

УДК 621.452.3.018.7:004.652.4

Д. И. ВОЛКОВ, Н. Н. ЛОПУНОВА

АО «Элемент», Одесса, Украина

ОПТИМИЗАЦИЯ ОБЪЁМА НЕРЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ И ВРЕМЕНИ ДОСТУПА К ДАННЫМ ПРИ МНОЖЕСТВЕННЫХ ЧАСТОТАХ РЕГИСТРАЦИИ ПАРАМЕТРОВ

Рассмотрено использование нереляционной базы данных для регистрации параметров работы ГТД. Сформулированы обобщенные технические требования к базе данных с учетом прикладной специфики задачи. Приведено обоснование выбранного формата сохранения данных, расчёты скорости записи и чтения данных для выбранных технических способов и с учетом большого размера файлов(10+Gb), сложность отдельных вычислительных алгоритмов, плотности хранения данных и т.д. Учтены разные типы данных и множественные частоты (10+) регистрации параметров работы ГТД (1000+). Представлено разработанное в АО «Элемент» программное обеспечение регистратора параметров для работы с базой данных летных испытаний ГТД, КПА, нереляционные базы данных

Ключевые слова: регистрация параметров работы двигателя и систем, лётные испытания ГТД, КПА, нереляционные базы данных

Введение

Базы данных (БД) условно делятся на реляционные и нереляционные.

Первые предназначены для хранения структурированных связанных между собой данных.

Вторые - это нишевые базы данных, которые решают свои узкоспециализированные задачи, такие как хранение документов, хранение потоковых данных и т.п.

В данной статье рассматривается решение задачи регистрации параметров работы двигателя и систем при стендовых и лётных испытаниях газотурбинных двигателей (ГТД). Задача была поставлена в процессе создания бортового регистратора параметров и решена с использованием нереляционной базы данных собственной разработки. База данных является доработанным вариантом предыдущей версии, которая применяется в программно-технических комплексах испытаний ГТД и контрольно-поверочной аппаратуре (КПА) производства АО «Элемент».

1. Постановка задачи

Необходимо выполнять запись и просмотр изменений во времени значений параметров работы ГТД и систем.

Частоты записи параметров различны и находятся в диапазоне от долей герц до десятков килогерц.

Количество параметров 1000+. Соответственно,

объём файла БД одного испытания составляет несколько десятков гигабайт. Количество частот записи обычно ограничено 10...20. Просмотр не должен требовать предварительной обработки и может выполняться не только после испытаний, но и одновременно с записью БД. При этом извлечение информации должно выполняться без значительных задержек, так, чтобы пользователь мог увидеть результат (например, изменения масштаба времени и интервала просмотра) за время, соизмеримое со временем восприятия графической информации, создавая у пользователя ощущение плавной работы и не приводя к прерыванию процесса осмысления результатов испытания ГТД – основного предназначения нашего продукта.

2. Концепция

2.1. Современные компьютеры (либо контроллеры) и SSD-накопители обеспечивают высокую скорость записи базы данных независимо от ее формата. Единственным, разве что, условием является потоковая запись, при которой можно пренебречь временем доступа к накопителю.

Определяющим по времени является чтение базы данных. Здесь важное замечание: большой размер файла исключает чтение всего файла при инициализации, если следовать ранее заявленным целям, а именно моментальному началу работы и, особенно, работе параллельно с выполнением записи.

Даже при использовании самых современных SSD-накопителей узким местом является использу-

емый интерфейс. Например, SATA 3.0 ограничивает скорость чтения 6 Гбитами. Использование PCIe 3.0 может, в теории, используя 4 линии, дать 4 ГБ, но на практике можно достичь порядка 2 ГБ, что соответствует десяткам секунд ожидания только считывания, не говоря уже об обработке упомянутого объема «на лету». И, более того, такая нагрузка недопустима и такая скорость недостижима параллельно с записью БД и отображением данных в реальном масштабе времени во время испытаний. Также данный подход упирается в объем устанавливаемой оперативной памяти.

Обобщенно, решением является либо индексация неких блоков записи данных по времени (в этом случае метки времени являются индексами и должны записываться отдельно), либо использование блоков, положение которых однозначно определяется меткой времени и просто вычисляется (опять производительность). С учетом того, что частоты измерения параметров являются неизменными в процессе записи одной БД (т.е. существует интервал времени, на котором объем данных тот же), был выбран второй вариант. В разделе 2 приведены детали данного решения. Здесь стоит заметить, что такой подход исключает непрерывную архивацию БД. Архивация выполняется отдельно, когда отсутствует необходимость оперативно получить доступ к данным.

Резюмируя: данные записываются блоками одного размера, неизменного в процессе записи БД, что позволяет при просмотре непосредственно считать интересующие блоки. При этом, например, при разрешении экрана по горизонтали 1920 точек имеем соответствующее ограничение на количество

блоков, которое требуется прочитать в режиме сёрфинга. При 95000/90000 IOps это порядка 20 мс в сумме на доступ, причём, в худшем случае, когда блоки не кешируются при большом масштабе просмотра. Естественно, это предварительное отображение, и время фонового чтения всех данных, выполняемое для детализированного отображения графиков с учётом локальных минимумов и максимумов, пропорционально объёму соответствующего участка БД.

Такой подход обеспечивает быстрое отображение участка БД при изменении момента времени или интервала времени – сёрфинг, когда пользователь может легко найти необходимый участок испытаний, понять общую картину происходящего (рис. 1). При этом, после загрузки детальной информации становятся видны кратковременные всплески, амплитуда колебаний и т.д. Теорему Котельникова никто не отменял.

2.2. Следующим концептуальным моментом является то, что измерение и регистрация параметров являются конфигурируемыми. Пользователь может задать, как набор параметров, так и установить частоты их записи.

Соответственно, в выбранном варианте размер и структура записи параметров в блоках определяются при каждом старте записи БД. Это накладывает временные ограничения на процесс их вычисления в составе общего времени старта программы.

Запись внутри блока может быть с постоянным и с переменным шагом.

При записи с постоянным шагом в блоке выделяются пакеты равного размера и для каждого параметра на этапе инициализации однозначно устанавливаются

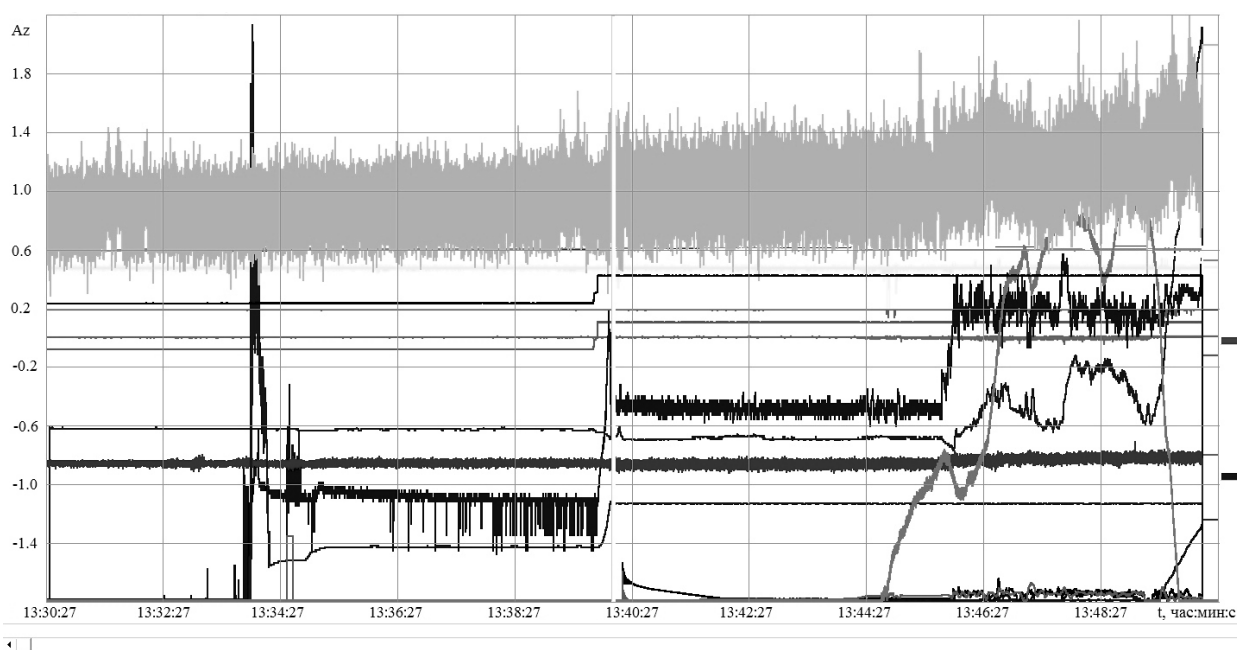


Рис. 1. Область просмотра графиков

ливаются адрес байтов и битов в пакете.

Процесс вычисления однозначный и простой. Но при этом плотность записи не достигает 100 %, хотя на практике, при ограниченном наборе частот записи, отклоняется незначительно. Согласно нашей статистике плотность записи составляет 95 %. Последующая архивация БД для хранения нивелирует данную разницу.

При записи с переменным шагом параметры записываются подряд, и последовательность записи параметров повторяется только в рамках блока. Если, как пример, рассмотреть вариант, что все частоты больше 1 Гц и при этом кратны 1 Гц, то размер блока и, соответственно, таблица индексов внутри блока достигнет десятков Мбайт. Размер блока и таблица индексов формируются при инициализации. В принципе, это не является большой проблемой ни с точки зрения памяти, ни с точки зрения получения индекса – сложность $O(\log n)$. Но при этом плотность записи может приближаться к 100 %.

3. Формирование базы данных

3.1. Вычисление размера блока записи

Критерием формирования блока записи является, с одной стороны, требование повторения последовательности записи, гарантирующее для каждого параметра однозначную связь между меткой времени и размещением значения параметра в блоке, а с другой стороны, минимальный размер блока.

Независимо от того, постоянный ли шаг записи внутри блока или переменный, алгоритм сводится к нахождению наименьшего общего кратного (НОК) всех частот записи параметров.

НОК для N частот находится путём последовательного нахождения НОК результата предыдущих вычислений и очередной частоты записи Ч

$$\text{НОК}_N = \text{НОК}(\text{НОК}(\text{НОК}(Ч_1, Ч_2), Ч_3) \dots, Ч_N).$$

НОК пары чисел находится как их произведение, поделенное на их наибольший общий делитель (НОД)

$$\text{НОК}(Ч_n, Ч_{n-1}) = Ч_n * Ч_{n-1} / \text{НОД}(Ч_n, Ч_{n-1}).$$

Для вычисления НОД применяется алгоритм Евклида, имеющий вычислительную сложность $O(\log \min(Ч_n, Ч_{n-1}))$.

3.2. Формирование оптимальной последовательности заполнения блока

Полученное значение НОК равно количеству срезов - пакетов данных, которые записываются в одном блоке. В свою очередь, период записи пакетов данных равен периоду измерения параметров, установленному в измерительной системе, либо кратен ему.

В случае постоянного шага записи значений параметров длина пакетов одинакова, и для каждого параметра определён адрес в пакете и периодичность записи. На рис. 2 показан фрагмент матрицы распределения битов данных между параметрами (1 – бит используется). Строки при этом соответствуют кванту времени, на котором выполнены измерения, а столбец соответствует 32-битному слову в пакете. Видно, что одно и то же слово в зависимости от момента времени может использоваться для хранения разных параметров. Шаг записи также постоянный на протяжении всей БД. Для наглядности некоторые параметры выделены, в том числе, различные по типу данных и, соответственно, размерности (1 бит, 16 бит, 32 бита).

При переменном шаге принцип сохраняется, параметры записываются последовательно без пауз, но без выделенного адресом слова. Строки «сжимаются», исключая паузы. Длина зависит от количества параметров, которые записываются. Разницы с точки зрения содержащихся в срезе параметров нет. Закономерности на протяжении блока может не быть. Однако между блоками она повторяется.

Здесь стоит отметить, что рассматриваемый блок не является неделимым элементом записи в БД. Он используется исключительно для формирования однозначно повторяющихся последовательностей.

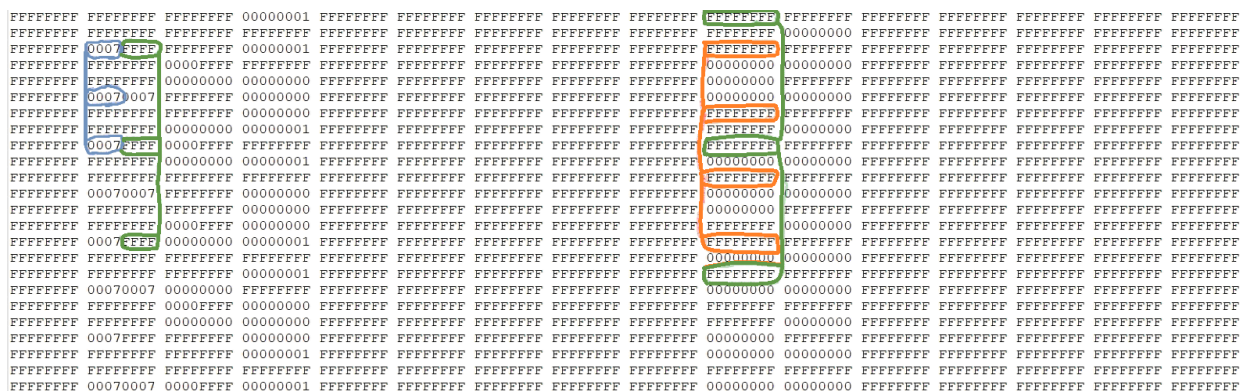


Рис. 2. Фрагмент матрицы заполнения данными блока записи при постоянном шаге записи параметров

Оценим среднее и минимальное значение коэффициента заполнения блока данных при постоянном шаге записи параметров. Неиспользуемые слова данных образуются при частотах, которые не кратны друг другу.

Максимальное количество неиспользуемых слов соответственно равно количеству не кратных друг другу частот. Среднее значение коэффициента заполнения при большом количестве параметров для каждой из частот может быть оценено как половина максимального.

Итого, на практике при 1000 аналоговых параметров количество частот записи не превышает 10...20. Соответственно, среднее значение неиспользуемых слов 5...10. Реальный пример: при 735 параметрах и количестве частот записи, равном 7, покрытие - 95 %.

Заключение

Выполняется регистрация и последующий просмотр 1000+ параметров с частотой до 10 кГц, в том числе при лётных испытаниях ГТД. Время обновления при сёрфинге не превышает 0.5 с независимо от размера базы данных и просматриваемого интервала времени. На рис.3 показан просмотр результатов лётных испытаний.

Литература

1. Cattell, R. *Scalable SQL and NoSQL Data Stores* [Electronic resource] / R. Cattell. – Access mode: <http://www.cattell.net/datastores/Datastores.pdf>. – 12.04.2018.

2. Васнев, Н. В. Система регистрации параметров испытаний сложных изделий на основе документно-ориентированной базы данных [Текст] / Н. В. Васнев, И. А. Шмидт // *Фундаментальные исследования*. – 2016. – № 11-3. – С. 500-504.

3. *Time-series data: Why (and how) to use a relational database instead of NoSQL* [Electronic resource]. – Access mode: <https://blog.timescale.com/time-series-data-why-and-how-to-use-a-relational-database-instead-of-nosql-d0cd6975e87c>. – 12.04.2018.

4. *Time series solutions* [Electronic resource]. – Access mode: <https://docs.microsoft.com/en-us/azure/architecture/data-guide/scenarios/time-series>. – 12.04.2018.

References

1. Cattell, R. *Scalable SQL and NoSQL Data Stores*. Available at: <http://www.cattell.net/datastores/Datastores.pdf> (accessed 12.04.2018).

2. Vasnev, N. V., Shmidt, I. A. *Sistema registratsii parametrov ispytaniy slozhnykh izdelii na osnove dokumentno-orientirovannoi bazy dannykh* [Registering parameters system of complex products testing on the basis of a document-oriented database]. *Fundamental'nye issledovaniya*, 2016, no. 11-3, pp. 500-504.

3. *Time-series data: Why (and how) to use a relational database instead of NoSQL*. Available at: <https://blog.timescale.com/time-series-data-why-and-how-to-use-a-relational-database-instead-of-nosql-d0cd6975e87c> (accessed 12.04.2018).

4. *Time series solutions*. Available at: <https://docs.microsoft.com/en-us/azure/architecture/data-guide/scenarios/time-series> (accessed 12.04.2018).

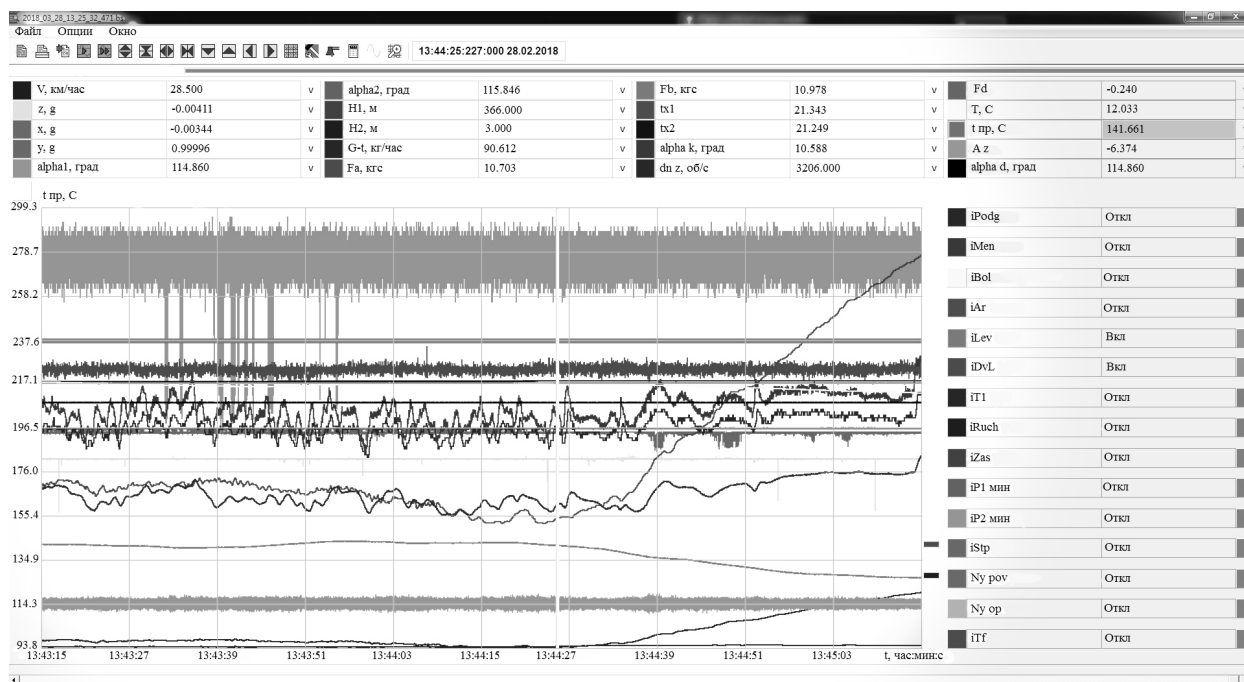


Рис. 3. Просмотр результатов лётных испытаний

Поступила в редакцию 12.06.2018, рассмотрена на редколлегии 25.07.2018

ОПТИМІЗАЦІЯ ОБСЯГУ НЕРЕЛЯЦІЙНОЇ БАЗИ ДАНИХ ТА ЧАСУ ДОСТУПУ ДО ДАНИХ ПРИ МНОЖИННИХ ЧАСТОТАХ РЕЄСТРАЦІЇ ПАРАМЕТРІВ

Д. І. Волков, Н. М. Лопунова

Розглянуто використання нереляційної бази даних для реєстрації параметрів роботи ГТД. Сформульовані узагальнені технічні вимоги до бази даних з урахуванням прикладної специфіки задачі. Наведено обґрунтування обраного формату збереження даних, розрахунки швидкості запису та читання даних для обраних технічних засобів та з урахуванням великого розміру файлів (10+Gb), складність окремих обчислювальних алгоритмів, щільності збереження даних тощо. Враховані різні типи даних та множинні частоти (10+) реєстрації параметрів роботи ГТД (1000+). Презентовано розроблене в АТ «Елемент» програмне забезпечення бортового реєстратора параметрів для роботи із базою даних льотних випробувань ГТД

Ключові слова: реєстрація параметрів роботи двигуна і систем, льотні випробування ГТД, КПА, нереляційні бази даних.

OPTIMIZATION OF DATABASE SIZE AND DATA ACCESS TIME TO STORE PARAMETERS WITH MULTIPLE RECORDING FREQUENCIES

D. I. Volkov, N. N. Lopunova

A non-relational database was developed at JSC Element for the recording operating parameters of GTE and systems during the test flight. The database is modernized version of one which is a part of the GTE test bench. The purpose of the database is the recording and analysis of GTE and systems operating parameters. The parameters are measured (collected) with multiple frequencies in range from fractions of Hz to tens of kHz. Number of recorded operating parameters is 1000+. Correspondingly the database file is huge and may reach tens of Gbytes. Number of different parameters recording frequencies is 10...20 (e.g. 20 parameters – 10 Hz, 30 parameters – 50 Hz, 20 parameters – 10000 Hz). Data reading and chart displaying should be possible as right after the test flight as just in parallel with database recording. There should be no significant delays in getting the data displayed independently from the zoom level chosen. The user should see the result in time comparable with time of the displayed chart perception making the user feel the smooth application operating and not interrupting the process of test flight result comprehension – main purpose of our product.

However, min-max chart displaying (minimal and maximal value are found to render one-pixel vertical lines) requires processing of all the data for the corresponding time interval. Correspondingly, time to process data depends on that. In order to provide data quickly on user's actions the sample is read according to the chart horizontal resolution. Then whole the data is processed in background mode to build min-max chart. This dual approach allows to surf the database quickly losing no details when the surfing is stopped.

Modern computers and SSD drivers provides high I/O performance. However huge files can be neither quickly read nor stored in RAM. Therefore, to read data in parallel with the recording only the choice is to store the data blocks which are either indexed by the time stamp or fixed size. Our solution is based on fixed size approach with the constant offset of each parameter storing. Size of the block is calculated as least common multiple of all the frequencies the parameters are recorded with. According to our statistic the disk usage rate is 95%. As result the time to display the sample for 2000 timestamps is less than 0.5 sec independently for the database size and the displayed duration of the test flight.

Keywords: recording of gas turbine engine and its systems operating parameters, flying test of gas turbine engine, non-relational databases.

Волков Дмитрий Иванович – канд. техн. наук, старший научный сотрудник АО «Элемент», Одесса, Украина, e-mail: odessa@element.od.ua.

Лопунова Нина Николаевна – ведущий инженер-программист, АО «Элемент», Одесса, Украина, e-mail: odessa@element.od.ua.

Volkov Dmitriy Ivanovich – Candidate Technical Science, Senior Reseacher, JSC Element, Odessa, Ukraine, e-mail: odessa@element.od.ua.

Lopunova Nina Nikolaevna – Leading Software Engineer, JSC Element, Odessa, Ukraine, e-mail: odessa@element.od.ua.