МНОГОКРИТЕРИАЛЬНОЕ ОЦЕНИВАНИЕ И ВЫБОР СТРУКТУРЫ БД В САПР MONGODB

УДК 004.4'22: 519.816

КАЛИТА Надежда Ивановна
к.т.н., доцент, доцент кафедры системотехники Харьковского национального университета радиоэлектроники.
Научные интересы: теория принятия решений, технологии проектирования информационных систем.
e-mail: kalita.nik@gmail.com

СТРЕЛЬЧЕНКО Вадим Вадимович
магистрант кафедры системотехники Харьковского национального университета радиоэлектроники.
Научные интересы: технологии проектирования информационных систем.
e-mail: bowlingforsoap@gmail.com

ВВЕДЕНИЕ

К настоящему времени существуют уже хорошо отработанные механизмы и подходы к проектированию структур реляционных баз данных (БД), которые долгое время считались канонической формой объективного представления информации. Альтернативной реляционному подходу является достаточно молодая и быстро эволюционирующая ветвь NoSQL (Not Only SQL) БД, которые объединяют в себе базы данных объектные, документные, «ключ-значение», на основе графов и т.п. Одной из самых популярных СУБД документных NoSQL БД у разработчиков информационных систем является MongoDB [1]. И, хотя и имеется довольно обширный набор практик и рекомендаций по проектированию ее структуры, единый формализованный подход проектирования отсутствует, поэтому разработка информационной технологии проектирования MongoDB является перспективным направлением.

ПОСТАНОВКА ЗАДАЧИ

Основная особенность MongoDB – это её динамическая структура данных. База данных состоит из коллекций (от англ., *collection*), которые представляются пользователю в виде JSON (*JavaScript Object Notation*)-документа. Динамическая структура позволяет разным документам одной коллекции содержать разные наборы полей или структуру, а одинаковые поля могут содержать разные типы данных. С одной стороны, такая особенность предоставляет гибкость при проектировании, с другой – от компетентности проектировщика в значительной степени зависит конечное качество спроектированной БД.

Качество структуры БД является комплексным и в какой-то степени субъективным показателем, для определения которого не существует универсальных методов. В отличие от реляционных БД, которые проектируются в соответствии с заложенным в них математическим аппаратом, структура MongoDB, предоставляющая большую гибкость, привносит в процесс проектирования большую долю эвристики.

В рамках САПР MongoDB первым этапом проектирования является автоматизированный синтез вариантов структур, и в [2, 3] предложена информационная технология их синтеза на основе XMI-документа или другого способа представления информации о предметной области в структурированном виде. Следующие этапы проектирования – автоматизированное оценивание качества сгенерированных структур и многокритериальный выбор наилучшего варианта отображения данных о предметной области [4].

Целью работы является разработка методов автоматизированного оценивания структур БД MongoDB и многокритериального выбора из них наилучшей.

Под оценкой структур понимается оценка качества каждой из них, т.е. определение степени их соответствия поставленным задачам. В случае с реляционными БД, достаточно привести логическую структуру к 3-й нормальной форме (3НФ), чтобы она могла называться «нормализованной» [5]. В отличие от реляционных баз данных в MongoDB создание структуры общего назначения заменяется подстройкой структуры базы данных на высокую производительность для определенного множества запросов. Кроме этого также могут иметь место аномалии модификаций данных, поскольку их размещение в одном документе позволяет быстро получить к ним доступ, а в случае модификации будет произведена транзакция, так как MongoDB поддерживает атомарность операций только на уровне одного документа.

На первом этапе синтеза структур MongoDB [3] каждая структура отслеживается на наличие таких аспектов, как: аномалии модификации, атомарность доступа к парам объектов и доступ к избыточным данным. Такая информация помогает оценить качество структур, но не является достаточной для выбора наилучшей. Необходимым является определение степени соответствия структур поставленным задачам в смысле наилучшей производительности, что может быть осуществлено с помощью нефункционального, а именно нагрузочного, тестирования БД по конкретным сценариям [6]. Из множества программных продуктов автоматизированного нагрузочного тестирования выбрано Apache JMeter [7], как средство с открытым кодом на языке Java, что обеспечивает совместимость с программными модулями синтеза структур БД и может быть интегрировано в САПР с настройкой под MongoDB.

РАЗРАБОТКА МЕТОДА РЕШЕНИЯ ЗАДАЧИ

При проектировании информационной системы у системного архитектора есть конкретные требования к данным, которые должны храниться в БД: это список полей по различным сущностям (объектам), которые можно объединить в различные структуры БД. Пусть необходимо хранить p полей в БД. В крайних случаях синтезированная структура будет содержать 1 или p коллекций – все поля в одной коллекции и по одной коллекции на поле, соответственно. Безусловно, такие случаи не представляют практического интереса, так как в редких случаях могут обеспечить конкурентную производительность. Задача анализа связей решается на первом этапе проектирования MongoDB [3], выходной информацией которого является синтезированное множество допустимых структур (коллекций) MongoDB $X = \{x_j\}$, $j = \overline{1, N}$. Количество экземпляров коллекций и максимальное количество структурных элементов в коллекции зависит от предметной области.

Будем полагать, что каждый вариант структуры БД $x \in X$ характеризуется некоторым n-мерным набором параметров (частных критериев) $k_i(x)$, $i = \overline{1, n}$, обусловливающих ее функциональность и производительность при выполнении различных типов операций с данными. Автоматизированное тестирование структур БД с помощью Apache JMeter позволяет получить количественные оценки основных временных параметров, характеризующих скорость записи, скорость чтения, скорость обновления, скорость удаления, скорость агрегации.

Для разработчика БД, который является лицом, принимающим решение (ЛПР), каждый частный критерий $k_i(x)$ имеет определенную относительную важность λ_i, которая зависит от множества ситуационных факторов и индивидуальных особенностей ЛПР. Тогда наилучшая структура БД MongoDB x^o на множестве оцениваемых вариантов X определяется как:

$$x^o = \arg \operatorname*{extr}_{x \in X} F[\lambda_i, k_i(x)], \quad (1)$$

где F – оператор, определяющий вид зависимости.

Один из подходов преобразования многокритериальной задачи принятия решения к однокритериальной состоит в использовании функции полезности на основе аддитивной, мультипликативной функций или их комбинаций [8]. С учетом известных достоинств и недостатков этих функций в качестве оператора F целесообразно использовать аддитивную функцию полезности вида

$$P(x) = \sum_{i=1}^{n} \lambda_i k_i(x). \quad (2)$$

Перейдя к нормализованным, т.е. приведенным к изоморфному виду значениям частных критериев $p_i[k_i(x)] = \left(\frac{k_i(x) - k_{i\text{нх}}}{k_{i\text{нл}} - k_{i\text{нх}}} \right)^{\alpha_i}$, где $k_{i\text{нх}}$, $k_{i\text{нл}}$ – соответственно наихудшее и наилучшее значение i-го частного критерия, α_i – параметр нелинейности, реализующий при $\alpha_i = 1$ линейную зависимость, при $\alpha_i < 1$ – выпуклую вверх зависимость, при $\alpha_i > 1$ – выпуклую вниз зависимость, модель выбора наилучшей структуры (1) примет вид:

$$x^o = \arg\max_{x \in X} \sum_{i=1}^{n} a_i \, p_i[k_i(x)], \qquad (3)$$

где a_i – безразмерные весовые коэффициенты относительной важности частных критериев, $0 \le a_i \le 1$, $\sum_{i=1}^{n} a_i = 1$.

При выборе наилучшей структуры (3) возможны следующие ситуации:

1) имеется информация о предпочтениях ЛПР к конкретным типам операций с БД, и тогда a_i можно присвоить точные количественные значения;

2) частные критерии $k_i(x)$, $i = \overline{1,n}$, упорядочены по важности $k_1(x) \succ k_2(x) \succ \ldots \succ k_n(x)$ и точные значения a_i неизвестны;

3) информации о предпочтениях ЛПР нет и, соответственно нет информации о значениях a_i.

В рамках САПР MongoDB целесообразно использовать единую универсальную адаптивную модель оценивания и оптимизации [8]:

$$x^o = \arg\max_{x \in X} \left\{ \sum_{i=1}^{n} a_i \, p_i[k_i(x)]^{\beta} \right\}^{\frac{1}{\beta}}, \qquad (4)$$

где β – адаптационный параметр, позволяющий реализовать конкретную информационную ситуацию.

В первом случае естественно обобщенную полезность варианта структуры $x \in X$ определить как аддитивную функцию вида (3) при выполнении указанных условий, а наилучшую структуру x^o определить по модели (4) при $\beta = 1$.

Во втором случае качественная информация о важности критериев наиболее полно используется при схеме оптимизации по последовательно применяемым критериям, когда поиск наилучшего решения проводится последовательно по каждому критерию путем решения однокритериальных оптимизационных задач. В универсальной модели (4) необходимо положить $\beta = 1$, и схема решения состоит в следующем. На первом шаге $a_1 = 1$, и из исходного множества допустимых решений X выделяется подмножество x_1^o решений оптимальных по первому (наиболее важному) критерию. Для этого решается однокритериальная оптимизационная задача вида

$$x_1^0 = \arg\max_{x \in X} p_1[k_1(x)]. \qquad (5)$$

Если множество x_1^o содержит более одного решения – переходим к следующему этапу, т. е. решаем задачу выбора оптимальных решений по второму по важности критерию, но уже из множества x_1^o и при $a_2 = 1$:

$$x_2^o = \arg\max_{x \in x_1^o} p_2[k_2(x)]. \qquad (6)$$

Оптимизация продолжается до тех пор, пока на i-м шаге не будет получено единственное решение или не исчерпаются все критерии.

В третьей ситуации используются схемы равенства или квазиравенства важности критериев. Предполагается, что все коэффициенты равны между собой, т.е. $a_i = 1/n$, $i = \overline{1,n}$. Тогда на основе (4)

$$x^o = \arg\max_{x \in X} \frac{1}{n} \sum_{i=1}^{n} p_i[k_i(x)]. \qquad (7)$$

РЕШЕНИЕ ЗАДАЧИ

Входными данными для автоматизированного оценивания средствами JMeter является множество допустимых структур БД X и наборы тестов – оценочных сценариев AS_l, $l = \overline{1,L}$. Структуры могут быть заданы вручную, экспортированы или получены автоматизированным способом на основе разработанной технологии синтеза схем БД для MongoDB [3]. Различные структуры БД, содержащие одинаковый набор полей, а также одни и те же структуры БД с различными параметрами работы и запуска могут быть оценены одними и теми же AS_l, что позволяет получить их объективную оценку. Под параметрами работы и запуска БД понимаются: настройки среды, свойства БД и коллекций,

настройки репликации и шардинга. Таким образом, по каждому частному критерию $k_i(x), i = \overline{1,n}$, может быть получено множество из m_i значений $\{k_{iv}(x)\}$, $v = \overline{1,m_i}$, которые формируются в JMeter как средние в соответствии с центральной предельной теоремой, поэтому считается, что они нормально распределены [9]. Тогда в качестве оценки исследуемого временного параметра можем взять среднее значение $K_i(x) = \frac{1}{m_i}\sum_{v=1}^{m_i} k_{iv}(x)$, $i = \overline{1,n}$, и математическая модель выбора наилучшей структуры БД примет вид:

$$x^o = \arg\max_{x\in X}\left\{\sum_{i=1}^{n} a_i\, p_i[K_i(x)]^{\beta}\right\}^{\frac{1}{\beta}}. \qquad (8)$$

В ситуации, когда варианты тестируемых структур имеют равные по значениям оценки полезностей частных критериев $p_i[K_i(x)]$, используем дополнительную информацию, которую предоставляет JMeter. Это величина доверительного интервала полученного значения среднего времени выполнения операции, определяющая качество значений $k_{iv}(x)$:

$$k_{iv}(x) - \frac{t\sigma_{iv}}{\sqrt{N}} < k_{iv}(x) < k_{iv}(x) + \frac{t\sigma_{iv}}{\sqrt{N}}, \qquad (9)$$

где N – размер выборки (определяется автоматически в ходе теста); t – квантиль распределения Стьюдента для значения доверительной вероятности, которое задается в настройках тестирования; σ_{iv} – стандартное (среднеквадратичное) отклонение.

Тогда для описанной ситуации критерий выбора наилучшей структуры имеет вид:

$$x^o = \arg\max_{x\in X}\sum_{i=1}^{n} a_i\, p_i[\sum_{v=1}^{m_i}\sigma_{iv}(x)]. \qquad (10)$$

ОСНОВНЫЕ РЕЗУЛЬТАТЫ

Предложенный метод апробирован с использованием структурированной информации о деятельности страховой компании. Для тестирования и оценивания рассматривается 2 структуры: первая *Schema1* содержит коллекции *policies*, *customers*, *staffs* (рис.1); вторая структура *Schema2* содержит коллекции *policies*; *coverages*, *underwritings*, *coveredItems*, *customers*, *staffs* (рис.2). Также структура *Schema1* рассмотрена с другими параметрами работы БД, обозначена она как *Schema1(1)*.Тесты на скорость чтения, записи, обновления, удаления, агрегации, а также дополнительные параметры настройки работы БД задаются в соответствующих диалоговых окнах *JMeter,* тестовые сценарии применяются поочередно к каждой из структур БД.

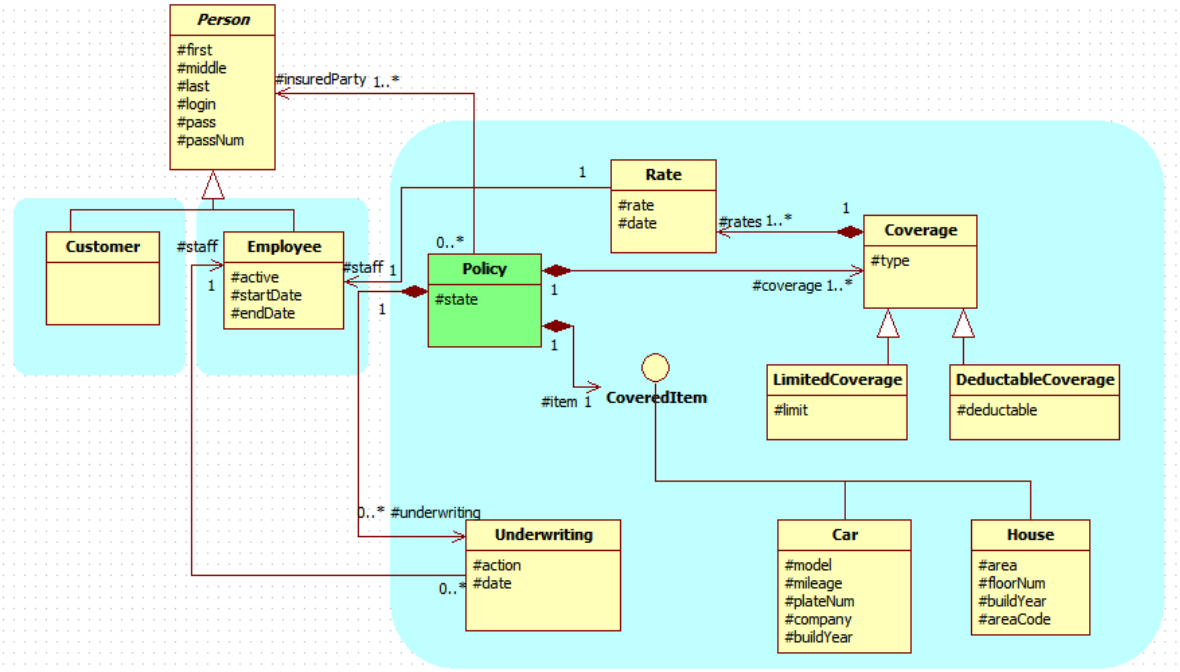


Рисунок 1 – Структура Schema1

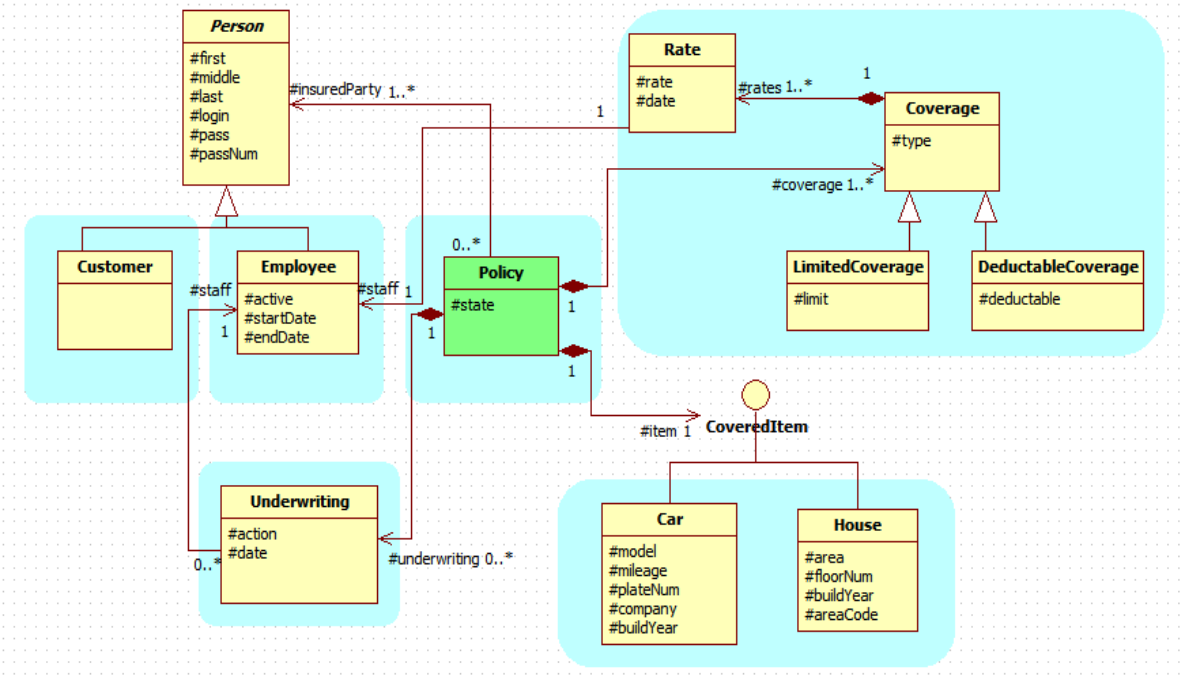
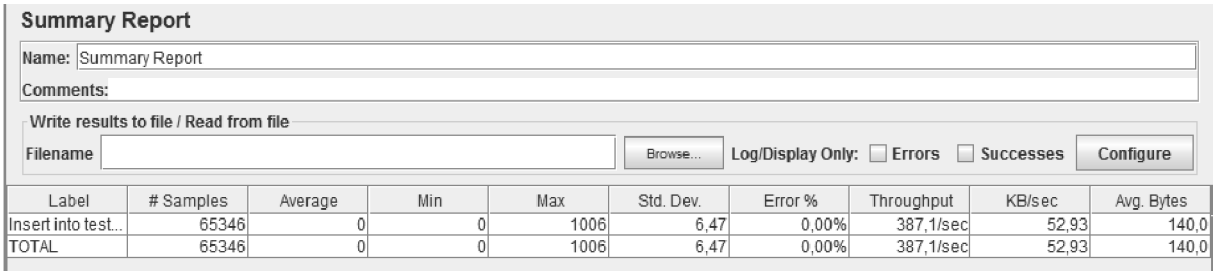


Рисунок 2 – Структура Schema2

Результаты тестирования и оценивания структур для одного сценария сохраняются в Summary Report в виде CSV или XML файлов (рис. 3).



Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec	Avg. Bytes
Insert into test...	65346	0	0	1006	6,47	0,00%	387,1/sec	52,93	140,0
TOTAL	65346	0	0	1006	6,47	0,00%	387,1/sec	52,93	140,0

Рисунок 3 – Визуализация результатов тестирования в JMeter

В табл. 1, 2 приведены результаты тестирования и расчетные оценки полезностей частных критериев $p_{iv}[k_{iv}(x)]$ и обобщенных полезностей структур БД $P(x)$.

Ситуация 1. Известны весовые коэффициенты важности частных критериев a_i, $i=\overline{1,n}$.

По значениям обобщенных полезностей вариантов структур $P(x)$ видно, что наилучшей является структура *Schema 1(1)*.

Ситуация 2. Известно упорядочивание частных критериев $k_1 \succ k_4 \succ k_3 \succ k_2 \succ k_5$. Применив алгоритм последовательной оптимизации (5)–(6), получаем, что в этой ситуации наилучшей является структура *Schema 1*.

Ситуация 3.Когда информация о значениях a_i отсутствует, согласно (7) наилучшей будет структура *Schema 2*, так как $P(x_2)=0.633$, а $P(x_3)=0.6$.

В табл. 2 приведен пример определения наилучшей структуры, если бы *Schema 1 (1)* и *Schema 2* имели одинаковые значения полезностей частных критериев. Используя формулу (10), видим, что наилучшей структурой является структура *Schema 1 (1)*.

Таблица 1 –

Результат оценивания структур при известных значениях важности частных критериев

	Label	Samples	Average $k_{iv}(x)$	Std. Dev.	Cardinality	$p_{iv}[k_{iv}(x)]$	a_i	$a_i p_{iv}[k_{iv}(x)]$	P(x)
Schema 1	Insert1	100000	1	0.22	5	**1**	0.4	0.4	0.516
	Find1	7000	1	3.53	2	0.6667	0.1	0.06667	
	Remove1	50000	3	7.3	3	0	0.2	0	
	Update1	75000	3	0.46	4	0	0.25	0	
	Aggreagate1	100000	2	9.2	1	0.5	0.05	0.05	
Schema 2	Insert1	100000	2	1.15	5	0.5	0.4	0.2	0.508
	Find1	7000	3	5.44	2	0	0.1	0	
	Remove1	50000	1	0.15	3	0.6667	0.2	0.13334	
	Update1	75000	2	7.13	4	0.5	0.25	0.125	
	Aggreagate1	100000	1	8.36	1	1	0.05	0.05	
Schema 1 (1)	Insert1	100000	3	0.5	5	0	0.4	0	**0.55**
	Find1	7000	0	4.75	2	1	0.1	0.1	
	Remove1	50000	0	6.79	3	1	0.2	0.2	
	Update1	75000	1	2.82	4	1	0.25	0.25	
	Aggreagate1	100000	3	4.36	1	0	0.05	0	

Таблица 2 –

Результат оценивания структур при наличии подмножества «равнополезных» альтернатив

	Label	Samples	Average $k_{iv}(x)$	Std. Dev.	$p_{iv}[k_{iv}(x)]$	$p_{iv}[\sigma_{iv}(x)]$	a_i	$a_i p_{iv}[\sigma_{iv}(x)]$	P(x)
Schema 2	Insert1	100000	3	1.15	1.0000	0	0.4	0	0.2
	Find1	7000	0	5.44	1.0000	0	0.1	0	
	Remove1	50000	0	0.15	1.0000	1.0000	0.2	0.2	
	Update1	75000	1	7.13	1.0000	0	0.25	0	
	Aggreagate1	100000	3	8.36	1.0000	0	0.05	0	
Schema 1 (1)	Insert1	100000	3	0.5	1.0000	1.0000	0.4	0.4	**0.8**
	Find1	7000	0	4.75	1.0000	1.0000	0.1	0.1	
	Remove1	50000	0	6.79	1.0000	0	0.2	0	
	Update1	75000	1	2.82	1.0000	1.0000	0.25	0.25	
	Aggreagate1	100000	3	4.36	1.0000	1.0000	0.05	0.05	

ВЫВОДЫ

Рассмотрена задача автоматизированного тестирования, многокритериального оценивания и выбора наилучшей структуры БД MongoDB как один из этапов информационной технологии проектирования MongoDB в рамках САПР. Впервые предложен метод оценивания допустимого множества вариантов структур на основе интеграции средства нагрузочного тестирования JMeter с подсистемой генерации структур MongoDB и программными модулями выбора наилучшей структуры, что обеспечивает достоверность и точность количественных оценок качества структур БД. Наилучшая структура БД определяется с использованием обощенной функции полезности. Для ситуации, когда полезности частных критериев имеют равные значения, для принятия решения используется информация о статистической оценке качества измерений – среднеквадратическом отклонении.

Практическое значение разработанного метода состоит в его использовании для обоснованного выбора

наилучшей с точки зрения производительности и функциональности структуры БД MongoDB при проектировании и разработке документоориентированных хранилищ в информационных системах различного назначения.

ЛИТЕРАТУРА:

1. DB-Engines Ranking : http://db-engines.com/en/system/MongoDB –03.05.2015.
2. Strelchenko V.V. Avtomatizaciya procedur proektirovaniya i testirovaniya struktur bazi dannih MongoDB // Sb. materialov 18 Mezhdunar. molodyozhnogo foruma « Radioelektronika i molodyozh v XXI v.» – 2014. – T.6. – S.329-330.
3. Kalita N.I., Strelchenko V.V. Avtomatizaciya proektuvannya struktur bazi danih MongoDB // Sistemi obrobki informacii. – Harkiv: Harkivskij universitet Povitryanih Sil im. I. Kozheduba. – 2015. – Vip.9 (134). – S. 109–114.
4. Strelchenko V.V. Avtomatizirovannoe ocenivanie struktur BD MongoDB // Zb. materialiv 5 Mizhnar. nauk. konf. studentiv ta molodih vchenih «Modern Information Technology 2015» 21-22 kvitnya 2015. – 2015. – S.151-152.
5. C.J. Date. An Introduction to Database Systems. Addison-Wesley. –1999. – p. 290.
6. Protesting: http://www.protesting.ru/testing/testtypes.html – 20.04.2015.
7. Apache JMeter: http://jmeter.apache.org/index.html – 13.05.2015.
8. Petrov E.G., Novozhilova M.V., Grebennik I.V. Metodi i zasobi prijnyattya rishen u socialno-ekonomichnih sistemah: Navch.posib./ Za red. E.G. Petrova.– K.: Tehnika, 2004. – 256 s.
9. Some thoughts on stress testing web applications with JMeter (part 2): http;//nico.vahlas.eu/2010/03/30/ – 24.05.2015.

Рецензент: *д.т.н., проф. Филатов В.А., Харьковский национальный университет радиоэлектроники.*