

---

## Методы мониторинга в транспортных системах

УДК 004.658:629.7.058

**А. И. Зимовец**

Кафедра математического и программного обеспечения  
Военно-космическая академия им. А. Ф. Можайского, Санкт-Петербург

**А. Д. Хомоненко, д-р техн. наук**

Кафедра «Информационные и вычислительные системы»  
Петербургский государственный университет путей сообщения  
Императора Александра I, Санкт-Петербург

### **ОБОСНОВАНИЕ ВЫБОРА МОДЕЛИ ХРАНЕНИЯ ДАННЫХ ДЛЯ СИСТЕМЫ МОНИТОРИНГА КОСМИЧЕСКОГО ПРОСТРАНСТВА**

Решается задача выбора модели системы управления базами данных для системы мониторинга космического пространства. Основным источником данных о событиях, происходящих в околоземном космическом пространстве, в настоящее время являются радиолокационные и оптические информационно-наблюдательные средства. Одни и те же космические объекты, большую часть которых составляет космический мусор, могут отслеживаться несколькими комплексами за один суточный сеанс наблюдения. Информация обо всех космических объектах хранится в единой базе данных большого размера. При работе с большими данными в системе мониторинга космического пространства важным является обоснованный выбор модели системы управления базами данных с учетом оперативности доступа к хранимым данным. Представлен сравнительный анализ производительности реляционных (PostgreSQL) и нереляционных (MongoDB) систем управления базами данных по времени выполнения операций вставки, обновления и поиска записей. Сделан вывод о целесообразности выбора нереляционной (NoSQL) модели для хранения данных о космических объектах в системе мониторинга космического пространства. Предложена агрегатная информационная модель для хранения данных о космических объектах в системе мониторинга космического пространства с помощью нереляционной системы управления базами данных.

агрегатная информационная модель, база данных, космическое пространство, космический мониторинг, космический мусор, система управления базами данных, NoSQL, PostgreSQL, MongoDB, Oracle.

DOI: 10.20295/2412-9186-2019-5-2-221-232

## Введение

Современный путь освоения космического пространства приводит к растущим информационным потребностям. В настоящее время имеет место рост числа запусков ракет космического назначения, увеличение времени жизни искусственных спутников Земли и их количества, повышается число элементов космического мусора на орбите Земли, кроме того, существуют естественные околоземные космические объекты (астероиды и метеориты). Отмеченные факторы указывают на необходимость совершенствования структуры хранения и обработки данных в системе мониторинга космического пространства (СМКП) [1–3].

Основным источником данных о событиях, происходящих в околоземном космическом пространстве (ОКП), в настоящее время являются радиолокационные и оптические информационно-наблюдательные средства. Ежедневно станциями наблюдения обрабатывается несколько десятков тысяч характеристик космических объектов (КО). Информация о каждом космическом объекте представляет собой двухстрочный набор элементов TLE (от англ. Two-Line Element set), содержащий в себе набор кеплеровских элементов орбит для каждого космического объекта, а также данные о времени и месте пуска, типе объекта, массе и назначении.

В настоящее время в околоземном космическом пространстве функционируют около полутысячи искусственных спутников Земли. В это число входят спутники связи, наблюдения, технологической демонстрации и навигации. Всего на орбите Земли находится более 4 тысяч спутников, а общее число космических объектов, включая космический мусор, достигает 20 тысяч элементов [3]. Это отделяемые ступени ракет-носителей, разгонные блоки и различные мелкие операционные элементы, отделяющиеся от ракетных комплексов в процессе запуска и полета. Наземные средства космического наблюдения могут обнаруживать элементы от 5 мм, а объекты менее 10 см могут наблюдаться только на низких орбитах (до 2000 км).

Бурно развивающаяся космическая деятельность влечет за собой необходимость хранить колоссальные объемы информации о космических объектах. Зачем хранить информацию о космическом мусоре? Кроме очевидной необходимости хранения информации о функционирующих космических аппаратах (КА) различного назначения и сложности, о космических аппаратах, прекративших свое активное существование и оставшихся на орбите, важно иметь информацию о неуправляемых космических объектах. Высокоскоростное движение объектов космического мусора в ОКП создает опасность столкновения с функционирующими космическими аппаратами, что может стать причиной их досрочного вывода из строя. С течением времени может снижаться быстродействие системы хранения информации по причине большого количества хронологических данных, необходимых для прогнозирования обстановки.

Эти факторы показывают необходимость организации хранения больших массивов данных с учетом требований по обеспечению оперативного доступа к ним для организации планирования запусков КА и обеспечения национальной безопасности. Программный комплекс хранения и интеллектуальной обработки информации о космических объектах должен обладать следующими функциями [1]:

- мониторинг в режиме реального времени (или в режиме, близком к реальному времени);
- прогнозирование положения космических объектов;
- выдача данных о положении объектов на заданном интервале времени;
- выдача атрибутивных данных об объектах;
- перевод координат положения объекта в принятые в России системы координат и отсчета;
- выявление пересечения орбит КО и оценка риска их столкновения.

В настоящее время не решена задача устранения избыточности информации, связанная с особенностями расположения отдельных средств мониторинга космического пространства (например, для оптических средств – близость к экваториальной области, высокогорная местность): одни и те же космические объекты могут отслеживаться несколькими комплексами за один суточный сеанс наблюдения.

## 1. Сравнение характеристик доступа к реляционным и нереляционным базам данных

Большинство современных программно-аппаратных комплексов используют реляционную модель хранения данных на основе структурированного языка запросов SQL. Основными свойствами реляционных систем управления базами данных (СУБД) являются наличие средств гибкого доступа к данным в области постоянной памяти, обеспечение параллельной работы с данными нескольких пользователей и общность подходов с точки зрения разработчиков. Несмотря на наличие четкого математического обоснования и высокой стандартизации, реляционная модель имеет свои недостатки: ограниченность и предопределенность набора возможных типов данных атрибутов, низкая оперативность доступа к большим объемам данных [4, 5].

Для решения этих проблем стали использовать ряд подходов, направленных на реализацию хранилищ баз данных, имеющих существенные отличия от моделей, используемых в традиционных реляционных СУБД – NoSQL [6]. В современных условиях развития информационного пространства технология нереляционных баз данных NoSQL имеет преимущества с точки зрения гибкости и оперативности доступа к данным.

Вопрос выбора лучшей модели СУБД для работы с большими объемами данных является важным при решении задач разработки баз данных. Причинами выбора базы данных NoSQL являются производительность работы программиста и эффективность доступа к данным. В разных ситуациях эти условия могут усиливать друг друга или противоречить друг другу. Их трудно обеспечить на ранних стадиях проекта, поскольку проблему выбора модели данных трудно абстрагировать так, чтобы со временем ее можно было заменить другой. Путем прогнозной оценки производительности баз данных важно выявить проблемные места. Это позволяет получить общее представление о целесообразности выбора наилучшей модели данных СУБД.

Для сравнения рассмотрим время работы СУБД PostgreSQL и MongoDB с форматом данных JSON (JavaScript Object Notation – текстовый формат обмена данными, наиболее удобный для работы с TLE), приведенное в статье [7]. Система PostgreSQL является свободным программным обеспечением, стремительно набирает популярность, в том числе в корпоративном сегменте, а MongoDB отличается от большого числа NoSQL решений хорошей проработанностью и поддержкой. На рис. 1–3 показано сравнение времени выполнения операций вставки, обновления и поиска записей (в секундах) для представленных систем с различным числом записей (в тыс. записей) [7].

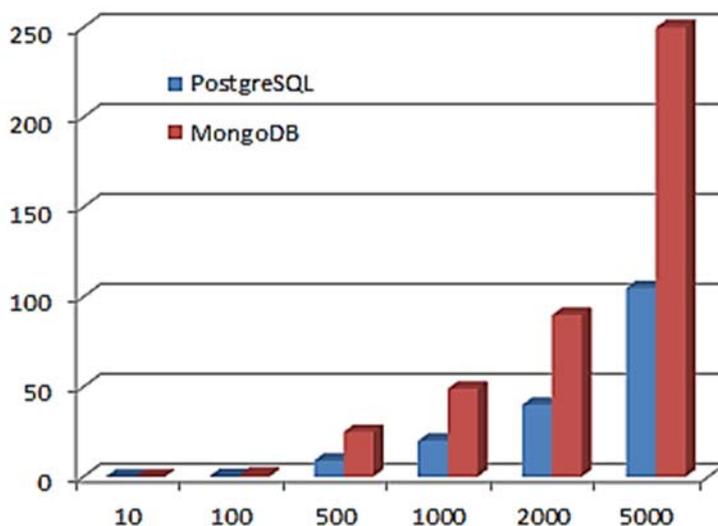


Рис. 1. Длительности вставки записей

Из приведенных зависимостей следует, что СУБД MongoDB опережает PostgreSQL для операций обновления и поиска неиндексированных данных. Однако PostgreSQL опережает MongoDB на операциях вставки данных, что опровергает существующее мнение о преимуществе MongoDB в задачах логирования информации. Следует учитывать, что денормализация

данных как способ превентивного присоединения кортежей в MongoDB является вполне допустимым решением задачи, что особенно ярко проявляется на больших объемах данных. Таким образом, применение СУБД MongoDB является оправданным для хранения редко изменяющихся и часто читаемых данных. Именно эта ситуация характерна для СМПК.

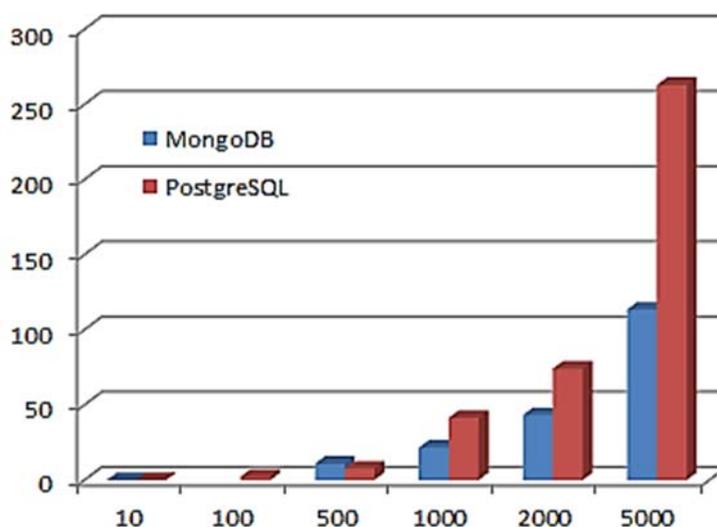


Рис. 2. Длительности обновления данных

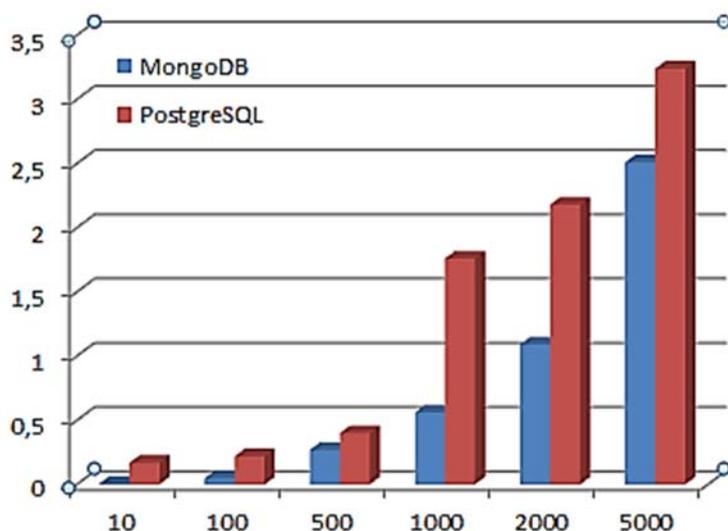


Рис. 3. Длительности неиндексированного поиска

Отметим также работу [8], в которой сравнивается производительность баз данных Oracle и MongoDB путем выполнения сложных запросов к большому набору данных, который доступен в режиме на основе документов и преобразуется в таблицы Oracle. Результаты показывают, что выборка данных (с помощью оператора SELECT) значительно быстрее вы-

полняется в MongoDB, однако некоторые математические запросы, такие как функции агрегирования (sum, count, AVG), заметно быстрее выполняются в Oracle.

Данные по сравнительной оценке характеристик доступа и обработки данных MongoDB и различных СУБД реляционных типов содержатся также в следующих статьях: в [9] проводится сравнение масштабируемых баз данных SQL и NoSQL, в [10] сравниваются характеристики доступа MongoDB и SQL DB, в [11] сравниваются характеристики скорости выполнения запросов в базах данных SQL и MongoDB для больших объемов коммерческих данных.

## 2. Агрегатная модель хранения данных СМКП

Одним из свойств систем NoSQL является отсутствие схемы данных и возможность хранения неструктурированных данных в виде агрегатов – информационных объектов, отражающих структуру предметной области [8] и представляющих собой поименованную совокупность элементов внутри записи, которую можно рассматривать как одно целое (рис. 4) [12].

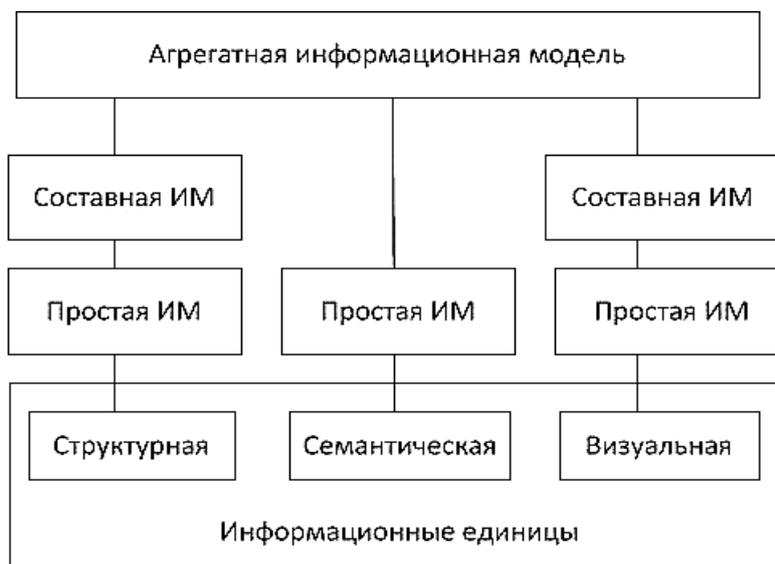


Рис. 4. Структура агрегатной информационной модели

Исходными моделями могут быть структурные, семантические, визуальные и другие информационные единицы. Возможность изменять модели на нижних уровнях позволяет динамически изменять поведение и свойства агрегатной информационной модели. Использование такой модели хранения данных позволит решить проблему избыточности информации в СМКП. В зависимости от условий и целей построения модели агрегирование может производиться по частям, по свойствам, по процессам и по связям.

Применительно к СМКП за основу агрегатной модели данных целесообразно взять формат данных TLE (рис. 5), используемый для определения положения космических объектов в системе NORAD. При этом выборку объектов для агрегатной модели можно осуществить на основании следующих характеристик [2]:

- 1) название объекта;
- 2) номер в каталоге;
- 3) статус активности;
- 4) тип орбиты;
- 5) функциональное назначение;
- 6) страна-владелец.

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34		
I S S ( Z A R Y A )																																			
1																																			
35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	
01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33			
1	2	5	5	4	4	U			9	8	0	6	7	A			0	8	2	6	4	.	5	1	7	8	2	5	2	8					
1	2			3	4			5	6	7			8																						
34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69
-	.	0	0	0	0	2	1	8	2			0	0	0	0	0	-	0	-	1	1	6	0	6	-	4	0		2	9	2	7			
9									10									11						12	13						14				
01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33			
2	2	5	5	4	4				5	1	.	6	4	1	6		2	4	7	.	4	6	2	7		0	0	0	6	7	0	3			
1	2			3						4						5																			
34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69
1	3	0	.	5	3	6	0		3	2	5	.	0	2	8	8		1	5	.	7	2	1	2	5	3	9	1	5	6	3	5	3	7	
6						7						8						9						10											

Рис. 5. Пример представления данных в формате TLE

**Нулевая строка** (необязательная) – заголовок с названием космического объекта.

**Первая строка:**

- 1 (01-01) – номер строки;
- 2 (03-07) – номер спутника в базе NORAD;
- 3 (08-08) – классификация (степень секретности);
- 4 (10-11) – международное обозначение (последние две цифры года запуска);
- 5 (12-14) – международное обозначение (номер запуска в этом году);
- 6 (15-17) – международное обозначение (часть запуска);
- 7 (19-20) – год эпохи (последние две цифры);
- 8 (21-32) – время эпохи (целая часть – номер дня в году, дробная часть – часть дня);

9 (34-43) – первая производная от среднего движения (ускорение), деленная на два [виток/день<sup>2</sup>];

10 (45-52) – вторая производная от среднего движения, деленная на шесть (подразумевается, что число начинается с десятичного разделителя) [виток/день<sup>3</sup>];

11 (54-61) – коэффициент торможения В (подразумевается, что число начинается с десятичного разделителя);

12 (63-63) – изначально – типы эфемерид, сейчас – всегда 0;

13 (65-68) – номер (версия) элемента;

14 (69-69) – контрольная сумма по модулю 10.

#### **Вторая строка:**

1 (01-01) – номер строки;

2 (03-07) – номер спутника в базе данных NORAD;

3 (09-16) – наклонение в градусах;

4 (18-25) – долгота восходящего узла в градусах;

5 (27-33) – эксцентриситет;

6 (35-42) – аргумент перицентра в градусах;

7 (44-51) – средняя аномалия в градусах;

8 (53-63) – частота обращения (оборотов в день);

9 (64-68) – номер витка на момент эпохи;

10 (69-69) – контрольная сумма по модулю 10.

Результатом выборки данных (см. рис. 5) является запись о космическом объекте в формате TLE, приведенная на рис. 6.

```
ISS (ZARYA)
1 25544U 98067A 08264.51782528 -.00002182 00000-0 -11606-4 0 2927
2 25544 51.6416 247.4627 0006703 130.5360 325.0288 15.72125391563537
```

**Рис. 6.** Запись в формате TLE

Для выбора необходимых характеристик предлагается объединить в агрегаты элементы с 4-го по 6-й первой строки, как определяющие статус активности космического объекта, и элементы с 3-го по 7-й второй строки, определяющие тип орбиты. Процедура выбора соответствующих характеристик основана на использовании динамических свойств агрегатной модели.

## **Заключение**

Использование агрегатной модели хранения данных совместно с технологией NoSQL целесообразно для повышения оперативности доступа к информации по космическим объектам в СМКП по сравнению с использованием традиционных подходов на основе технологий SQL.

Дальнейшие исследования целесообразно продолжить в направлении практической разработки базы данных в среде СУБД MongoDB и экспериментальной оценки оперативности доступа к информации по космическим объектам в СМКП. В качестве источников литературы по указанным направлениям исследований можно отметить работы [6, 8–11].

Практический интерес представляет решение задачи автоматического выбора параметров сжатия изображений с потерями на основе инвариантных моментов при дистанционном зондировании Земли и определении космических объектов [13].

Кроме того, большой научно-практический интерес, на наш взгляд, представляет решение задачи организации параллельной обработки запросов к базам данных на основе использования технологии NoSQL, а также сравнение с использованием распараллеливания запросов SQL. Здесь можно отметить работы [14, 15]. Важным, по нашему мнению, является оценивание оперативности функционирования бортового комплекса управления космическими аппаратами дистанционного зондирования Земли и определения космических объектов, а также организации адаптивной обработки запросов к базам данных [16, 17].

## Библиографический список

1. Горянский А.С., Пророк В.Я. Анализ планирования сбора некоординатной информации для средств контроля космического пространства // Труды ВКА им. А. Ф. Можайского. – 2015. – № 647. – С. 71–76.
2. Груздев С. С., Журкин И. Г., Орлов П. Ю., Панкин А. В. Анализ состояния и перспектив по созданию электронного каталога объектов околоземного космического пространства // Современные проблемы дистанционного зондирования Земли из космоса. – 2016. – Т. 13. – № 6. – С. 51–59
3. Левкина П. А. Физические и орбитальные характеристики объектов космического мусора по данным оптических наблюдений : дис. ... канд. физ.-мат. наук : 01.03.01 : защищена 10.08.2016 : утв. 23.09.2016 // Левкина Полина Александровна. – М., 2016. – 123 с. Библиогр. : с. 115–123. – 006651939.
4. Утепбергенов И. Т., Хомоненко А. Д. Базы данных в информационных системах : учебник. – Алматы : Экономика, 2013. – 540 с.
5. Модели информационных систем : учеб. пособие / В. П. Бубнов, М. Л. Глухарев, А. А. Корниенко, С. А. Краснов, В. В. Рогальчук, А. В. Тырва, В. В. Федянин, А. Д. Хомоненко. – М., 2015. – 188 с.
6. Садаладж П. Дж., Фаулер М. NoSQL: новая методология разработки нереляционных баз данных / пер. с англ. – М. : И. Д. Вильямс, 2013. – 192 с.
7. Новиков Б. А., Левин М. Ю. Сравнительный анализ производительности SQL и NoSQL СУБД // Компьютерные инструменты в образовании. – 2017. – № 4. – С. 48–63.
8. Faraj A., Rashid B., Shareef T. Comparative Study of Relational and Nonrelations Database Performances Using Oracle and Mongoddb Systems. International Journal of Computer Engineering and Technology (IJCET), Volume 5, Issue 11, November 2014. – Pp. 11–22.

9. Cattell R. Scalable SQL and NoSQL data stores // ACM SIGMOD Record, Vol. 39 Is. 4, December 2010. N. Y. : ACM New York. – Pp. 12–27.
10. Parker Z. et al. Comparing NoSQL MongoDB to an SQL DB, Proceedings of the 51st ACM Southeast. Conference. N. Y. : ACM New York, 2013. – Pp. 238–243.
11. Seyyed Hamid Aboutorabi ; Mehdi Rezapour ; Milad Moradi , Nasser Ghadiri. Performance evaluation of SQL and MongoDB databases for big e-commerce data. CSSE 2015 – 20th International Symposium on Computer Science and Software Engineering. Pp. 1–7.
12. Булгаков С. В. Агрегирование информационных моделей // Перспективы науки и образования. – 2014. – № 3 (9). – С. 9–13.
13. Старобинец Д. Ю., Хомоненко А. Д., Гаврилова Н. А. Автоматический выбор параметров сжатия изображений с потерями на основе инвариантных моментов при дистанционном зондировании Земли // Современные проблемы дистанционного зондирования Земли из космоса. – 2017. – Т. 14. – № 5. – С. 26–36.
14. Smirnov K., Chernishev G. Empirical study of parallel SQL query execution // Труды института системного программирования РАН. – 2011. – Т. 21. – С. 245–262.
15. Молодкин И. А., Свистунов С. Г. Сравнительный анализ алгоритмов распределения работ в мультипроцессорных системах // Интеллектуальные технологии на транспорте. – 2018. – № 2. – С. 41–45.
16. Хомоненко А. Д., Старобинец Д. Ю. Лохвицкий В. А. Модель оценки оперативности функционирования бортового комплекса управления космическими аппаратами дистанционного зондирования земли // Труды СПИИРАН. – 2016. – № 3 (46). – С. 49–64.
17. Deshpande A., Ives Z., Raman V. 2007. Adaptive query processing. Foud. Trends database 1, 1 (January 2007). – Pp. 1–140.

*Alena I. Zimovets*

«Mathematical and software» department Military Space Academy  
them. A. F. Mozhaisky

*Anatoly D. Khomonenko*

«Information and Computing Systems» department  
Emperor Alexander I St. Petersburg State Transport University

### **The rationale for choosing a data storage model for a space monitoring system**

The problem of choosing a model of a database management system for a space monitoring system is being solved. The main source of data on events occurring in near-Earth outer space are currently radar and optical information-observational means. The same space objects, most of which are space debris, can be tracked by several complexes in one daily observation session. Information about all space objects is stored in a single large database. When working with big data in the space monitoring system, it is important to make a reasonable choice of a model of a database management system taking into account the speed of access to stored data. A comparative analysis of the performance of relational (PostgreSQL) and non-relational (MongoDB) database management systems by the time of performing the operations of inserting, updating and searching records is presented. It is concluded that it is advisable to choose a non-relational (NoSQL) model for storing data on space objects in the space monitoring

system. An aggregate information model is proposed for storing data on space objects in a space monitoring system using a non-relational database management system.

aggregate information model, database, outer space, space monitoring, space debris, database management system, NoSQL, PostgreSQL, MongoDB, Oracle.

## References

1. Goryansky A. S., Prorok V. Y. (2015). Analysis of planning for the collection of non-coordinate information for space control [Analiz planirovaniya sbora nekoordinatnoi informacii dlya sredstv kontrolya kosmicheskogo prostranstva]. Proceedings of the military space Academy A. F. Mozhaisky [Trudi VKA im. A. F. Mojaiskogo], issue 647. – Pp. 71–76.
2. Gruzdev S. S., Zhurkin I. G., Orlov P. Y., Pankin A. V. (2016). Analysis of the state and prospects for the creation of an electronic catalogue of near-Earth space objects [Analiz sostoyaniya i perspektiv po sozdaniyu elektronnoho kataloga obektov okolozemnogo kosmicheskogo prostranstva]. Modern problems of remote sensing of the Earth from space [Sovremennye problemi distancionnogo zondirovaniya Zemli iz kosmosa], vol. 13, issue 6. – Pp. 51–59.
3. Levkina P. A. (2016). Physical and orbital characteristics of space debris objects according to optical observations: diss. cand. ph.-m. science, Moscow [Fizicheskie i orbitalnie harakteristiki obektov kosmicheskogo musora po dannim opticheskikh nablyudenii dis. kand. fiz.-mat. nauk, Moskva]. – 123 p.
4. Utebergenov I. T., Khomonenko A. D. (2013). Databases in information systems. Textbook [Bazi danih v informacionnih sistemah. Uchebnik]. Almaty : Economica [Almati : Ekonomika]. – 540 p.
5. Models of information systems [Modeli informacionnih sistem]. (2015). Bubnov V. P., Glukharev M. L., Kornienko A. A., Krasnov S. A., Rogalchuk V. V., Tyrva A. V., Fedyanin V. V., Khomonenko A. D. Study Guide. Moscow. – 188 p.
6. Pramod J. Sadalage, Martin Fowler. (2013). NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence. Addison-Wesley Professional. – 192 p.
7. Novikov B. A., Levin M. Y. (2017). Comparative Analysis of the Performance of SQL and NoSQL DBMS. Computer Tools in Education, issue. 4. – Pp. 48–63.
8. Faraj A., Rashid B., Shareef T. (2014). Comparative Study of Relational and Nonrelations Database Performances Using Oracle and MongoDB Systems. International Journal of Computer Engineering and Technology (IJCET), Volume 5, Issue 11, November 2014. – Pp. 11–22.
9. Cattell R. (2010). Scalable SQL and NoSQL data stores // ACM SIGMOD Record, Vol. 39 Is. 4, December 2010. N. Y. : ACM New York. – Pp. 12–27.
10. Parker Z. et al. Comparing NoSQL MongoDB to an SQL DB, Proceedings of the 51st ACM Southeast. Conference. N. Y. : ACM New York, 2013. – Pp. 238–243.
11. Seyyed Hamid Aboutorabi, Mehdi Rezapour, Milad Moradi, Nasser Ghadiri. (2015). Performance evaluation of SQL and MongoDB databases for big e-commerce data. CSSE 2015 – 20th International Symposium on Computer Science and Software Engineering. – Pp. 1–7.
12. Bulgakov S. V. (2014). Aggregation information models [Agregirovanie informacionnih modelei]. Perspectives of Science and Education [Perspektivi Nauki i Obrazovaniya], issue 3(9). – Pp. 9–13.

13. Starobinets D. Yu., Khomonenko A. D., Gavrilova N. A. (2017). Automatic selection of image compression parameters with losses based on invariant moments for Earth remote sensing purposes [Avtomaticeskii vibor parametrov sjatiya izobrajenii s poteryami na osnove invariantnih momentov pri distancionnom zondirovanii Zemli]. Modern problems of remote sensing of the Earth from space [Sovremennie problemi distancionnogo zondirovaniya Zemli iz kosmosa], vol. 14. issue 5. – Pp. 26–36.
14. Smirnov K., Chernishev G. (2011). Empirical study of parallel SQL query execution. Proceedings of the Institute for System Programming RAS [Trudi instituta sistemnogo programmirovaniya RAN], vol. 21. – Pp. 245–262.
15. Molodkin I. A., Svistunov S. G. (2018). Comparative Analysis of Scheduling Algorithms in Multiprocessor Systems [Sravnitel'nyi analiz algoritmov raspredeleniya rabot v multiprocessornih sistemah]. Intellectual Technologies on Transport [Intellektualnie tehnologii na transporte], issue 2. – Pp. 41–45.
16. Khomonenko A. D., Starobinets D. Yu., Lokhvitsky V. A. (2016). A model for evaluating the operational efficiency of an onboard complex for controlling remote sensing spacecraft [Model ocenki operativnosti funkcionirovaniya bortovogo kompleksa upravleniya kosmicheskimi apparatami distancionnogo zondirovaniya Zemli]. SPIIRAS Proceedings [Trudi SPIIRAN], issue 3 (46). – Pp. 49–64.
17. Deshpande A., Ives Z., Raman V. (2007). Adaptive query processing. Foud. Trends database 1, 1 (January 2007). – Pp. 1–140.

*Статья представлена к публикации членом редколлегии Вал. В. Сапожниковым  
Поступила в редакцию 04.11.2018, принята к публикации 17.01.2019*

*ЗИМОВЕЦ Алена Игоревна* – адъюнкт кафедры «Математического и программного обеспечения» Военно-космической академии им. А. Ф. Можайского;

e-mail: zim\_alenka@rambler.ru

*ХОМОНЕНКО Анатолий Дмитриевич* – доктор технических наук, профессор кафедры «Информационные и вычислительные системы» Петербургского государственного университета путей сообщения Императора Александра I, профессор кафедры «Математического и программного обеспечения» Военно-космической академии им. А. Ф. Можайского;

e-mail: khomon@mail.ru

© Зимовец А. И., Хомоненко А. Д., 2019