперативна поддршка и сл. резултира со документ од овој тип во ЦРМ (CRM – Customer Relationship Management) модулот од АртАИИС. Тикетите содржат повеќе описни податоци внесени од страна на клиентите, како опис и манифестирање на проблем, како и информации кои се внесуваат од страна на вработените кои го разрешуваат проблемот, како што се причина за појава на проблем, мислење, опис на решение на проблемот и сл.

Во продолжение, во табела 1 се прикажани основни статистики за колекцијата на документи (тикети):

Табела 1. Статистики за колекцијата на документи

<b>Бр. на документи</b>	2908	<b>Бр. на зборови</b>	363166
<b>Просечен бр. на зборови во документ</b>	124.8	<b>Големи на индекс на хард диск</b>	1.12МВ
<b>Време на генерирање на док.</b>	60ms	<b>Време на индексирање на документ</b>	422ms

### 3.1.2. Тест множество на упити

Тест множеството од информациски потреби претставува збир од различни видови на упити или пребарувања (queries) кои се генерирани од експерти, односно лица кои секојдневно ги креираат или пак употребуваат на било каков начин документите од тип *тикет*.

Колекцијата содржи вкупно 25 пребарувања кои се поделени во две категории:

1. *Филтер, DB (Database)-ориентирани пребарувања* - Пребарувањата кои припаѓаат на оваа категорија може релативно лесно и точно да генерираат резултати од пребарувањето користејќи ги постоечките филтри за пребарување во ЕРП системот АртАИИС. Тикетот е збир од полиња кои се пополнуваат според шифрарници, односно еден вид на формулар, како и од полиња кои се текстуални. Доколку за пребарувањето е доволно пополнување на одредени филтер полиња или пак датумски полиња, тогаш пребарувањето припаѓа во оваа категорија. Оваа категорија содржи 14 од вкупно 25 пребарувања и се користи за моделирање на потребите на корисниците кои се во секојдневна интеракција со системот и функционалности кои се блиски со обврските кои ги имаат на работното место.
2. *Текст-ориентирани пребарувања* - Пребарувањата кои припаѓаат на оваа категорија, односно резултатите кои се очекуваат од истите не може да се добијат користејќи ги постоечките филтри за пребарување во ЕРП системот АртАИИС, односно системот или не поседува можност за пребарување според одредени услови или пак генерира огромна листа од резултати која не е подредена според никаква релевантност во однос на она што се пребарува. Овие пребарувања најчесто се комбинација од зборови кои може да се појават како дел од кое било од текстуалните, т.е. описните полиња на тикетот. Пример за такво пребарување е комбинација од неколку збора и датум или пак дел од датум. Оваа категорија содржи 11 од вкупно 25 пребарувања и се користи за моделирање на потребите на корисниците кои не се доволно обучени да го користат системот или одреден дел од системот кој не е поврзан со нивните секојдневни обврски.

### 3.1.3. Множество на релевантни документи

Последниот сегмент од тест колекцијата е множеството на релевантни документи за секој упит или пребарување. Резултатите од пребарувањата во двата система, како и листата на релевантни документи потоа се користи како влез во TREC\_EVAL.8.1 кој пресметува множество на метрики кои се користат за евалуација (оценување) и споредба на двата система.

Итеративно се поминуваат сите пребарувања и за секое се извршуваат следните чекори кои резултираат со добивање на множество од максимум 80 документи за евалуација за секое пребарување, како и вектор на релевантност за секој од овие 80 документи:

1. *Пребарување во пребарувачка машина* - Се извршува пребарување во артисофт пребарувачката машина со одреден сет на клучни зборови и се евидентира времето на пребарување, како и листа од идентификациски броеви на документите кои се добиваат како резултат сортирани според алгоритам за Scoring на документи на Lucene (ОкариBM25) за релевантност. Пребарувањето го извршува корисник кој не е многу искусен во работа со документи од тип *тикети*.
2. *Пребарување во АртАИИС* - Се извршува пребарување во АртАИИС со сет на филтри кои ги дефинира корисникот од информациската потреба на пребарувањето.

Се евидентира времето на пребарување, како и листа од идентификациски броеви на документите кои се добиваат како резултат сортирани според редослед на внес на документите во системот. Редоследот на приказ на даден документ е всушност неговиот ранг. Пребарувањето го извршува корисник кој не е многу искусен во работа со документи од тип *тикети*.

3. *Сет на резултати кои се повторуваат* – Од евидентираната листа на идентификациски броеви на тикети, како и нивните рангови, програмски се селектираат оние кои се повторуваат како резултат во пребарувањата извршени во првите два чекора. Доколку има 80 или пак помалку од 80 вакви документи сите стануваат дел од множеството на документи за евалуација на релевантност. Доколку има повеќе од 80 документи, се селектираат првите 80 сортирани според средната вредност на рангот на документот во двата система. Во овој случај се прескокнува чекорот 4.
4. *Сет на резултати кои не се повторуваат* – Доколку во претходниот чекор не се формира колекција од 80 документи, истите се дополнуваат со документи кои се појавуваат како резултати само во еден од системите. Распределбата е подеднаква.
5. *Оценување на релевантност на пребарувања* – Во овој чекор се избираат експерти, односно лица кои ја познаваат проблематиката и кои ќе ја оценуваат релевантноста на секој од документите во колекцијата на документи за евалуација на даденото пребарување. Секој од документите бинарно се означува дали е релевантен за моменталното пребарување (со 1 или 0). Оваа листа на вредности го формира векторот на релевантност.
6. *Генерирање на влез во TREC\_EVAL* – Од генерираните листи на резултати, придружени со нивните рангови, како и колекцијата на документи за евалуација и векторот на релевантност, програмски се генерираат текстуални документи кои потоа претставуваат влезни аргументи во програмата за евалуација. При ваквата обработка на резултатите се добиваат и одредени статистички за тест колекцијата.

Во продолжение се прикажани општи статистички податоци за тест колекцијата за секоја категорија на пребарувања (табела 2 и табела 3).

Табела 2. Статистички податоци за тест колекцијата за категорија *текст*

Резултати	Максимален бр.	Минимален бр.	Просечен бр.
Артисофт пребарувачка машина	2899	22	1474
АртАИИС	2445	0	458
Релевантни док. во колекција на док. за евалуација	80	0	40

Табела 3. Статистички податоци за тест колекцијата за категорија *DB*

Резултати	Максимален бр.	Минимален бр.	Просечен бр.
Артисофт пребарувачка машина	2894	25	1394
АртАИИС	2445	0	660
Релевантни док. во колекција на док. за евалуација	80	7	29

### 3.2. Експериментални резултати

Во продолжение се прикажани резултатите, табеларно (табела 4 и табела 5) и графички (слика 1 и слика 2) од евалуација на двата система за пребарување со двете категории, DB базирани и текст базирани пребарувања.

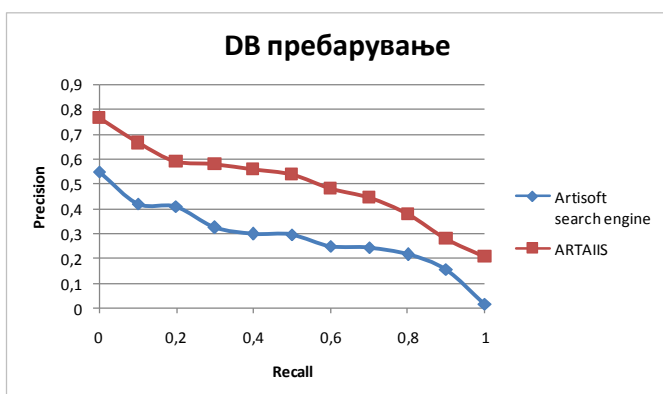
Табела 4. Резултати од пребарување со DB базирани пребарувања

DB БАЗИРАНО		DB БАЗИРАНО	
Артисофт пребарувачка машина		АртАИИС	
Метрика	Вредност	Метрика	Вредност
P5	0.27	P5	0.43
P10	0.26	P10	0.44
P15	0.25	P15	0.42
P20	0.26	P20	0.42
P30	0.27	P30	0.42
P100	0.28	P100	0.41
MAP	0.25	MAP	0.45
AP	0.05	AP	0.19

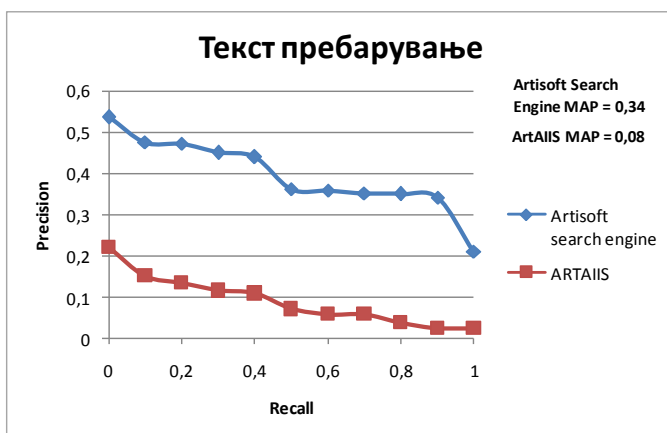
Табела 5. Резултати од пребарување со текст базирани пребарувања

ТЕКСТ БАЗИРАНО		ТЕКСТ БАЗИРАНО	
Артисофт пребарувачка машина		АртАИИС	
Метрика	Вредност	Метрика	Вредност
P5	0.36	P5	0.11
P10	0.32	P10	0.07
P15	0.31	P15	0.08
P20	0.30	P20	0.08
P30	0.25	P30	0.08
P100	0.27	P100	0.08
MAP	0.34	MAP	0.08
AP	0.22	AP	0.01

Резултатите покажуваат дека артисофт пребарувачката машина е значително поефективна при текст-базирани пребарувања (табела 5 и слика 2), додека очекувано стандардното филтер пребарување во АртАИИС ЕРП системот се покажа како поефективно во пребарувањето на информации поврзани со дадени точно познати вредности за атрибутите на документот кој се бара (табела 4 и слика 1). Секако, вториот случај подразбира детално познавање на начинот на кој функционира целиот систем, додека првиот е прикладен за корисници кои воопшто не се запознаени со спецификите на системот и се научени да пребаруваат информации користејќи стандардни методи на веб (или Google-like) пребарување.



Слика 1. Резултати Precision и Recall од пребарување со DB базирани пребарувања



Слика 2. Резултати Precision и Recall од пребарување со текст базирани пребарувања

#### 4. Заклучок и идна работа

Во овој труд прикажавме практична имплементација на пребарувачка машина за постоечки ЕРП систем наречен АртАИИС. Преку експериментални резултати покажавме дека без разлика колку и да му е добро познат системот на корисникот кој пребарува, времето потребно да се навигира до одреден кориснички екран за пребарување низ даден тип на документи, како и времето потребно да се пополнат сите филтри, не смее да се запостави. Придонесот од имплементација на пребарувачка машина врз кое било парче софтвер е во можноста за брзо и ефективно пребарување низ множество на документи и текстуални содржини од различни типови, користејќи единствен и едноставен кориснички интерфејс.

Во иднина би можело да се направат неколку дополнителни оптимизации на пребарувачката машина во поглед на претпроцесирањето на документите, како и во алгоритмот за рангирање на истите. Поддршка на латинично и кирилично пребарување е исто така потребна идна надградба на пребарувачката машина, со цел да се овозможи намалено време за доаѓање до потребната информација преку олеснување на корисничката интеракција со системот.



## Литература

- [1] Frey, L, (2013). Reinventing the ERP Engine. Tech Trends 2013 – Enablers  
<http://www.deloitte.com/assets/Dcom-UnitedKingdom/Local%20Assets/Documents/Services/Consulting/uk-con-reinventing-erp-engine.pdf>
- [2] Gaudin, S, (2012). Google upgrades Search Appliance for enterprises. ComputerWorld,  
[http://www.computerworld.com/s/article/9232198/Google\\_upgrades\\_Search\\_Appliance\\_for\\_enterprises](http://www.computerworld.com/s/article/9232198/Google_upgrades_Search_Appliance_for_enterprises)
- [3] Halim, N. 2013, Stream processing. Director and Chief Architect of Big Data, IBM. прегледано на 20.11.2013 <http://www.ibm.com/smarter-computing/us/en/technical-breakthroughs/stream-processing.html>
- [4] Manning, C, D, Raghavan, P, Schütze, H, (2009). *An Introduction to Information Retrieval*, Cambridge University Press
- [6] Ziliak, P, (2013). Powerful ERP search tool now available for Sage 100 ERP v2013.  
<http://www.prweb.com/releases/2013/2/prweb10478394.htm>, прегледано на 10.12.2013

