

ОБЕСПЕЧЕНИЕ ОПЕРАТИВНОГО КОНТРОЛЯ И ЭФФЕКТИВНОЙ АВТОНОМНОЙ РАБОТЫ СУПЕРКОМПЬЮТЕРНОГО КОМПЛЕКСА МГУ¹

*А.С. Антонов, Вад.В. Воеводин, А.А. Даугель-Дауге, С.А. Жуматий,
Д.А. Никитенко, С.И. Соболев, К.С. Стефанов, П.А. Швец*

В НИВЦ МГУ разрабатывается система для обеспечения оперативного контроля и поддержки эффективного автономного функционирования суперкомпьютерных комплексов. Данная система внедряется в Суперкомпьютерном центре МГУ. В работе описывается опыт установки, настройки и эксплуатации системы для контроля работы суперкомпьютера «Чебышёв».

Ключевые слова: суперкомпьютер, граф, графовая модель, мониторинг, оперативный контроль, автономная работа, Octotron.

Введение

Система Octotron, разрабатываемая в НИВЦ МГУ имени М.В. Ломоносова, предназначена для обеспечения оперативного контроля функционирования суперкомпьютерных комплексов [1, 2]. Система отслеживает наступление нештатных ситуаций в работе всех компонентов комплекса. При возникновении таких ситуаций система выполняет определенный набор действий. Оперативное реагирование на сбои различного рода позволяет минимизировать их негативные последствия, тем самым обеспечивая эффективную автономную работу комплекса.

Кратко напомним основные понятия и идеи, заложенные в систему Octotron. Система работает на основе модели суперкомпьютерного комплекса, представленной в виде мультиграфа [3]. Вершины графа в модели соответствуют физическим или логическим компонентам суперкомпьютера, работу которых необходимо контролировать (вычислительные узлы, источники бесперебойного питания, очереди, лицензии на ПО и т.д.), а дуги — связям между компонентами («состоит из», «обеспечивает электропитанием», «соединены сетью Infiniband» и т.д.). С каждой из вершин связан набор атрибутов — характеристик состояния компонентов (температура процессора, объем памяти, число заданий в очереди и т.д.). Значения атрибутов поставляются штатными системами мониторинга суперкомпьютера либо получаются обращением к внешним интерфейсам компонентов. При изменении значений атрибутов срабатывают зависимые от них правила, определяющие факт наличия нештатной ситуации. В случае определения такой ситуации вызывается определенный набор действий, т.е. выполняется реакция.

Новизна подхода, реализованного в системе Octotron, заключается в использовании модели суперкомпьютерного комплекса в качестве входных данных контролирующей системы: фактически, суперкомпьютер начинает самостоятельно контролировать собственную работу. Среди функциональных аналогов можно назвать решения известных вендоров (HP OpenView [4], IBM xCAT [5]), китайскую систему Iaso [6], созданную спе-

¹ Статья рекомендована к публикации программным комитетом Международной научной конференции «Параллельные вычислительные технологии – 2015».

циально для суперкомпьютера Tianhe-2, и российскую разработку «Система Автоматического Отключения Оборудования» (CAOO) [7] компании Т-Платформы. Упомянутые разработки являются закрытыми и не являются универсальными: возможности первых двух систем во многом завязаны на аппаратуру собственного производства, для Iaso необходима модификация ядра ОС и установка дополнительных компонентов ПО, система CAOO контролирует только инфраструктуру вычислительного комплекса.

В настоящее время система Octotron проходит апробацию в Суперкомпьютерном комплексе МГУ. Система уже находится в штатной эксплуатации на суперкомпьютере «Чебышёв». С начала 2015 года система запущена в опытную эксплуатацию на суперкомпьютере «Ломоносов», где она работает параллельно штатной системой автоматического отключения оборудования. Данная статья посвящена особенностям настройки и эксплуатации системы Octotron для обеспечения оперативного контроля работы суперкомпьютера «Чебышёв».

Статья организована следующим образом. Раздел 1 посвящен описанию модели суперкомпьютера «Чебышёв», построенной в системе Octotron. В разделе 2 приведен список источников данных о состоянии суперкомпьютера, используемых системой Octotron для поддержания автономной работы суперкомпьютера. В разделе 3 описываются правила системы Octotron для определения сбоев в работе суперкомпьютера, а в разделе 4 — методы реагирования системы на сбои. В заключении приводится краткое описание текущего состояния системы Octotron и планы ее развития.

1. Модель суперкомпьютера «Чебышёв»

Модель суперкомпьютера «Чебышёв» (625 вычислительных узлов, 5 000 процессорных ядер) содержит 10 228 вершин, 25 698 дуг и 205 044 атрибута. В модели отражены следующие компоненты суперкомпьютера [3]:

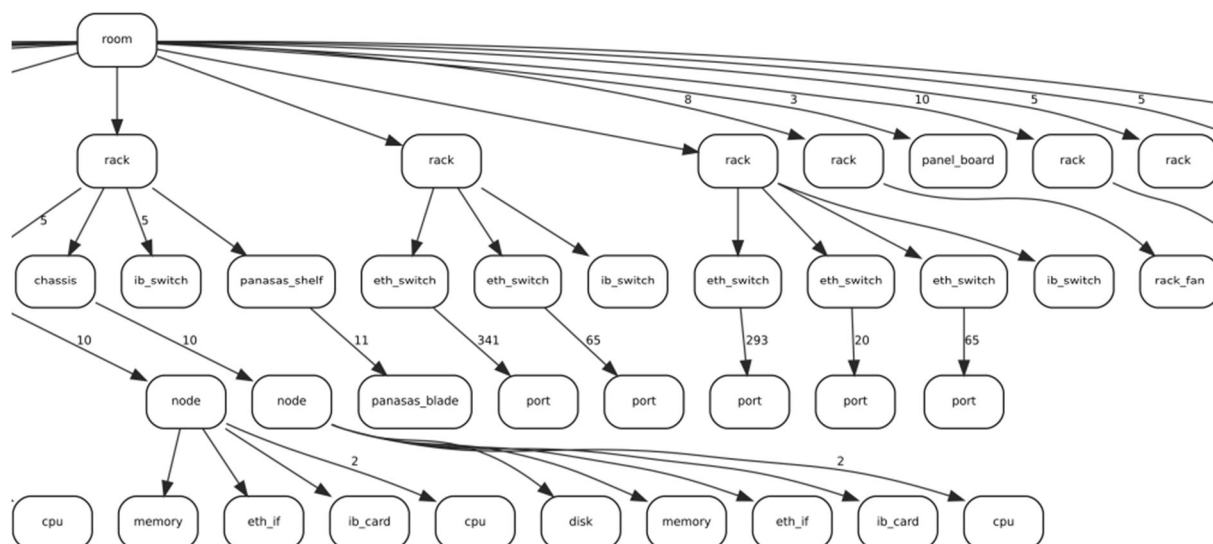


Рис. 1. Фрагмент модели суперкомпьютера «Чебышёв»: физические связи между компонентами

- система электропитания (источники бесперебойного питания, модули с батареями);
- система охлаждения (холодильные установки, воздушные кондиционеры, мониторинг среды);
- управляющая часть (узлы доступа, очереди задач);

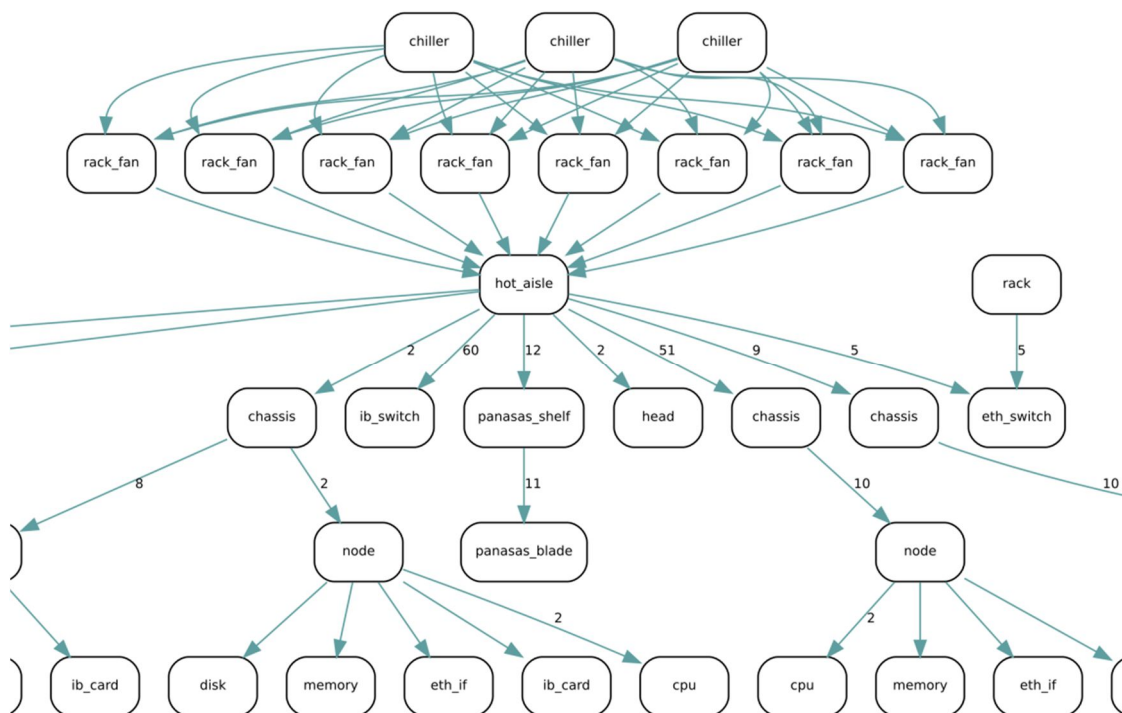


Рис. 2. Фрагмент модели суперкомпьютера «Чебышёв»: охлаждение

- вычислительная часть (шасси, узлы, диски, память);
- файловая система (разные компоненты для разных типов ФС);
- сеть Ethernet (коммутаторы, порты);
- сеть Infiniband (коммутаторы, менеджер сети).

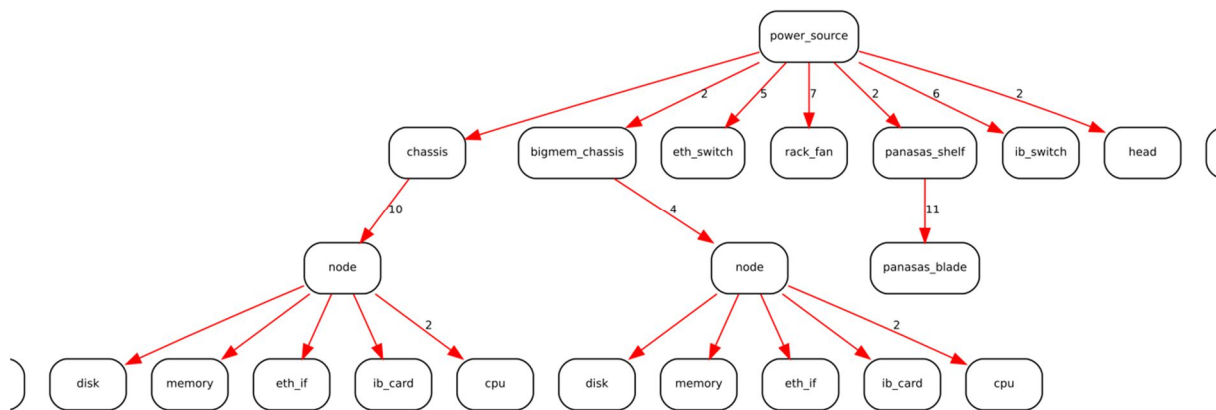


Рис. 3. Фрагмент модели суперкомпьютера «Чебышёв»: электропитание

Следующие типы связей реализованы в модели суперкомпьютера «Чебышёв»:

- содержит;
- охлаждает;
- соединены сетью Ethernet;
- соединены сервисной сетью;
- соединены сетью Infiniband;
- включает в себя;
- обеспечивает электропитанием.

На рис. 1 приведен фрагмент модели суперкомпьютера «Чебышёв», содержащий компоненты, связанные дугами типа «содержит». Количество вершин графа и дуг между ними велико, однако суперкомпьютер имеет в целом довольно регулярную структуру, поэтому модель содержит множество изоморфных подграфов. На рисунках подобные подграфы объединяются в один подграф, при этом дуге, ведущей к этому подграфу, приписывается число — количество объединенных подграфов.

Аналогичным образом отображаются фрагменты модели, описывающие охлаждение вычислительного комплекса (рис. 2) и электропитание его компонентов (рис. 3).

Разработка метода визуализации графа модели потребовала выполнения отдельного исследования. Подсистема визуализации графа на текущий момент позволяет получить компактные изображения, удобные для визуальной проверки корректности модели, для большинства типов связей, за исключением тех, которые образуют циклы в графе (как, например, связь «соединены сетью Infiniband»). Созданная подсистема стала существенным подспорьем при создании модели суперкомпьютера «Чебышёв».

В настоящий момент ведется активная разработка новой интерактивной подсистемы визуализации. Она позволяет разворачивать и свертывать вершины графа, обеспечивая при необходимости доступ и просмотр списка атрибутов любой вершины, а также поддерживает различные варианты группировки вершин. Пример интерфейса новой подсистемы и отображения фрагмента модели приведен на рис. 4.

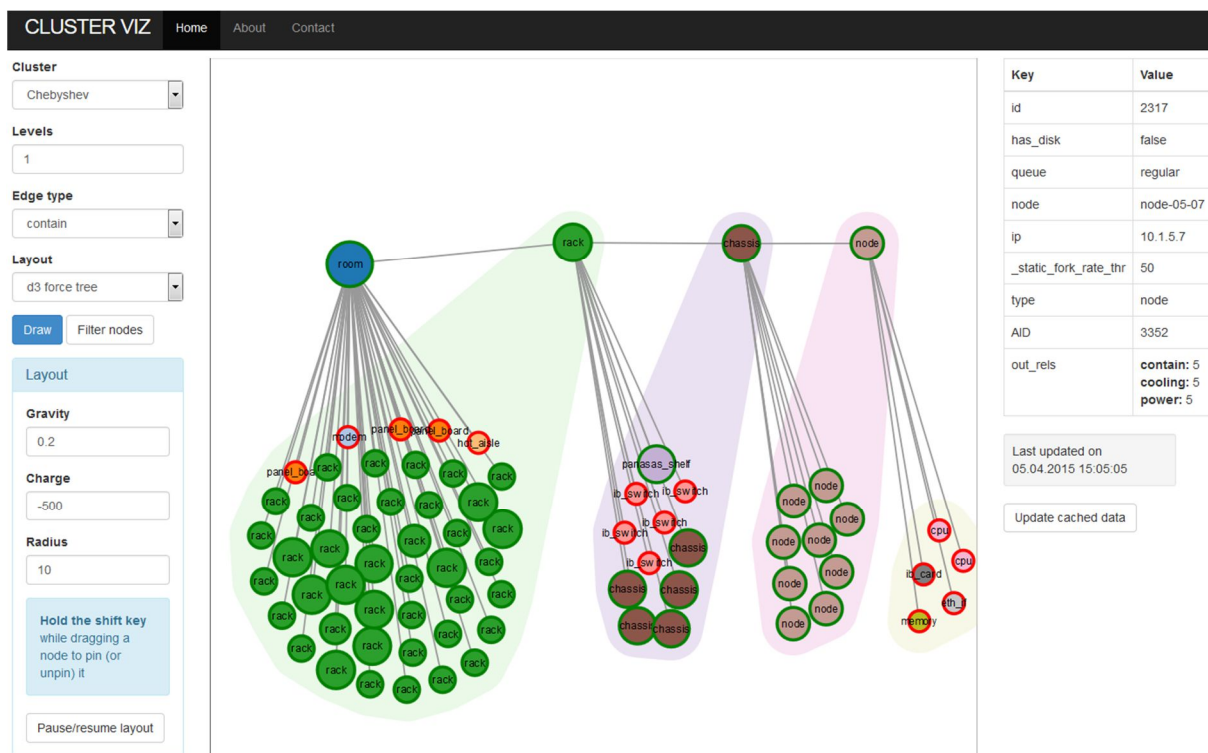


Рис. 4. Фрагмент модели суперкомпьютера «Чебышёв»: новые возможности отображения графа

2. Источники данных о состоянии суперкомпьютера

Основным поставщиком данных о работе компонентов суперкомпьютера «Чебышёв» является система мониторинга на основе collectd. С инфраструктурного оборудования данные поступают по протоколу SNMP. Все собираемые данные периодически импорти-

руются в Octotron. Заметим, что сама система Octotron жестко не привязана к какой-либо конкретной системе мониторинга. Использована может быть любая система мониторинга, а для ее совместной работы с системы Octotron потребуется разработка несложного модуля импорта данных. Подчеркнем также, что Octotron самостоятельно не опрашивает оборудование и не обращается к датчиками напрямую. В системе есть средства получения сигналов SNMP traps, отправляемые поддерживающим стандарт SNMP оборудованием в критических случаях. Для «Чебышёва» таким образом определяются критические сбои в системе электропитания.

С головных машин суперкомпьютера «Чебышёв» раз в 10 минут снимаются следующие данные:

- число активных ssh-сессий пользователей;
- число активных лицензий на ПО;
- количество задач в каждой из очередей: общее, ожидающих, готовящихся к запуску, выполняющихся, завершенных;
- число процессоров: общее, доступных для запуска задач, заблокированных;
- баланс счета GSM-модема, подключенного к одной из головных машин и используемого для рассылки экстренных SMS-оповещений.

Данные, собираемые раз в 10 минут со всех вычислительных узлов:

- температура внутри узла;
- температура каждого процессора;
- идентификатор выполняющейся на узле задачи;
- состояние файловой системы;
- состояние памяти (общий объем, объем свободной/занятой памяти);
- состояние карты Infiniband (счетчики переданных/принятых пакетов, ошибки);
- состояние карты Ethernet (ошибки);
- другие системные данные: средняя загрузка узла, количество процессов-зомби и т.д.

Для узлов с жесткими дисками (около 100 шт.) дополнительно собирается информация SMART о состоянии HDD. Кроме того, один раз в час на каждом узле проверяется работа сервиса ssh, видимость в сети Infiniband и работоспособность MPI.

Для общей файловой системы на основе Panasas раз в 10 минут собираются общие данные (объемы свободного/занятого пространства, производительность), а также статус и загрузка каждого blade-модуля (всего их 132). С той же периодичностью собираются данные с коммутаторов Ethernet.

Информация о работе климатической системы вычислительного комплекса собирается чаще — 1 раз в минуту. Она включает в себя значения с нескольких датчиков температуры и влажности воздуха в помещении, а также состояние каждого из 8 кондиционеров (температура воздуха и охлаждающей жидкости на входе и на выходе, различные предупреждения). С той же периодичностью собираются данные с пяти источников бесперебойного питания – около 60 параметров с каждого: состояние внешнего питания, собственных режимов, аккумуляторных батарей и т.д.

3. Правила определения сбоев в работе суперкомпьютера

Правила в системе Octotron представляют собой функции, имеющие доступ к атрибутам вершин графа модели суперкомпьютера. Правила могут получать доступ к зна-

чениям атрибутов соседних вершин по заданному типу связи. Octotron позволяет формализовать правила нескольких типов:

- сравнение значения датчика с константой. Пример: выход температуры компонента за пределы заданного порога;
- наличие аварийных значений одновременно у нескольких датчиков. Пример: сообщение о повышении температуры несколькими датчиками горячего коридора;
- значения датчиков у смежных (по графу модели) компонентов не соответствуют друг другу. Пример: порты на двух концах связи Ethernet находятся в разных режимах;
- сохранение определенного уровня значения датчика в течение заданного промежутка времени. Пример: уровень загрузки узла может ненадолго превысить штатные значения, но достаточно продолжительный высокий уровень загрузки свидетельствует о проблеме на узле;
- получение ошибочных значений датчика несколько раз подряд. Пример: узел более трех раз подряд не проходит проверку доступа по ssh.

Все сбои в работе суперкомпьютера, определяемые правилами, имеют свой уровень критичности. В настоящее время мы используем 4 уровня: Info (информация), Warning (предупреждение), Danger (опасность), Critical (авария). Сбои уровня Critical в основном связаны с повышением температуры воздуха в помещении, горячем коридоре и на компонентах. Подобные сбои могут нанести существенные повреждения оборудованию и помещениям вычислительного комплекса. К сбоям уровня Danger отнесены ситуации, существенно затрагивающие работу всех пользователей суперкомпьютера, например, отказы системы хранения данных, проблемы с очередями, а также незначительные отклонения в работе климатической и энергетической инфраструктуры. Сбои уровня Warning — локальные проблемы на узлах.

Для контроля работы суперкомпьютера «Чебышёв» в настоящий момент используется около 160 правил. Вот некоторые из них:

- баланс счета GSM-модема близок к порогу отключения;
- сбои в работе двух или трех холодильных установок;
- значительный рост ошибок на сетевых интерфейсах;
- слишком малое количество пользовательских сессий на головной машине;
- слишком большое число заблокированных узлов;
- рассинхронизация времени на узлах;
- значение LoadAVG на «свободном» узле превышает значение 3.

4. Методы реагирования на сбои

Если правило определило сбой в работе суперкомпьютера, система может выполнить некоторую реакцию. По умолчанию для каждого события производится запись о нем в лог-файл и отправка e-mail администраторам. Информация о событиях уровня Critical дублируется по SMS. В критических случаях (срабатывание пожарной сигнализации, резкое повышение температуры в помещении, низкий уровень заряда аккумуляторных батарей при питании от них) суперкомпьютер может быть автоматически выключен. В случае недоступности вычислительных узлов по протоколу ssh или неработоспособности на них сервисов MPI они автоматически выводятся системой из счетного поля.

```

*** reported events ***

13, DANGER, "ntpd drift on node is too big"
2, DANGER, "ems sensor: front temp is very high"

4, WARNING, "bad system temp on node"
1, WARNING, "zombies present on node for last 1000 seconds"

13, RECOVER, "ntpd drift on node is ok"
3, RECOVER, "system temp on node is ok"
2, RECOVER, "ems sensor: front temp is back to normal"

*** suppressed events ***

8, DANGER, "too many free cpus"
    
```

Рис. 5. Пример дайджеста событий за сутки

На практике множество сбоев, обнаруживаемых системой Octotron в ходе работы суперкомпьютера, не имеют критического характера и позволяют продолжать штатное функционирование вычислительного комплекса. Чтобы уменьшить интенсивность потока сообщений администраторам суперкомпьютера, в системе предусмотрена возможность блокировки повторяющихся оповещений. Она применяется в абсолютно типичных ситуациях: проблема не является критичной, о ней известно администраторам, но по каким-либо причинам устранить ее не удается.

Еще один полезный для администратора суперкомпьютера сервис – ежедневный дайджест событий, рассылаемый по e-mail. На рис. 5 приведен пример дайджеста с перечнем событий, случившихся на суперкомпьютере «Чебышёв» 11 ноября 2014 г. В секции «reported events» перечислены актуальные обнаруженные сбои, в секции «suppressed events» – сбои, для которых была отключена отправка отдельных сообщений.

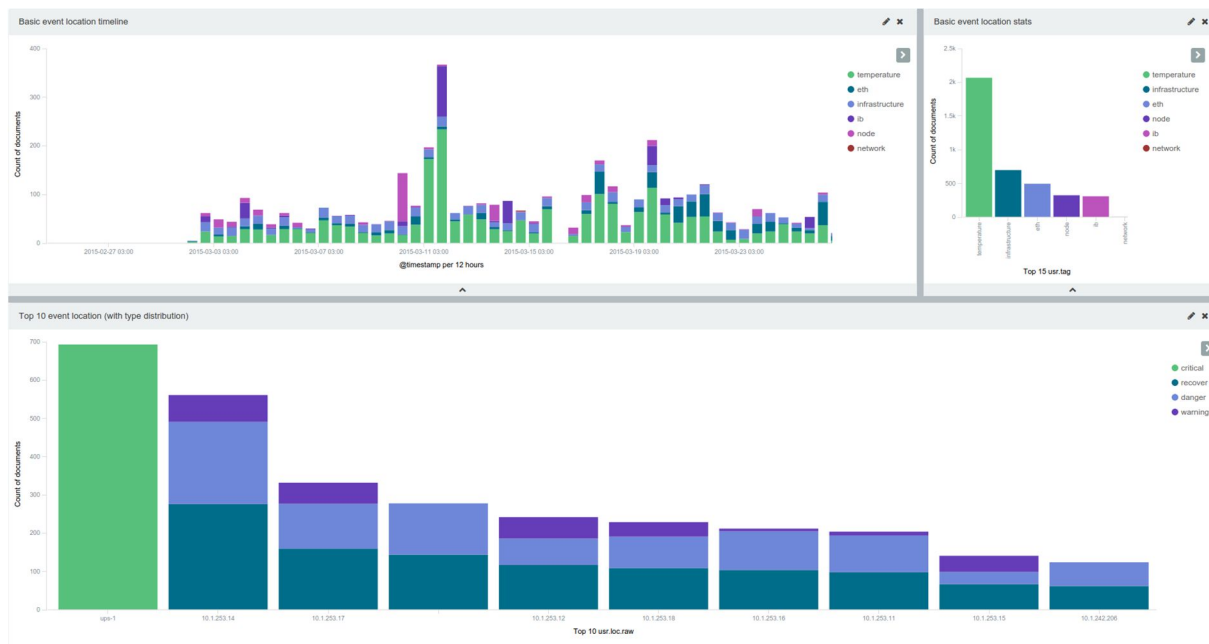


Рис. 6. Графическое представление потока событий, фиксируемого системой Octotron.

Лог-файл системы Octotron ведется в формате JSON. Это позволяет использовать внешние утилиты для его обработки. С помощью набора инструментов logstash, elasticsearch и kibana было реализовано средство построения статистики и базового визуального анализа потока событий, фиксируемого системой Octotron (рис. 6). Левый верхний график отображает число событий различного типа за интервал времени (каждому типу событий соответствует свой цвет), правый верхний — общее число событий различных типов в исследуемом интервале, нижний график — распределение событий по уровню критичности (цвет) и их источникам (горизонтальная ось).

Заключение

Система Octotron разрабатывается с конца 2012 г., и уже сейчас она успешно эксплуатируется в Суперкомпьютерном комплексе МГУ. На текущий момент она полностью контролирует работу суперкомпьютера «Чебышёв». В тестовом режиме, дублируя штатную систему автоматического отключения оборудования, Octotron работает на суперкомпьютере «Ломоносов».

Развитие системы ведется одновременно в нескольких направлениях. Так, статистика сбоев суперкомпьютера представляет исключительно ценный материал для анализа и прогнозирования его поведения. Методы определения типичных сбоев, выраженные единообразно в терминах модели Octotron, могут отчуждаться и тиражироваться; крайне перспективным направлением представляется создание коллективного банка неисправностей суперкомпьютерных комплексов и методов реагирования на них. Наличие в основе системы модели суперкомпьютера позволяет реализовать различные средства визуализации сбоев для оперативного поиска и устранения неисправностей. Сама по себе разработка модели суперкомпьютера является нетривиальным процессом, который, однако, может быть автоматизирован [8]. Важно, что вся подобная функциональность может быть реализована с помощью подключаемых к системе модулей, в то время как ядро системы Octotron уже полнофункционально, удовлетворяет заданным при разработке требованиям и выполняет поставленные задачи.

Разработанная система доступна под открытой MIT лицензией [9, 10].

Работа выполнена при финансовой поддержке РФФИ, грант №12-07-33047.

Литература

1. Антонов, А.С. Разработка принципов построения и реализация прототипа системы обеспечения оперативного контроля и эффективной автономной работы суперкомпьютерных комплексов / А.С. Антонов, Вад.В. Воеводин, Вл.В. Воеводин, С.А. Жуматий, Д.А. Никитенко, С.И. Соболев, К.С. Стефанов, П.А. Швец // Вестник УГАТУ. — 2014. — Т. 18, № 2. — С. 227–236.
2. Соболев, С.И. Суперкомпьютер в штатном режиме / С.И. Соболев // Открытые системы. — 2014. — № 8.
3. Швец, П.А. Об одном подходе к моделированию суперкомпьютерных комплексов / П.А. Швец, Вад.В. Воеводин, С.И. Соболев // Научный сервис в сети Интернет: многообразие суперкомпьютерных миров: Труды Международной суперкомпьютерной конференции (22–27 сентября 2014 г., г. Новороссийск). — Изд-во МГУ Москва, 2014. — С. 197–204.

4. HP OpenView. URL: http://www.openview.hp.com/solutions/ams/ams_bb.pdf (дата обращения: 06.04.2015).
5. xCAT, An extreme cluster/cloud administration toolkit. URL: http://sourceforge.net/p/xcat/wiki/Main_Page/ (дата обращения: 06.04.2015).
6. Lu K. et al. Iaso: an autonomous fault-tolerant management system for supercomputers //Frontiers of Computer Science. — 2014. — Т. 8. — №. 3. — С. 378–390. DOI: 10.1007/s11704-014-3503-1
7. Программное обеспечение компании Т-Платформы. URL: <http://www.t-platforms.ru/products/software.html> (дата обращения: 06.04.2015).
8. Воеводин, Вад В. Автоматическое определение и описание сетевой инфраструктуры суперкомпьютеров / Вад.В. Воеводин, К.С. Стефанов // Вычислительные методы и программирование: Новые вычислительные технологии — 2014. — Т. 15, № 3. — С. 560–568.
9. Полный исходный код Octotron. URL: https://github.com/srcc-msu/octotron_core (дата обращения: 06.04.2015).
10. Рабочее окружение Octotron для создания модели на языке Python. URL: <https://github.com/srcc-msu/octotron> (дата обращения: 06.04.2015).

Антонов Александр Сергеевич, к.ф.-м.н., ведущий научный сотрудник лаборатории параллельных информационных технологий НИВЦ, Московский государственный университет имени М.В. Ломоносова (Москва, Российская Федерация), asa@parallel.ru.

Воеводин Вадим Владимирович, к.ф.-м.н., научный сотрудник лаборатории параллельных информационных технологий НИВЦ, Московский государственный университет имени М.В. Ломоносова (Москва, Российская Федерация), vadim@parallel.ru.

Даугель-Дауге Артем Александрович, студент факультета Вычислительной математики и кибернетики, Московский государственный университет имени М.В. Ломоносова (Москва, Российская Федерация), daugeldaage@gmail.com.

Жуматий Сергей Анатольевич, к.ф.-м.н., ведущий научный сотрудник лаборатории параллельных информационных технологий НИВЦ, Московский государственный университет имени М.В. Ломоносова (Москва, Российская Федерация), serg@parallel.ru.

Никитенко Дмитрий Александрович, к.ф.-м.н., научный сотрудник лаборатории параллельных информационных технологий НИВЦ, Московский государственный университет имени М.В. Ломоносова (Москва, Российская Федерация), dan@parallel.ru.

Соболев Сергей Игоревич, к.ф.-м.н., старший научный сотрудник лаборатории параллельных информационных технологий НИВЦ, Московский государственный университет имени М.В. Ломоносова (Москва, Российская Федерация), sergeys@parallel.ru.

Стефанов Константин Сергеевич, к.ф.-м.н., старший научный сотрудник лаборатории параллельных информационных технологий НИВЦ, Московский государственный университет имени М.В. Ломоносова (Москва, Российская Федерация), cstef@parallel.ru.

Швец Павел Артемович, программист лаборатории параллельных информационных технологий НИВЦ, Московский государственный университет имени М.В. Ломоносова (Москва, Российская Федерация), shpavel@parallel.ru.

Поступила в редакцию 9 марта 2015 г.

SECURING OF ACTIVE CONTROL AND EFFICIENT AUTONOMOUS OPERATING OF MSU SUPERCOMPUTING CENTER

A.S. Antonov, Research Computing Center, Moscow State University (Moscow, Russian Federation) asa@parallel.ru,

Vad.V. Voevodin, Research Computing Center, Moscow State University (Moscow, Russian Federation) vadim@parallel.ru,

A.A. Daugel-Dauge, Faculty of Computational Mathematics and Cybernetics, Moscow State University (Moscow, Russian Federation) daugeldauge@gmail.com,

S.A. Zhumatiy, Research Computing Center, Moscow State University (Moscow, Russian Federation) serg@parallel.ru,

D.A. Nikitenko, Research Computing Center, Moscow State University (Moscow, Russian Federation) dan@parallel.ru,

S.I. Sobolev, Research Computing Center, Moscow State University (Moscow, Russian Federation) sergeys@parallel.ru,

K.S. Stefanov, Research Computing Center, Moscow State University (Moscow, Russian Federation) cstef@parallel.ru,

P.A. Shvets, Research Computing Center, Moscow State University (Moscow, Russian Federation) shpavel@parallel.ru

At RCC MSU we are working on the system for securing of active control and efficient autonomous operating of supercomputers. This system is been implemented at MSU Supercomputing Center. The paper describes system installation, setting and usage experience for the control of the "Chebyshev" supercomputer.

Keywords: supercomputer, graph, model, monitoring, active control, autonomous operating, Octotron.

References

1. Antonov A.S., Voevodin Vad.V., Voevodin Vl.V., Zhumatiy S.A., Nikitenko D.A., Sobolev S.I., Stefanov K.S., Shvets P.A. Razrabotka printsiptov postroeniya i realizatsiya prototipa sistemy obespecheniya operativnogo kontrolya i effektivnoy avtonomnoy raboty superkomp'yuternykh kompleksov [Securing of Reliable and Efficient Autonomous Functioning of Supercomputers: Basic Principles and System Prototype] // Vestnik UGATU [Vestnik UGATU]. 2014. Vol. 18, No. 2. P. 227–236.
2. Sobolev S.I. Superkomp'yuter v shtatnom rezhime [Supercomputer in Regular Conditions] // Otkrytye sistemy [Open Systems]. 2014. No. 8.
3. Shvets P.A., Voevodin V.V., Sobolev S.I. Ob odnom podkhode k modelirovaniyu superkomp'yuternykh kompleksov [On a One Approach to Supercomputers Simulation] //

- Nauchnyy servis v seti Internet: mnogoobrazie superkomp'yuternykh mirov: Trudy Mezhdunarodnoy superkomp'yuternoy konferentsii (22–27 sentyabrya 2014 g., g. Novorossiysk) [Internet Services & Internet: Variety of Supercomputing Worlds. International Supercomputing Conference Proceedings, Sep 22–27, 2014, Novorossiysk]. Izd-vo MGU, Moskva [MSU Publishing, Moscow]. 2014. P. 197–204.
4. HP OpenView. URL: http://www.openview.hp.com/solutions/ams/ams_bb.pdf (accessed: 06.04.2015).
 5. xCAT, An extreme cluster/cloud administration toolkit. URL: http://sourceforge.net/p/xcat/wiki/Main_Page/ (accessed: 06.04.2015).
 6. Lu K. et al. Iaso: an autonomous fault-tolerant management system for supercomputers // Frontiers of Computer Science. 2014. Vol. 8, No. 3. P. 378–390. DOI: 10.1007/s11704-014-3503-1
 7. Programmnoe obespechenie kompanii T-Platformy [T-Platforms Software]. URL: <http://www.t-platforms.ru/products/software.html> (accessed: 06.04.2015).
 8. Voevodin Vad V., Stefanov K.S. Avtomaticheskoe opredelenie i opisanie setevoy infrastruktury superkomp'yuterov [Automated Detection and Description of Supercomputer Network Structure] // Vychislitel'nye metody i programmirovaniye: Novye vychislitel'nye tekhnologii [Numerical Methods and Programming]. 2014. Vol. 15, No. 3. P. 560–568.
 9. Polnyy iskhodnyy kod Octotron [Octotron Full Source Code]. URL: https://github.com/srcc-msu/octotron_core (accessed: 06.04.2015).
 1. 10. Rabochee okruzhenie Octotron dlya sozdaniya modeli na yazyke Python [Octotron Environment for Model Creating with Python]. URL: <https://github.com/srcc-msu/octotron> (accessed: 06.04.2015).

Received March 9, 2015.