

УДК 681.5

ВЫБОР ОПТИМАЛЬНОЙ JSON-МОДЕЛИ ДЛЯ ХРАНЕНИЯ РЕЗУЛЬТАТОВ ИСПЫТАНИЙ

Шмидт И.А., Васенев Н.В.

*Пермский национальный исследовательский политехнический университет,
Пермь, e-mail: shmidt@msa.pstu.ru*

Статья посвящена исследованию способов хранения результатов испытания сложных технических изделий. Результатом проведения испытания является массив более чем 100 000 разнородных параметров. При сохранении данных в документноориентированном виде возможны различные варианты компоновки данных в документы. В качестве системы хранения данных выбрана документно-ориентированная БД MongoDB. Рассматриваются различные конфигурации JSON-документов для сохранения данных и определяется оптимальная с точки зрения быстродействия. Результаты получены при помощи моделирования, воспроизводящего процесс испытаний. Моделирование реализовано в многофункциональной платформе LabVIEW. Проводится анализ полученных результатов и выбирается самая быстрая по скорости записи структура хранения данных. Результаты исследований показали, что для типичного набора данных, генерируемого испытаниями, наибольшая скорость записи достигается при использовании структуры типа «временной ряд».

Ключевые слова: структура хранения, сбор данных, Mongo DB, временной ряд

THE CHOICE OF THE OPTIMAL JSON MODEL FOR STORING TEST RESULTS

Shmidt I.A., Vasenev N.V.

Perm National Research Polytechnic University», Perm, e-mail: shmidt@msa.pstu.ru

The article is devoted to the method of storing test results of complex technical products. The result of the test is an array of more than 100 000 diverse settings. There are various options of data packaging in documents by storing the data in document-oriented view. As the storage system, was selected document-oriented database MongoDB. Discusses the various configuration documents for saving data, determined the optimal in terms of performance. The simulation is implemented in the multi-platform LabVIEW. At the end of this article there is the analysis of the obtained results and select the fastest recording speed of data storage structure. For a typical data set the highest recording speed is achieved using the structure type «time series».

Keywords: storage structure, collection data, MongoDB, timing series

Стенды для проведения испытаний сложных технических изделий оснащены несколькими сотнями измерительных каналов и большим количеством управляющих элементов. Следовательно, системы сбора, передачи, записи и хранения информации довольно сложны и требуют грамотного подхода в разработке. Система архивации должна обладать массой противоречивых качеств: высокой частотой записи данных, скоростью доступа – и учитывать ограничения по объему памяти [2].

Цель данной статьи – разработка и выбор структуры хранения результатов испытания, при которой сохранится высокая скорость записи данных и повысится гибкость системы.

Научная новизна заключается в разработке универсальной структуры хранения результатов параметров испытаний, отличающейся высокой скоростью записи и гибкостью системы, что позволит добавлять и изменять новые параметры, не влияя на систему хранения.

Для хранения параметров испытаний была выбрана документноориентированная база данных MongoDB. Эта БД отличается

рядом преимуществ: гибкость, скорость записи, простота в получении данных из базы, надежность, бесплатность. Подробное обоснование выбора базы данных MongoDB приводится в статье [4].

При сохранении трендов, полученных при проведении испытаний, в виде набора документов возникает проблема поиска оптимальной структуры таких документов [3]. В MongoDB документы изначально представляются в JSON-формате, поэтому будем говорить о выборе оптимальной JSON-модели. Под оптимальностью, прежде всего, мы понимаем скорость записи трендов в БД. Система сбора параметров стенда реализована в многофункциональной платформе LabVIEW [3], поэтому средой разработки приложений для исследования и выбора JSON-модели, оптимальной для сохранения параметров, также выбрана LabVIEW [5]. Обоснование выбора среды разработки приводится в статье [1].

Для определения оптимальной структуры было разработано специальное приложение на платформе LabVIEW. Приложение имитирует поток данных от испытательного

стенда, формирует JSON-документы и записывает их в БД [4].

Идея, примененная в приложении, заключается в использовании буфера, который получает сформированные документы, а затем выгружает их в базу данных в виде массива. Массив также является документом. В базе данных MongoDB каждому новому документу присваивается уникальное поле, что занимает некоторое время. Чем больше документов сохраняется, тем больше времени требуется на их обработку. С увеличением размера буфера, уменьшается частота записи документов в БД, следовательно, уменьшается общее время их обработки, что, в свою очередь, влияет на скорость записи.

При разработке структуры хранения параметров испытания необходимо использовать идею с применением буфера. Также разрабатываемая структура должна отвечать трем основным требованиям:

1. Гибкость. Набор регистрируемых параметров при проведении испытаний разного типа может меняться. Изменение или добавление новых датчиков в исследовательский стенд не должно влиять на систему хранения с разработанной структурой записи данных.

2. Быстродействие. Частота записи параметров может достигать 200 Гц, поэтому скорость сохранения данных в БД должна быть максимально быстрой;

3. Простота получения информации. Для обработки и анализа полученной информации, структура должна позволять легко получать данные из базы с помощью высокоуровневых языков запроса.

Были рассмотрены три различных структуры хранения параметров испытаний, которые условно можно назвать «по вертикали», «по горизонтали» и «подо-

кументно». Суть каждой разработанной структуры проиллюстрирована на рис. 1.

На данном рисунке представлена таблица, в которой по строке расположены названия параметров, а по столбцу – кадры. Кадр – набор значений параметров, относящихся к одной временной отметке. На пересечении кадра и параметра вносится записанное значение физической величины. Число параметров в таких стендах приблизительно 100–500 штук, а частота снятия может достигать 200 Гц. Испытания могут проводиться несколько часов, следовательно, количество столбцов меньше количества строк на несколько порядков. Для подробного рассмотрения разработанных структур, необходимо обратить внимание на выделенные области на рис. 1.

Данные, соответствующие первой разработанной структуре, выделены синим. Формат такой структуры выглядит следующим образом:

название_параметра_1: [{время: значение_1, физ_величина: значение_1}, {время: значение_2, физ_величина: значение_2}, ..., {время: значение_n, физ_величина: значение_n}].

В подсистеме регистрации параметров испытаний происходит последовательный опрос датчиков с помощью буфера опроса. Датчики, которые реагируют на сигнал опроса, посылают полученное значение физической величины и свой номер. К полученному физическому значению добавляется поле с временем записи. Такая пара формируется в отдельный документ для каждого параметра. Эти документы поступают в массив, ключом которого является номер датчика. Контролируя размер массива, контролируется размер буфера, что позволяет изменять скорость записи индивидуально.

		Параметр						
		P1	P2	P3	P4	P5	P6	Pn
Время	T1=0.1	Значение				Значение		
	T2=0.2	Значение		Значение		Значение		
	T3=0.3	Значение				Значение		
	T4=0.4	Значение		Значение		Значение		
	T5=0.5	Значение				Значение		
	T6=0.6	Значение	Значение	Значение	Значение	Значение	Значение	
	Tn=...							

1		Запись по каждому параметру в массив
2		Запись временного ряда в массив
3		Запись каждого значения параметра без массива

Рис. 1. Разработанные структуры регистрации параметров испытаний

Второй разработанный формат на рис. 1 обведен оранжевым. Структура называется «временной ряд». Работа подсистемы регистрации параметров испытаний начинается с опроса всех датчиков. Один цикл опроса – это один кадр. Если за один кадр хотя бы один датчик посылает сигнал на запись вместе с измеренным параметром, то имя этого параметра и его физическое значение записывается в отдельный документ. Далее, если за этот же кадр произошло получение данных с других датчиков, то их имена и физические значения также записываются в отдельные документы. Все созданные документы за один кадр формируются в массив. После завершения опроса происходит выгрузка всех данных в БД. Формат рассмотренной структуры следующий:

время_1: [{название_параметра: имя_1, физ_величина: значение_1},
{название_параметра: имя_2, физ_величина: значение_2}, ...,
{название_параметра: имя_n, физ_величина: значение_n}]

Частота опросов равна минимальной частоте записи параметра, существующего

в исследовательском стенде. Размер массива зависит от количества опрошенных датчиков, поэтому изменять этот размер невозможно. Структура наглядно отображает частоту записи параметров и полученные физические значения, что удобно для дальнейшей обработки и анализа.

Третья разработанная структура на рис. 1 обозначена зеленым. При такой структуре в ходе опроса датчиков формируются документы, состоящие из времени записи, названия параметра и его физического значения. Формат такой структуры выглядит следующим образом:

внешний_буфер: [{название_параметра: имя_1, физ_величина: значение_1, время: значение_1},

{название_параметра: имя_2, физ_величина: значение_1, время: значение_1},

{название_параметра: имя_1, физ_величина: значение_2, время: значение_2}, ...,

{название_параметра: имя_t, физ_величина: значение_n, время: значение_n}]

Такие документы формируются для каждого датчика по каждому полученному значению физической величины.

Таблица 1

Результат работы приложения

Необходимое количество элементов, шт	Размер буфера, шт	Общее время записи, мс	Время записи одного документа, мс
5000	20	2250	0,45
5000	40	1192	0,24
5000	60	1151	0,23
5000	80	1073	0,21
5000	100	974	0,19

Зависимость времени записи одного документа от размера буфера

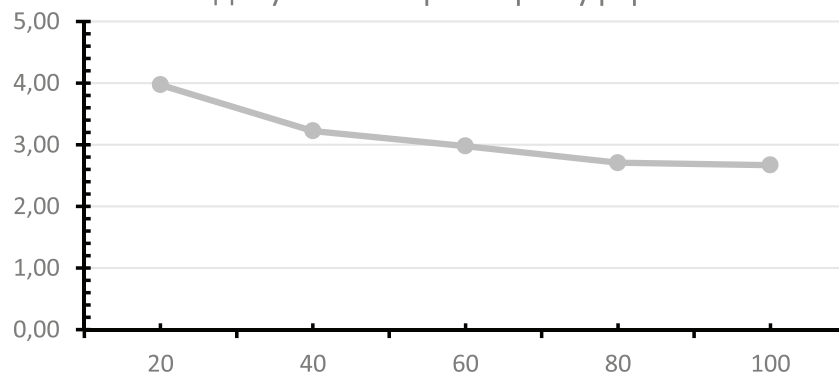


Рис. 2. Зависимость времени записи одного документа от размера буфера

Таблица 2

Результат работы приложения

Необходимое количество элементов, шт	Количество опрашиваемых датчиков, шт	Общее время записи, мс	Время записи одного документа, мс
5000	10	3355	0,6
5000	50	1002	0,2
5000	100	492	0,09

Таблица 3

Результат работы приложения

Необходимое количество элементов, шт.	Размер буфера, шт.	Общее время записи, мс	Время записи одного документа, мс
5000	20	19865	3,97
5000	40	16120	3,22
5000	60	14900	2,98
5000	80	13540	2,71
5000	100		2,67

В предыдущих двух структурах были использованы возможности БД MongoDB объединения документов в массив, который выступал в роли буфера. Для данной структуры использование таких массивов не предусмотрено, поэтому для контролирования скорости записи предлагается ввести дополнительный внешний буфер. Такой буфер не относится к структуре БД MongoDB и является массивом среды разработки.

Разработанные исследовательские приложения также выполняют следующие функции:

1. Добавление и изменение информации о датчиках. Информация содержит частоту снятия им норме датчика.
2. Изменение размера буфера.
3. Подключение к БД MongoDB.
4. Расчет времени записи документов в БД MongoDB.
5. Расчет количества записанных документов.

Для приложения были созданы подпрограммы, которые формируют одну из трех разработанных структур.

Для определения скорости записи одного документа в базу данных проводится ряд экспериментов. В каждом эксперименте изменяется размер буфера. При изменении размера буфера изменяется количество документов, записанных в массив. Размер буфера равен 20, 40, 60, 80, 100 элементам на разных этапах исследования. Результаты исследования представлены в табл. 1.

Как видно из рис. 2, при увеличении размера буфера уменьшается общее время

записи в БД, следовательно, и время записи одного элемента.

Таким образом, идея с применением буфера для созданной структуры работает.

Исследование проводится в несколько этапов. На каждом этапе вместо изменения размера буфера меняется количество датчиков. Были выбраны 10, 50, 100 датчиков. Для структуры «временной ряд» массив формируется из документов, созданных за один кадр. Поэтому количество документов в массиве будет разным. Результаты работы приложения представлены в табл. 2.

Итоговая таблица показывает, что при увеличении количества датчиков резко снижается время записи в БД. Для структуры «временной ряд» идея с буфером также работает, однако в отличие от предыдущей рассмотренной структуры здесь нет возможности контролировать размер буфера. На рис. 3 представлен график зависимости времени записи одного документа от числа опрашиваемых датчиков.

Исследование скорости проводится в несколько этапов. На каждом этапе меняется размер буфера, который равен 20, 40, 60, 80, 100 значениям. Разработанная структура подразумевает запись всех данных в БД без использования массива документов. В качестве буфера для данной структуры применяется массив ссылок на документы. Такой массив является классом среды разработки LabVIEW. Результаты исследования приведены в табл. 3.

Зависимость времени записи одного документа от количества датчиков

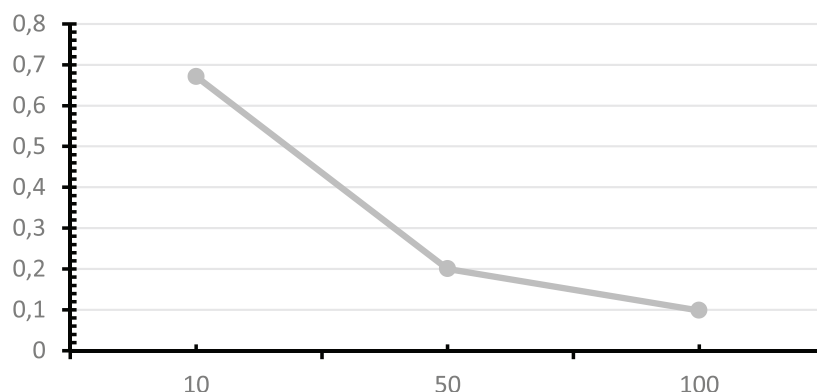


Рис. 3. Зависимость времени записи одного документа от количества датчиков

Зависимость времени записи одного документа от размера буфера

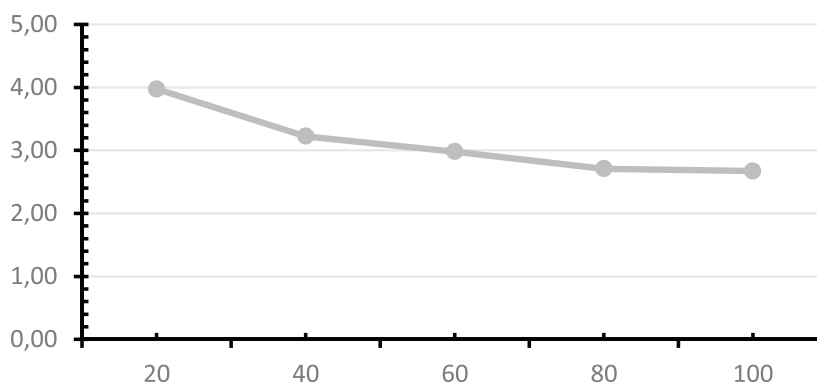


Рис. 4. Зависимость времени записи одного документа от размера буфера

Как видно из таблицы, приложение с рассматриваемой структурой записи тратит много времени на сохранение полученных данных в БД. Это связано с тем, что каждое полученное значение формируется в отдельный документ и записывается в БД без использования массива документов. База данных MongoDB обрабатывает каждый записанный документ и присваивает ему уникальный номер. Когда каждое полученное значение физической величины формируется в отдельный документ, обработка и присвоение номера проходит чаще, что существенно снижает скорость записи в БД. Кроме этого, при расчете скорости записи в базу данных учитывается время обработки массива ссылок на документы.

На рис. 4 представлен график зависимости общего времени записи от размера буфера.

Заключение

В статье были рассмотрены разработанные структуры хранения параметров испытаний и с помощью созданных приложений были проведены исследования скорости записи каждой структуры. Результаты работы исследований показали наибольшую скорость записи у структуры с формированием массива для каждого датчика и структуры «временной ряд». Это связано с процессом обработки сформированных документов и массивов в БД MongoDB. Каждому такому документу присваивается уникальный ключ. Когда поступает большое количество

документов, уходит некоторое время на его формирование и присваивание уникального ключа. Чем больше таких документов поступает, тем больше уходит времени на их обработку. Поэтому при структуре с меньшим числом документов требуется меньше времени на запись. Кроме этого, для структуры «временной ряд» использование динамически меняющегося размера буфера позволяет не нагружать подсистему регистрации параметров и записывать столько данных, сколько было получено.

Список литературы

1. Попов Д.А., Шмидт И.А. Оперативный сбор параметров при испытаниях различных типов ГТУ на базе оборудования NATIONAL INSTRUMENTS // Научно-технический вестник Поволжья. – 2014. – № 2. – С. 192–196.
2. Попов Д.А., Шмидт И.А. Разработка системы управления архивными данными испытаний газотурбинных установок большой мощности // Современные проблемы науки и образования. – 2014. – № 2.; URL: <http://science-education.ru/ru/article/view?id=12035>.
3. Попов Д.А., Шмидт И.А. Разработка функциональной структуры программного комплекса испытаний газотурбинных установок мощностью до 40 мвт. // На-

учные исследования и инновации. – 2012. – Т. 6. № 1–4. – С. 264–270.

4. Igor Shmidt, Storing Data in the Trial of Complex Technical Products. // Proc. of the 2nd International Conference on Applied Innovations in IT, (ICAИТ), March 2014.

5. LabVIEW. Руководство пользователя [Электронный ресурс]. – URL:ftp://ftp.ni.com/pub/branches/russia/software/labview_user_manual.pdf (дата обращения: 27.04.2014).

References

1. Popov D.A., Shmidt I.A. Operativnyj sbor parametrov pri ispytaniyah razlichnyh tipov GTU na baze oborudovaniya NATIONAL INSTRUMENTS // Nauchno-tehnicheskij vestnik Povolzhja. 2014. no. 2. pp. 192–196.

2. Popov D.A., Shmidt I.A. Razrabotka sistemy upravlenija arhivnymi dannymi ispytaniy gazoturbinnyh ustanovok bolshoj moshhnosti // Sovremennye problemy nauki i obrazovaniya. 2014. no. 2.; URL: <http://science-education.ru/ru/article/view?id=12035>.

3. Popov D.A., Shmidt I.A. Razrabotka funkcionalnoj struktury programmnoho kompleksa ispytaniy gazoturbinnyh ustanovok moshhnostju do 40 mvt. // Nauchnye issledovaniya i innovacii. 2012. T. 6. no. 1–4. pp. 264–270.

4. Igor Shmidt, Storing Data in the Trial of Complex Technical Products. // Proc. of the 2nd International Conference on Applied Innovations in IT, (ICAИТ), March 2014.

5. LabVIEW. Rukovodstvo polzovatelja [Elektronnyj resurs]. URL:ftp://ftp.ni.com/pub/branches/russia/software/labview_user_manual.pdf (data obrashheniya: 27.04.2014).