Grafana и Splunk как пример решения проблем визуализации данных в современных системах сбора информации

Е.И. Александров, И.Н. Александров, М.А. Минеев[a]

Объединенный институт ядерных исследований,
141980, Россия, Дубна, ул. Жолио-Кюри, д.6

E-mail: [a]mineev@jinr.ru

Настраиваемые среды визуализации данных, такие как Grafana и Splunk, решают большой спектр задач возникающих при разработке Web-приложений для визуализации данных. Такие задачи обычно решались путем написания большого количества программного обеспечения с применением графики. Настраиваемые системы могут работать с различными типами данных, обеспечивают доступ к разным типам источников данных, имеют встроенные возможности по настройке графического представления результатов с использованием встроенных Web-серверов.

На основе опыта работы авторов в группе подсистемы конфигурации и управления (Configuratuion and Control group) системы сбора и обработки данных детектора АТЛАС на Большом Адронном Коллайдере в ЦЕРН представлены мотивирующие причины для перехода к использованию таких фреймворков для обработки логов и для создания приборных панелей мониторинга (дашбордов). Рассмотрены такие достоинства использования фреймворков визуализациии, как быстрота создания работающего прототипа, гибкость создаваемого продукта, небольшой объем требующегося для написания при этом кода и улучшения в визуализации при переходе на более современные версии фреймворков. Также обсуждены трудности, возникающие при их использовании. Возможности фреймворков и их инструментариев рассмотрены на примере использования пакетов Splunk и Grafana.

Ключевые слова: визуализация данных, Splunk, Grafana, системы сбора данных, Web-приложения.

© 2016 Евгений Игоревич Александров, Игорь Николаевич Александров, Михаил Анатольевич Минеев

1.Введение

Функционирование больших компьютерных систем, таких, как система сбора данных (Trigger DAQ - TDAQ) детекторов физики высоких энергий, требует для контроля ее функционирования отслеживания операторами большого количества параметров, в том числе и возможности видеть изменения их во времени. Программное обеспечение для таких целей создается с привлечением огромного количества различных технологий, а сам процесс разработки очень трудоемок. В последнее время появилась новая тенденция: возникли специалилированные настраиваемые среды (фреймворки) для разработки Web-приложений приборных панелей визуализации данных мониторинга (дашбордов). Такие фреймворки обладают встроенным набором технологий от механизма доступа к данным, инструментария для визуализации результатов, предоставления информации пользователю (Web-сервер) до средств создания самих приложений.

Есть несколько аспектов, отличающих разработку с примененем фреймворков от работы без них. С одной стороны, код, написанный для работы с фреймворком, должен быть создан по определенным правилам для его интеграции с фреймворком, с другой стороны сам текст кода получается значительно короче, что облегчает его поддержку, но требует времени на освоение правил. Также все технологии внутри самого фреймворка уже интегрированы друг с другом. Каждый такой фреймворк реализуется под конкретную область, например Splunk – для работы с логами, Grafana – для создания дашбордов метрик.

2. Мотивация применения настраиваемых сред

Мотивирующей причиной перехода к использованию настраиваемых сред является большое число типовых задач требующих создания приложений визуализации данных. Современные системы сбора и обработки данных (ССОД) детекторов физики высоких энергий, таких как ССОД детектора АТЛАС, состоят из большого количества компьютеров, на каждом из которых запущено несколько процессов. Для отслеживания правильности работы системы важную роль играет мониторинг как широкого класса параметров, так и самих физических данных, а также анализ результатов работы системы. Классы данных и параметров, которые необходимо мониторировать, могут меняться в процессе эксплуатации системы, соответственно важно иметь удобный инструментарий для задач визуализации. Дополнительные возможности для создания такого инструментария дают фреймворки. Приведем примеры использования фреймворков, а именно Splunk и Grafana, а также мотивации использования для ССОД детектора АТЛАС. Количество информации, которая поступает от ССОД в виде логов, поступающих из разных компьютеров и процессов настолько велико, что не может просто быть записана в файлы, поэтому для работы с логами в АТЛАС была разработана специализированная система – Log Service [Lehamnn Miotto et al., 2011]. Данные (логи) записываются в базу данных на Oracle. Для возможности поиска и навигации по логам внутри сети АТЛАС Control Network операторами используется написанное на JAVA приложение - Log Manager. Из-за работы файрвола данное приложение недоступно из-за пределов ЦЕРН. В процессе эксплуатации TDAQ было введено дежурство экспертов, находящихся вне ЦЕРН и привлекающихся к разрешению внештатных ситуций по мере необходимости, соответственно понадобился доступ к логам извне ЦЕРН. Возникла необходимость быстрого создания нового приложения на основе клиент-серверных взаимодействий (ERS Browser). Splunk уже был протестирован группой TDAQ АТЛАС для других задач [Yasu, Kazarov, 2014] и показал хорошую производительность. Он и был использован, как фреймворк для создания ERS Browser в кратчайшие сроки в CERN [Kolos, 2014]. ЛИТ ОИЯИ участвовал в улучшении данной системы.

Еще один важный класс задач, где важна удобная и быстро перенастраиваемая система визуализации – это мониторинг. ССОД создает огромное количество метрических данных, которые изменяются в течении времени, например количество данных поступающих на ROS [Borga, 2014] или на HLT [Bains, 2004]. Для хранения таких данный в ССОД была разработана специальная база данных PBEAST [Alexandru D. Sicoe, 2012]. Использование Grafana позволило достаточно быстро создать Web страницы для визуализации наиболее используемой информации. По умолчанию Grafana поддерживает только несколько типов баз данных, но у нее есть возможность создания плагинов для поддержки других баз, что и было сделано для PBEAST. Опыт создания дашбордов для TDAQ был настолько удачен, что сетевая служба АТЛАС для мониторинга трафика предложила использовать Grafana, так как существующая система мониторинга не удовлетворяла службу по некоторым параметрам, а изменения требовали больших трудозатрат. Работа по адаптации WEB страничек системы мониторинга под Grafana произведена в ЛИТ ОИЯИ с участием соответствующих служб АТЛАС.

3. Преимущества использования настраиваимых сред визуализации

На примере работы с пакетами Splunk и Grafana рассмотрим те преимущества, которые дает использование настраиваемых сред. Как уже было указано, первый из пакетов применяется для обработки логов и визуализации соответствующей информации, Grafana используется для создания панелей приборов. В обоих случаях основное преимущество – это быстрое получение удовлетворяющей пользователя работающей системы за счет настраивания имеющихся баз данных для работы с фреймворком, задания необходимых для работы параметров и написания по заданным в фреймворке правилам относительно небольшого количества кода, учитывающего особенности создаваемой системы визуализации.

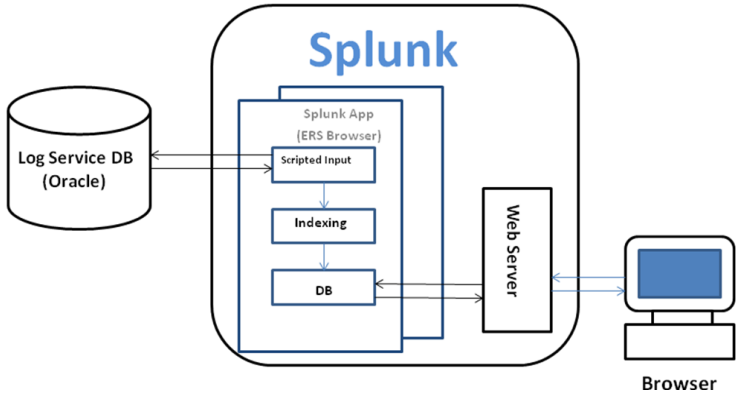

Рис.1 ERS Browser как приложение Splunk. Общая схема.

Splunk, базирующийся на библиотеке Backbone.js, обеспечивающей парадигму близкую MVC (Model-View-Controller - «модель-представление-контроллер») для создания одностраничных Web-приложений, публикуемых на встроенном Web сервере, использовался в АТЛАС для расчета и получения в виде диаграмм статистических параметров, таких как распределение логов по тем или иным параметрам, например по имени хостов или приложений, по коду серьезности – info, warning, error, fatal и.т.д. Splunk имеет встроенные возможности для обработки статистической информации по логам и создания таких диаграмм. Без использования фреймворков их написание занимает существенно больше времени.
ERS Browser написан в виде приложения Splunk – Splunk apps [Developing Views and Apps for Splunk Web]. Для создания приложения и настройки Web-интерфейса использовался разрабо-

танный для Splunk язык - Search Processing Language (SPL). Общая схема приложения приведена на рис.1. При реализации приложения была учтена такая особенность, как вложенные логи, которые представляют из себя цепочку логов порождаемую разными процессами, но имеющих одну и ту же причину возникновения. Результат представления вложенных логов можно увидеть на рисунке 2.

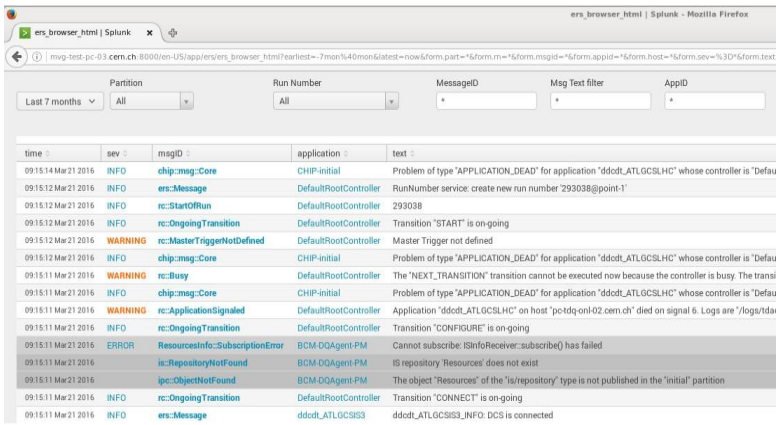

Рис. 2. ERS Browser. Визуализация вложенных логов.

Авторы статьи участвовали в реализации функционала, отсутствующего в используемом визуальном инструментарии Splunk. Возникла необходимость получения логов, которые лежат в заданном интервале времени с центром в виде выбираемого пользователем лога с помощью контекстного меню. Данный функционал был реализован путем написания скрипта с использованием сторонней библиотеки context.js и jQuery.

Доступность технологий, уже встроенных в фреймворк – одно из больших преимуществ таких сред, позволяющих быстро решать задачи увеличения функционала в соответствии с запросами пользователей. Динамическое развитие таких фреймворков означает также, что с появлением каждой новой версии облегчается процесс разработки и улучшаются возможности визуализации данных. Особенно ярко это можно увидеть на примере Grafana, в настоящее время выходит уже Grafana 4 с более широкими возможностями, которые можно использовать при построении дашбордов без написания дополнительного кода.

4. Трудности использования настраиваимых сред визуализации

Трудности при работе с такими средами во многом связаны с тем, что для большего соответствия проблемной области они используют уникальные компоненты. Так в Splunk использован собственный язык SPL, освоение которого требует времени. Также может оказаться, что конкретная задача не ложится автоматически в предлагаемые фреймворком парадигмы. Например при работе со Splunk необходимо было решить проблему связанную с форматом логов применяемых в Log Service. Внутренняя база данных Splunk, куда записываются логии, полученные после их индексации, имеет свою собственную архитектуру и является нереляционной. В нашем же случае некоторые логи были т.н. вложенными, т.е. такие логи, которые содержат в себе несколько добавочных, часть параметров всех членов вложенного лога одинакова (время, имя приложения и т.д.). Сложности возникали при поиске данных: в результатах поиска вложенные логи должны всегда приходить вместе (Рис.2) для правильного анализа причины. Эта проблема была решена переписыванием всех поступающих в систему логов в виде плоского

списка (вложенные логи представляются также, как и обычные) с добавлением 2 добавочных полей: chained, показывающего является ли лог обычным или вложенным. Учитывалось также является ли лог родительским или потомком. Использовался специальный marker – параметр для группирования всех членов вложенного лога (родителей и потомков), для всех членов группы он одинаков. То или иное представление вложенных логов в виде плоского списка может усложнять или упрощать последующую работу с базой логов с помощью SPL. Сам запрос к внутренней базе исполузет возможности SPL по поиску по добавочным полям. Еще одно возможное, обсуждавшееся решение этой проблемы – поиск групп вложенных логов, с помощью добавочного менеджера поиска (search manager), который бы искал недостающую информацию после работы основного менеджена поиска.

Grafana - это свободно распространяемая система и, как и у большинства таких систем, у нее есть проблемы с полнотой документации. В документации рассмотрены наиболее простые случаи, но если требуется сделать что-то более продвинутое, то могут возникнуть проблемы. Система мониторинга трафика — это динамическая система, конфигурация которой меняется с течением времени. Динамическая конфигурация не позволяет создать приборную доску (дашборд) через WEB интерфейс самой Grafan-ы, как это было сделано для системы ССОД АТЛАС, но существует возможность создавать дашборд динамически на Javascript. В документации по Grafana приводится только простой случай создания дашборда через скрипт и не приводится описания структуры используемой для этого. Другой сложностью при адаптации существующей системы мониторинга является сама структура системы. Она состоит из нескольких частей написанных на разных языках программирования. При изменении части ответственной за отображение данных необходимо изменить и способ хранения данных, т. е. структуру хранения данных приходится адаптировать под Grafana.

5.Заключение

Применение настраиваемых сред визуализации данных Splunk и Grfana для построения Web приложений делающих акцент на визуализации данных показало их высокую эффективность. Код в обоих случаях получается компактным, что значительно снизило затраты времени на созданние приложений и их последующую поддержку. Внутри фреймворков уже решена проблема интеграции различных технологий, что позволяет разработчику дашбордов сконцентрироваться на проблемной части задачи и уйти от решения рутинных вопросов взаимодействия библиотек и пакетов. Встроенные возможности по графическому представлению данных и доступу к источникам данных позволяет использовать уже готовые шаблоны. Во многих случаях их оказывается достаточно для большинства задач. В обоих пакетах предусмотрена возможность создания дашбордов с использованием Web – интерфейсов самих пакетов. Такие дашборды могут быть созданы за короткое время и поддержка их не занимает больших усилий. В случае, если конечному пользователю не хватает встроенных возможностей, оба пакета предоставляют возможность написания своего кода с применением сторонних библиотек. Специфические компаненты (как в случае с SPL) для обеих систем требуют времени на изучение, но их применение на практике по сравнению с использованием стандартных методов оказывается обычно более эффективным.

Список литературы

Пример использования Splunk для анализа логов. [Электронный ресурс]. URL: https://habrahabr.ru/post/160197/ (Дата обращения: 07.11.2016).
Primer ispolzovaniya Splunk dlya analiza logov. [Electronic resource]. (In Russ.) https://habrahabr.ru/post/160197/ (accessed 07.11.2016)

Alexandru D. Sicoe et al, A persistent back-end for the ATLAS TDAQ online information service (P-BEAST) // J. Phys.: Conf. Ser. 368 (2012) 012002, URL http://iopscience.iop.org/article/10.1088/1742-6596/368/1/012002/meta (Accessed: 07.11.2016)

Baines J.T. et al. An Overview of the ATLAS High Level Trigger Dataflow and Supervision // IEEE Transactions On Nuclear Science, June 2004. Vol. 51, No. 3, JUNE 2004 URL https://hal.archives-ouvertes.fr/in2p3-00145340/document (Accessed: 07.11.2016)

Borga A. et al. Evolution of the ReadOut System of the ATLAS experiment // Technology and Instrumentation in Particle Physics 2014, 2-6 June, 2014 Amsterdam, the Netherlands URL https://cds.cern.ch/record/1710776/files/ATL-COM-DAQ-2014-063.pdf (Accessed: 07.11.2016)

Developing Views and Apps for Splunk Web. [Electronic resource]. URL: http://docs.splunk.com/Documentation/Splunk/latest/AdvancedDev/Whatsinthismanual (Accessed: 07.11.2016).

Grafana Home page. [Electronic resource]. URL: http://grafana.org (Accessed: 07.11.2016).

Kolos S.et al. The Error Reporting in the ATLAS TDAQ System // 16th International workshop on Advanced Computing and Analysis Techniques in physics (ACAT), Prague, Czech Republic, 1 5 Sep 2014, J. Phys.: Conf. Ser. 608 (2015) 012004. URL . http://iopscience.iop.org/article/10.1088/1742-6596/608/1/012004/pdf (Accessed: 07.11.2016)

Lehamnn Miotto G. et al. A revised design and implementation of the ATLAS log service package // J.Phys.Conf.Ser. 331 (2011) 042037 2011. URL . http://iopscience.iop.org/article/10.1088/1742-6596/331/4/042037/pdf (Accessed: 07.11.2016)

Splunk Inc. Home Page [Electronic resource]. URL https://www.splunk.com

Yasu Y., Kazarov A. Performance of Splunk for the TDAQ Information Service at the ATLAS experiment // Real Time Conference (RT), 2014 19th IEEE-NPSS 26-30 May 2014. URL http://ieeexplore.ieee.org/document/7097473/?reload=true&tp=&arnumber=7097473 (Accessed: 07.11.2016)

Grafana and Splunk as an example of the data visualization solution for the modern data taking systems

E.I.Alexandrov, I.N.Alexandrov, M.A.Mineev [a]

Joint Institute for Nuclear Research,
6 Joliot Curie, Dubna, 141980, Russia

E mail: mineev@jinr.ru

The adaptable data visualization frameworks such as Grafana and Splunk solve a wide spectrum of problems in the Web application development for data visualization, for example for the dashboard creation. Earlier, such kind problems were solved by means of coding a large amount of software including packages with graphics. The adaptable frameworks provide access to the different data source types. They have a built in possibility for the graphical data presentation of the results with their own Web servers.

The motivation to use such frameworks for logout systems and for dashboards is presented on the basis of experience the authors have gained working in the Configuration and Control group of the Data Acquisition System of ATLAS detector in Large Hadron Collider at CERN. The advantages to use visualization frameworks such as a fast time to get alive system, a flexibility of the developed product, a short amount of implementation code, easy improvements of the data visualization on the base of the new versions of the framework are presented. The difficulties appearing in the process of the work are discussed as well. The problem of studying the framework possibilities and corresponding tools is presented on the basis of examples of applying Splunk and Grafana.

Keywords: data visualization, Splunk, Grafana, data acquisition systems, web applications

© 2016 Evgeny I. Alexandrov, Igor N. Alexandrov, Mikhail A. Mineev