

УДК 004.416.6

СИСТЕМА МОНИТОРИНГА СЕРВИСА ПЕРЕДАЧИ ДАННЫХ (FTS) ПРОЕКТА EGEE/WLCG

В. В. Кореньков¹, А. В. Ужинский¹

Надежность, производительность и удобство — факторы, определяющие тенденции в развитии web-сервисов в настоящее время. В работе рассматривается структура, архитектурные особенности, а также общее устройство системы, разработанной для улучшения этих качеств в сервисе передачи данных (FTS) глобальной инфраструктуры EGEE/WLCG. Представлена стратегия, в соответствии с которой создавалась система. Описываются наиболее интересные отчеты и возможности системы.

Ключевые слова: сервисы, мониторинг, GRID, FTS, надежность, анализ ошибок, статистика, экспертная система.

Введение. В 2008 г. в тестовом режиме был запущен самый большой в мире ускоритель элементарных частиц — Большой адронный коллайдер (Large Hadron Collider, LHC) [1], основной задачей которого является открытие новых фундаментальных частиц и выявление их свойств. Четыре эксперимента LHC ежегодно будут генерировать порядка 15 Пбайт данных. Создание отдельной инфраструктуры для хранения и обработки подобных объемов данных потребовало бы колоссальных организационных и финансовых затрат, вследствие чего было решено использовать для решения данной задачи grid-технологии.

Грандиозный проект распределения данных с ускорителя по всему миру получил название LHC Computing GRID (LCG), в дальнейшем проект стал называться WLCG (Worldwide LHC Computing GRID) [2]. В рамках проекта решаются вопросы построения распределенной иерархической архитектуры системы региональных центров, на которых и будет производиться хранение и обработка информации. Планируется, что доступ к экспериментальным данным получают более 5000 ученых из более чем 500 исследовательских институтов и университетов, распределенных по всему миру. Объемы передаваемых данных уже на этапе тестирования поражают своей масштабностью. В среднем в январе–феврале 2009 г. в день передавалось порядка 38 терабайт данных при пропускной способности 425 мегабит в секунду. Однако когда ускоритель будет запущен на полную мощность и будет задействован механизм репликации данных (в Grid каждый файл имеет несколько своих копий (реплик) на различных элементах хранения), объемы передач значительно возрастут. Большие массивы данных будут ежедневно распределяться между различными центрами по всему миру. Для реализации этих проектов были разработано и создано соответствующее инструментальное и программное обеспечение, которое в настоящее время находится в режиме пробной эксплуатации.

Основой для построения любой grid-инфраструктуры при наличии оборудования является подпрограммное обеспечение — middleware. Разработка middleware для WLCG ведется в рамках проекта EGEE (Enabling Grids for E-sciencE) [3]. Основной задачей проекта является объединение мировых компьютерных ресурсов в единую однородную среду, которой ученые могли бы пользоваться совместно в мировом масштабе. Сообщество ученых, занимающихся физикой высоких энергий, — главный пользователь инфраструктуры EGEE, поэтому часто эту интегрированную инфраструктуру называют EGEE/WLCG.

Основными строительными блоками в middleware являются grid-сервисы. Одним из важнейших сервисов в EGEE/WLCG является сервис передачи файлов — FTS (File Transfer Service) [4]. Сервис устанавливается только на ключевых сайтах распределенной инфраструктуры. Несомненно, что надежный, производительный и удобный сервис передачи данных — это обязательная составляющая успешной работы проекта EGEE/WLCG и, как следствие, проекта LHC. Общая информация о вышеупомянутых проектах обсуждается в [5]. Очевидно, что для согласованной работы всех узлов системы и надежной передачи данных необходим постоянный контроль grid-сервисов — мониторинг. В частности, мониторинг FTS является критически важной задачей. Теоретическая информация по вопросам мониторинга распределенных систем и grid приводится в [9, 10].

¹ Объединенный институт ядерных исследований, лаборатория информационных технологий, ул. Жолио-Кюри, 6, 141980, г. Дубна; В. В. Кореньков, зам. дир. лаб., e-mail: Korenkov@cv.jinr.ru; А. В. Ужинский, инж.-программист, e-mail: zalexandr@list.ru

Первоначально набор средств мониторинга FTS ограничивался инструментами для анализа объемов данных на каналах — GridView [6], а также отчетом о проценте самых распространенных и вторых по распространенности ошибок по каналам за прошедший день и неделю. Существовали также некоторые инструменты, разработанные на различных сайтах, позволяющие получить информацию о текущем состоянии компонентов системы (агентах, настройках каналов и т.д.) через web-интерфейс без использования командной строки. Однако большой спектр вопросов о функционировании FTS оставался не охвачен: полный список ошибок, частота возникновения определенных ошибок, история развития проблемных ситуаций, причины возникновения ошибок и т.д. В результате возникла острая необходимость в создании системы, способной решать эти задачи, а при возможности также и интегрировать функционал уже имеющихся систем. Поэтому сотрудниками НИИЯФ МГУ (Научно-исследовательский институт ядерной физики, МГУ им. М. В. Ломоносова, Москва.) и ОИЯИ (Объединенный институт ядерных исследований, Дубна) была разработана и успешно зарекомендовала себя система мониторинга FTS, основанная на разборе системных отчетов (log-файлов) [7]. Система в течение полутора лет активно использовалась в ЦЕРНе (Европейском центре ядерных исследований, Женева, Швейцария) для поддержания работоспособности FTS-каналов и позволила выявить ряд программных ошибок в различных приложениях. Опираясь на данные, полученные за время работы с системой, впервые был проведен статистический анализ сбоев, возникающих на каналах связи. Анализ позволил определить наиболее часто встречаемые ошибки и причины их возникновения, основные ареалы возникновения ошибок, а также различные программные и логические ошибки в приложениях. Подробная информация об исследовании приводится в [8]. Опыт, полученный как при создании системы, так и за время работы с ней, позволил разработать стратегию создания средств мониторинга FTS, которая может быть использована и другими web-сервисами. Ниже будут представлены основные положения этой стратегии (раздел 1), реализация которой в приложении к FTS рассмотрена в разделе 2.

Основное назначение систем мониторинга — обеспечение информации для принятия управленческих решений. Информация, предоставляемая системой, не должна быть просто статистикой, которая накапливается годами и никем не используется. Она должна помогать сделать мониторируемую систему лучше, производительнее, надежнее. Кроме того, общей тенденцией последних 20–30 лет является стремление к автоматизации и уменьшению ручного труда операторов, вследствие чего следует изначально, еще при проектировании системы, предусмотреть для нее подобные варианты развития. В рамках настоящей статьи рассматриваются возможности применения технологии экспертных систем для автоматизации исходной системы. Обсуждение экспертной системы и описание ее реализованного прототипа в применении к FTS содержится в разделе 3. Рассмотрение завершается кратким заключением.

1. Стратегия создания средств мониторинга FTS. Использование единой стратегии создания средств мониторинга FTS позволяет решить сразу несколько задач: определить общие архитектурные принципы и функциональные возможности приложений, стандартизировать работу и взаимодействие разработчиков, сформулировать основные требования к системе и т.д. Ниже представлен перечень базовых положений разработанной нами стратегии.

1. *Все компоненты должны разрабатываться в соответствии с единой классификацией ошибок FTS.* Данное правило позволит избежать недопонимания между пользователями различных уровней, значительно упростить процесс разработки приложений и снизить нагрузку на базы данных (БД) при выполнении большинства запросов по обработке исходных данных.

2. *Агрегация и визуализация информации должны предоставлять возможность точного определения причин и мест возникновения проблем.* У пользователей должна существовать возможность получения как общих отчетов (например, количество удачных/неудачных передач для всего сервиса или канала), так и максимально детализированных, вплоть до количества определенных ошибок на конкретном хосте (элементе хранения, SE) для любой из организаций. Должна существовать возможность работы с различными уровнями детализации и получения отчетов в наиболее удобных для различных пользователей формах. Использование различных уровней детализации позволяет значительно сократить время анализа данных.

3. *Большинство вычислений должны производиться в базе данных.* СУБД Oracle, применяемая в FTS, является достаточно мощной платформой, чтобы можно было бы ее использовать в качестве вычислительной среды, не опираясь на какие-либо дополнительные средства. Кроме удобного механизма добавления изменений, наш подход позволяет не зависеть от дополнительных программных и аппаратных ресурсов, использовать единый формат написания кода и значительно снизить время при доработке процедур.

4. *Агрегированная для различных отчетов информация должна храниться в БД.* В идеальном ва-

рианте должны быть известны все возможные запросы пользователей, однако в реальности это далеко не так. Тем не менее, большинство нужных пользователям отчетов — это различные “срезы” одних и тех же исходных данных. Поэтому следует еще на этапе проектирования системы мониторинга определить основные объекты, информация о которых может заинтересовать пользователей, и разработать механизмы объединения исходных данных для большинства типов отчетов. Кроме того, необходимо определить временные интервалы, по истечении которых данные могут быть “безболезненно” суммированы.

2. Система мониторинга, интегрированная в схему данных FTS. Разработанная нами система мониторинга FTS была спроектирована в соответствии с описанной выше стратегией, пожеланиями пользователей, учетом недостатков уже существующих систем и различных наработок из параллельных областей. Основная часть системы находится непосредственно в базе данных и напрямую взаимодействует с таблицами FTS. В качестве одного из методов взаимодействия с системой разработан web-интерфейс, однако данные из системы могут быть использованы и другими приложениями.

При проектировании системы было решено предоставить пользователям удобный механизм управления настройками FTS непосредственно из административного модуля web-интерфейса, а также дополнительные средства для анализа взаимозависимостей между различными ошибками, возникающими на каналах связи. Кроме того, в виде отдельных модулей были разработаны механизм предупреждений при возникновении нештатных ситуаций и прототип экспертной системы (ЭС).

2.1. Хранение и обработка данных в системе. За время работы с первоначальной версией системы, обрабатывающей log-файлы [7], было выделено более 400 паттернов ошибок, возникающих на каналах связи. Большинство сообщений об ошибках содержат персонафицированную информацию — дату, название файла, атрибуты пользователя и т.д., поэтому для их хранения применяется технология паттернов — устойчивых составных частей, однозначно характеризующих данный тип сбоев. Возьмем, к примеру, ошибку типа “failed to contact on remote SRM http://ccsrn.in2p3.fr:8443/srm/ [managerv2]. GivinTup after 3 tries”: подстрока “http://ccsrn.in2p3.fr:8443/srm/ managerv2” может меняться в зависимости от оборудования, однако это не изменяет тип ошибки. Поэтому паттерн будет состоять из двух частей: часть 1 — “failed to contact on remote SRM”, часть 2 — “GivinTup after 3 tries”. Изучение подобных паттернов позволило разработать и применить в FTS новую, улучшенную классификацию ошибок, в соответствии с которой сообщение о сбое состоит из четырех частей.

1. Score — источник ошибки. Он может принимать значения: SOURCE — сайт-источник, DESTINATION — сайт назначения, TRANSFER — транспортные протоколы, проблемы связи и т.д.

2. Category — класс ошибки. Определено более 30 неизменных классов ошибок (FILE_EXIST, NO_SPACE_LEFT, TRANSFER_TIMEOUT и др.).

3. Phase — этап в жизненном цикле передачи, на котором произошла ошибка (ALLOCATION, TRANSFER_PREPARATION, TRANSFER и т.д.).

4. Message — детальное описание ошибки. В настоящее время существует список из более чем 400 различных паттернов, который постоянно изменяется и дополняется.

Подобная форма представления информации об ошибках позволяет достаточно легко решать вопросы разделения, объединения и детализации сбоев. При ее использовании определение точного места возникновения ошибки, ее класса, а при необходимости и получение детального описания не составляет особого труда.

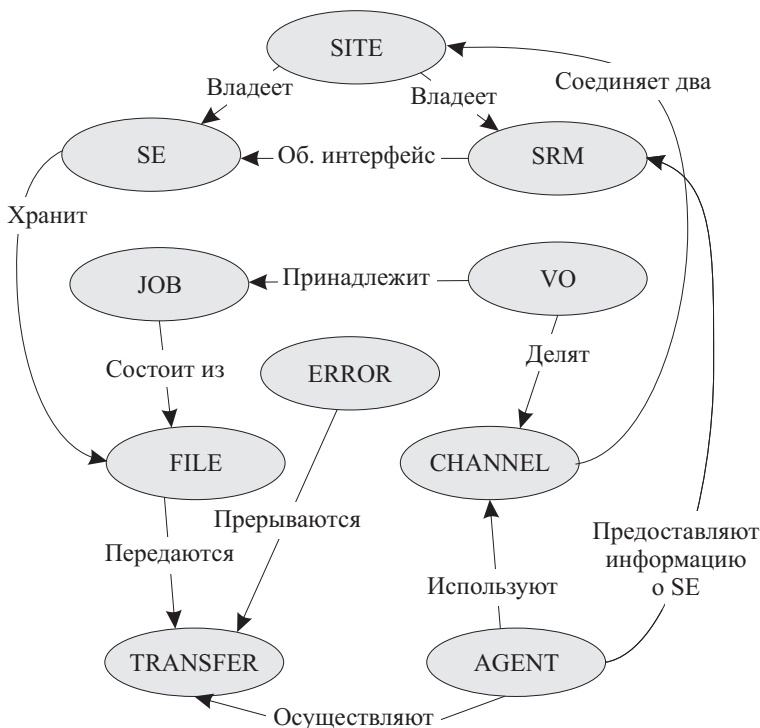


Рис. 1. Схема взаимодействия объектов FTS

На рис. 1 представлена схема взаимодействия объектов системы мониторинга. Обозначенные на ней

объекты представляют наибольший интерес для пользователей, а информация о них в явном или неявном виде содержится в схеме данных FTS.

Как видно из схемы, любая задача (JOB) принадлежит одной из виртуальных организаций (VO) и состоит из набора файлов (FILE). FTS-агенты (AGENT) отвечают за передачу файлов, используя каналы (CHANNEL). Естественно, в процессе передачи может произойти ошибка (ERROR), которая ее прерывает. Канал соединяет два сайта (SITE), на которых установлена система управления элементами хранения (SRM), и непосредственно элементы хранения (SE, hosts), на которых хранятся файлы.

Пользователей интересует как разнообразная общая информация (число удачных/неудачных передач, число активных, готовых к исполнению, завершенных задач и т.д.), так и информация о конкретных ошибках для различных объектов, а администраторов — еще и настройки FTS. Наибольший интерес для пользователей различных уровней представляют следующие объекты: каналы, виртуальные организации, сайты, хосты (SE) и ошибки. Информация о файлах и задачах как об отдельных объектах не представляет большого интереса для пользователей, их скорее интересуют отчеты типа количества выполненных/невыполненных задач для конкретной виртуальной организации или канала или суммарное количество переданных по каналу файлов (т.е. файлы и задачи в качестве параметров основных объектов), нежели изучение каких-либо общих тенденций их самостоятельного развития во всей системе. Об FTS-агентах администраторов интересуют лишь их параметры совместно с настройками FTS для получения представления о структуре сервиса. Это же касается и информации о SRM, установленных на сайтах.

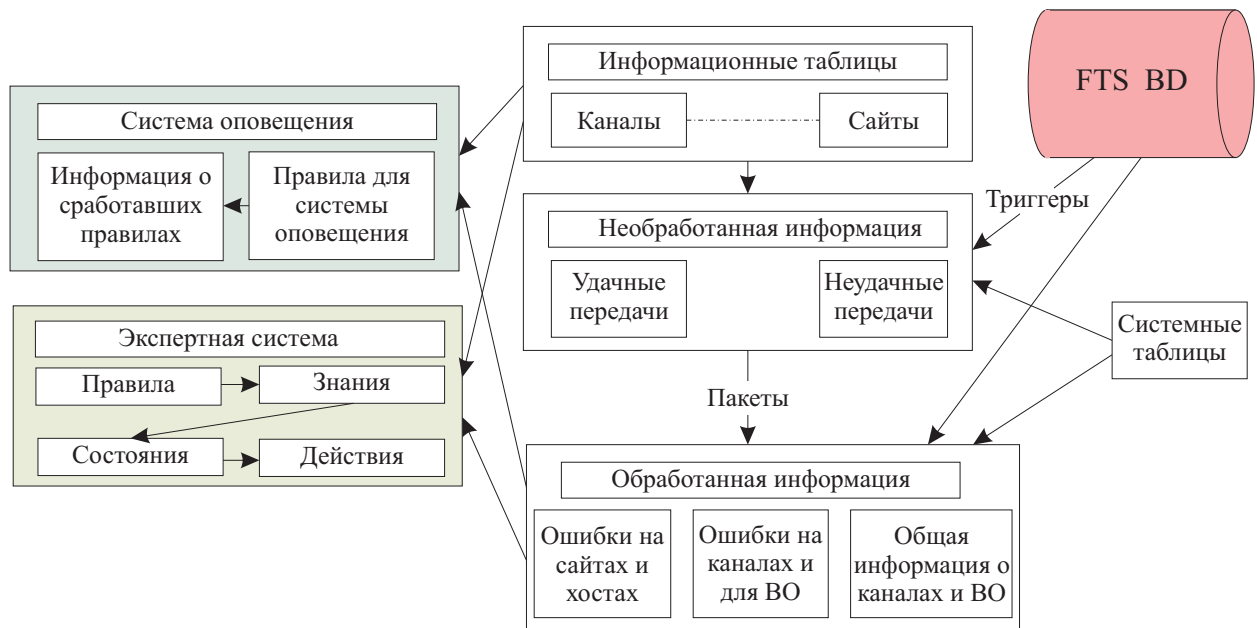


Рис. 2. Модель данных системы мониторинга FTS

В соответствии с изложенной стратегией разработки средств мониторинга FTS, получение, хранение и обработка данных в системе производится непосредственно в БД (Oracle). Общая модель данных системы представлена на рис. 2.

В системе имеются информационные таблицы, отвечающие за хранение общей информации о сайтах, каналах, виртуальных организациях и т.д. Таблицы содержат данные об удачных и неудачных передачах, информация о которых поступает из схемы данных FTS посредством триггеров. Данные из этих таблиц обрабатываются процедурами и функциями, агрегируются либо разделяются по определенным признакам и заносятся в таблицы для обработанной информации. Существуют таблицы, содержащие информацию об ошибках для сайтов и хостов, а также для каналов и виртуальных организаций. Кроме того, имеются таблицы с общей информацией для объектов: количество удачных/неудачных передач, число ошибок на сайте-источнике и сайте-назначения либо произошедших во время передачи, время выполнения задачи и т.д. В соответствии с модульной архитектурой системы, блоки “Экспертная система” и “Система оповещения” имеют собственные отдельные таблицы и устанавливаются либо не устанавливаются в зависимости от пожеланий пользователя. В своей работе представленные модули используют обработанную информацию и данные из информационных таблиц.

2.2. Отчеты и интерфейсы системы. В процессе работы с версией системы, обрабатывающей log-файлы, были определены четыре основные группы пользователей — менеджеры виртуальных организаций, высший менеджмент, FTS-операторы и администраторы сайтов. Каждая из представленных групп заинтересована в различных данных, собранных за различные временные интервалы и представленных в различном виде. Например, менеджеров виртуальных организаций интересует как информация об общем состоянии дел в масштабе всего сервиса, так и ситуация на конкретных каналах и сайтах. В основном, менеджеров интересует информация за определенный период, например неделю или месяц, для различных отчетов, собраний и совещаний. Администраторов сайтов интересует информация о настройках и текущем состоянии каналов “из” и “в” их сайт, а также данные об ошибках на стороне их сайта и его хостов. Более востребованы последние данные и информация за прошедшие сутки. Администраторов могут интересовать как FTS-категории для определения узких мест, так и конкретное описание ошибок для их устранения. Высший менеджмент интересуют самые общие отчеты, представленные в презентабельном виде за достаточно большие промежутки времени. Операторов FTS могут интересовать довольно специфические отчеты: например, случаи появления конкретных ошибок на каналах, сайтах или у определенной виртуальной организации, степень связанности различных ошибок и т.д. Естественно, приветствуются любые отчеты, позволяющие в кратчайшие сроки с наименьшими затратами найти и определить причину ошибок, возникающих на каналах связи. Одним словом, существовал довольно большой спектр отчетов, которые необходимо было стандартизировать и реализовать (см. в разделе 1 положение 4 стратегии).

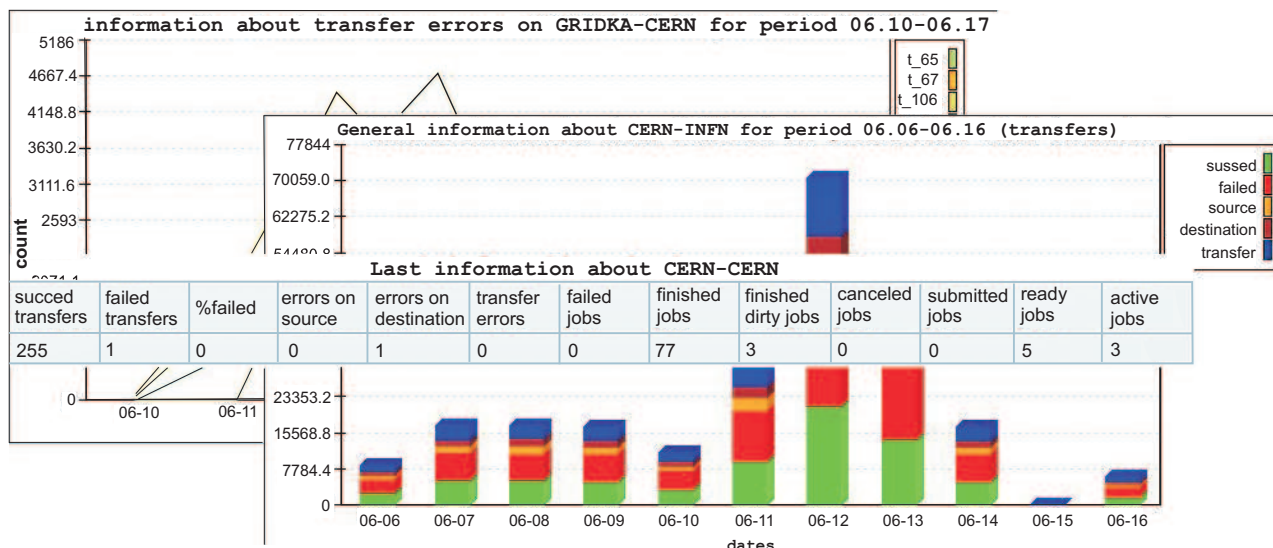


Рис. 3. Примеры отчетов в системе мониторинга FTS

В настоящее время при работе с системой у пользователя есть возможность получить актуальную информацию за последние 24 часа или любой определенный период в днях. Подобные временные интервалы позволяют перекрыть большинство пользовательских запросов. Данные в системе представлены в виде таблиц, графиков и диаграмм. Примеры отчетов приведены на рис. 3. Благодаря продуманной схеме организации данных, пользователю может быть предоставлено большое число объектов для анализа и фильтров для работы с ними. Мы постарались стандартизировать общий вид интерфейса пользователя. Он состоит из списка объектов, выбора временных интервалов, выбора вида представления результатов, а также различных фильтров.

Рис. 4. Основной интерфейс модуля “Channels”

На рис. 4 и 5 в качестве примера представлены интерфейсы модулей “Channels” и “Sites”. Как можно заметить, модуль “Channels” позволяет получить информацию об общей ситуации на каналах (General sit-

Рис. 5. Основной интерфейс модуля “Sites”

uation), всех (All errors) либо только транспортных (Transfer errors) ошибках, а также об FTS-категориях ошибок (Categories). Существует возможность создания отчетов для каждой виртуальной организации в отдельности. Модуль “Sites” позволяет получить информацию об ошибках и FTS-категориях. Кроме фильтра разделения по виртуальным организациям, существует возможность получить отчет об ошибках, произошедших на сайте, когда он выступал в роли сайта-источника или сайта-назначения.

Web-интерфейс системы состоит из модулей, предоставляющих информацию для различных групп пользователей. При работе с системой у пользователя есть возможность начать работу непосредственно из интересующего его модуля либо с главной страницы, на которой представлены общие отчеты, позволяющие определить состояние сервиса и возможные источники проблем.

Главная страница web-интерфейса системы содержит следующие отчеты: отчет о передачах (количество удачных/неудачных, количество ошибок на сайтах-источниках, на сайтах назначения и в процессе передачи) и задачах (количество завершенных успешно/неудачно/“грязно”, отмененных, направленных в очередь, готовых к исполнению и активных) для всего сервиса; отчет о передачах, успешно и неудачно завершенных на каналах (позволяет определить самые активные или проблемные каналы за последние сутки); общий отчет о ситуации для виртуальных организаций и каналов, с возможностью дальнейшей детализации, для анализа производительности и определения неполадок.

Ниже дано краткое описание модулей системы мониторинга. Нам представляется, что подобный подход при разделении функций может быть полезен и в других системах мониторинга.

Модуль “FTS Settings” содержит статические настройки FTS-каналов, распределение квот канала между различными виртуальными организациями, а также настройки FTS-агентов.

Модуль “Alarms” содержит список сработавших триггеров, которые создал системный администратор, и является одним из инструментов оповещения пользователей в механизме оповещений системы.

Модуль “Channels” предоставляет информацию об общей ситуации и ошибках на каналах передачи данных.

Модуль “Expert System” содержит историю развития состояния различных объектов и рекомендации по исправлению сложившейся ситуации. Естественно, для получения данного отчета необходимо, чтобы эксперты заполнили базу знаний.

category/mistakes list

TOP-7 channels with biggest amount of errors for period 2008.05.01-2008.06.20 for dteam

place	volume	channels
1	176081	GRIDKA-CERN
2	106121	CERN-INF2P3
3	100792	CERN-CERN
4	97174	PIC-CERN
5	82251	CERN-PIC
6	67766	CERN-BNL
7	64674	IN2P3-CERN

Рис. 6. Пример интерфейса и отчета в модуле “VO”

Модуль “VO” предоставляет информацию об общей ситуации и ошибках для виртуальных организаций. Существует возможность создания отчетов для каждого канала или сайта в отдельности. Дополнительно существует возможность создания рейтингов каналов или сайтов с наибольшим количеством

ошибок за последний день либо за определенный временной интервал. Пример интерфейса и отчета представлен на рис. 6.

Модули “Sites” и “Hosts” предоставляют информацию об FTS-категориях и ошибках на сайтах и хостах.

Модуль “Errors” создан специально для отслеживания тенденций в развитии ошибок как во всей структуре в целом, так и на конкретных объектах. Пример интерфейса модуля представлен на рис. 7.

Monitoring tools					
Select error [1] 421 Timeout (▲) [2] 426 Transfer a... [3] SOAP-ENV:Cli [4] service timeou...	Select channel all GRIDKA-CERN CERN-INFN INFN-CERN	Select time options <input checked="" type="radio"/> last info <input type="radio"/> last 24h <input type="radio"/> period From - 06 2008 Till 20 06 2008	Separate by <input type="checkbox"/> VO <input type="checkbox"/> Channel	result form tables	submit
Statistic tools					
Enter size of Top: 10	Select time options <input checked="" type="radio"/> last day <input type="radio"/> period	From - 06 2008 Till 20 06 2008	Sorted by <input checked="" type="radio"/> frequency <input type="radio"/> total amount	Separate by <input type="checkbox"/> VO <input type="checkbox"/> Channel	submit
Correlation					
Select error1 [1] 421 Timeout (▲) [2] 426 Transfer a... [3] SOAP-ENV:Cli [4] service timeou...	Select error2 [1] 421 Timeout (▲) [2] 426 Transfer a... [3] SOAP-ENV:Cli [4] service timeou...	Select time period From - 06 2008 Till 20 06 2008	Type of correlation <input checked="" type="radio"/> Spirmen <input type="radio"/> Pirsen <input type="radio"/> Scatter plot	submit	

Рис. 7. Пример интерфейса модуля “Errors”

Кроме предоставления информации о развитии конкретных ошибок на каналах или у виртуальных организаций, в модуле существует возможность создания рейтингов ошибок с наибольшей частотой появления либо с наибольшим общим количеством для VO или каналов. Работа с ранней версией системы показала, что при определении причины возникновения новых ошибок (неисправности оборудования, неточности настроек, программные ошибки и т.д.) очень полезна информация о степени их связанности с ошибками, для которых причины уже определены. В модуле “Errors” существует возможность определить для пары ошибок коэффициент линейной корреляции Пирсона или коэффициент ранговой корреляции Спирмена.

Коэффициент линейной корреляции Пирсона определяется как

$$r = \frac{\sum_{i=1}^n (x_i - \bar{X})(y_i - \bar{Y})}{(n-1)S_x S_y},$$

где x_i и y_i — значения двух переменных, \bar{X} и \bar{Y} — их средние значения, S_x и S_y — их стандартные отклонения, а n — количество пар значений.

Коэффициент ранговой корреляции Спирмена имеет вид: $r = 1 - \frac{6 \sum d^2}{n(n^2 - 1)}$,

где $\sum d^2$ — сумма квадратов разностей рангов, а n — число парных наблюдений.

Существует также возможность построить для пары ошибок диаграмму рассеивания (рис. 8).

Практически все отчеты снабжены перекрестными ссылками, предоставляющими удобный способ детализации результатов. Например, из общего отчета по каналу можно сразу же перейти к анализу ошибок на сайте-источнике, сайте назначения или к транспортным ошибкам, а из отчета об ошибках на сайте — к анализу ошибок на его хостах и т.д.

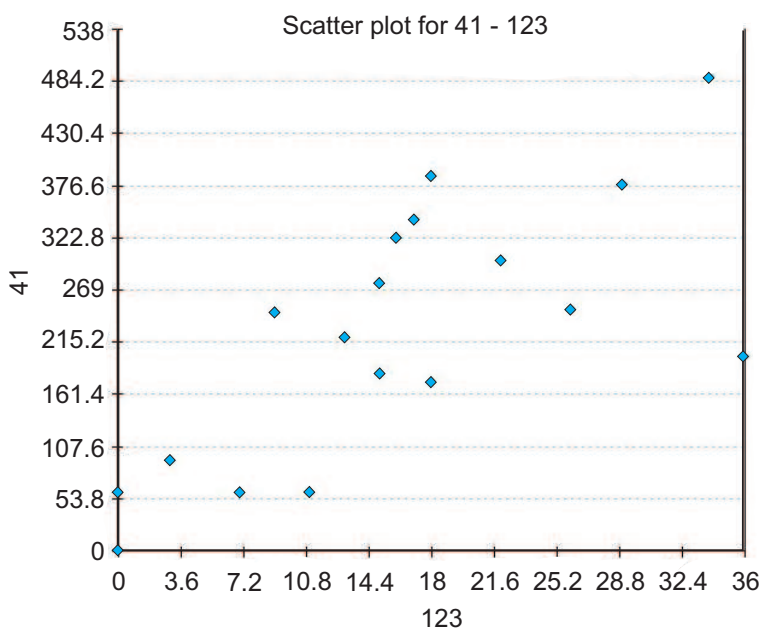


Рис. 8. Диаграмма рассеивания для 41 и 123 ошибок за период 10.06.2008–20.06.2008

В системе существует функция кэширования изображений. Например, если за последний час кто-то уже запрашивал информацию об ошибках на канале CERN-INFN в виде диаграмм, то при повторном запросе будет выдана сохраненная картинка, а запросы к базе данных производиться не будут.

2.3. Панель администратора в системе. Настройка системы мониторинга является одним из важнейших этапов работы с ней, вследствие чего проектирование удобной и функциональной панели администратора должно являться обязательным пунктом при разработке системы. Основные функции панели администратора — это управление триггерами оповещения (добавление, удаление, изменение) и объектами системы (добавление, изменение, удаление, поиск). Объем данных довольно велик, так что возможность автоматического обнаружения объектов совсем не лишняя, а в случае работы с новыми ошибками просто необходимая. При использовании модуля экспертной системы (ЭС) панель администратора является инструментом заполнения базы знаний ЭС. Интерфейс панели администратора представлен на рис. 9.

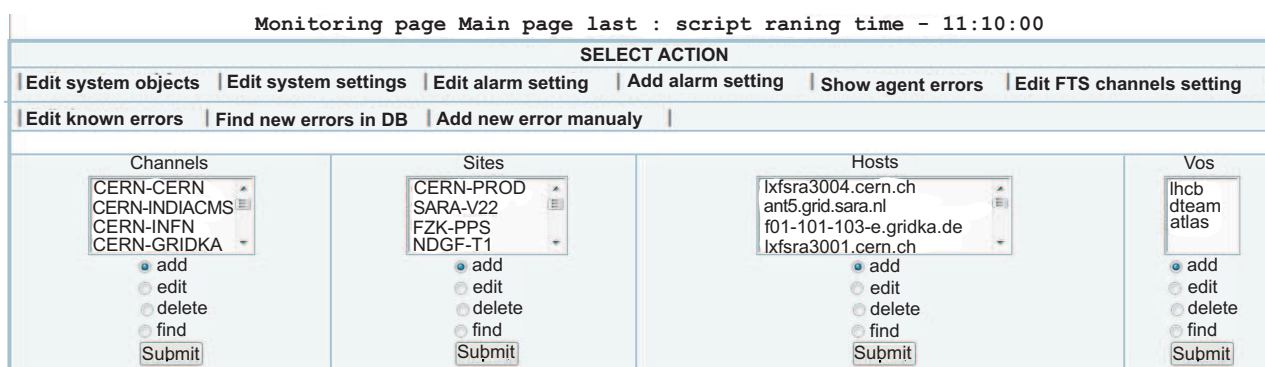


Рис. 9. Интерфейс панели администратора

Администратор может изменить настройки системы мониторинга: схему оформления интерфейса (в системе содержатся несколько шаблонов дизайна), настройки механизма оповещения, отображение информации о хостах, кэширование изображений и т.д. В рабочую версию системы было решено включить возможность изменения некоторых настроек FTS непосредственно из web-интерфейса. Подобное решение выглядит вполне логичным, поскольку позволяет администратору при обнаружении проблем сразу же внести необходимые изменения в настройки FTS. Например, обнаружив несоответствие распределения квот между организациями на канале, администратору не надо запускать терминал, переходить на хост, где установлены агенты виртуальных организаций, и изменять настройки через интерфейс командной строки: он может просто зайти в панель администрирования, изменить и сохранить необходимые настройки.

2.4. Механизм оповещений в системе. Существенным нововведением в системе мониторинга является специально разработанный механизм оповещения. Любой администратор FTS-сервиса может создать свои собственные триггеры оповещения, т.е. наборы правил, при срабатывании которых будут выполнены определенные действия (отправлены оповещения посредством web-интерфейса, электронных сообщений, sms и т.п.). Триггер можно создать для каналов, сайтов, хостов и виртуальных организаций. Существует три типа триггеров.

1. При превышении числа ошибок определенного уровня.
2. При изменении уровня ошибок более чем на заданную величину. Предположим, что у нас создан подобный триггер с величиной 200, тогда если в 13.00 число ошибок равнялось 100, а в 14.00 — 301, то он сработает, так как уровень ошибок изменился на 201. Теоретически это модификация триггеров первого типа, но более пригодная для ситуаций, когда у нас имеются ошибки, постоянно возникающие на канале.
3. При превышении процента неудачных передач определенного уровня.

Если пользователь работает с триггерами типа 1 и 2, то он может указать id определенной ошибки, чтобы отслеживать только ее развитие. При работе с каналами, сайтами и хостами пользователь может указать также виртуальную организацию с теми же целями. Интерфейс работы с триггерами представлен на рис. 10.

В соответствии с описанной в первой части работы стратегией, обработка триггеров производится процедурой в БД, а информация о том, сработал триггер или нет, может быть получена из специальной таблицы. Таким образом, если кому-либо потребуется данная информация для создания электронных сообщений, sms, вычисления коэффициентов надежности и т.д., то нет необходимости производить сложные

SELECT ACTION														
Edit system objects		Edit system settings		Edit alarm setting		Add alarm setting		Show agent errors		Edit FTS channels setting				
Edit known errors		Find new errors in DB		Add new error manually										
Object-	CERN-GRIDKA	Object type-	channel	A_TYPE-	2	VO-	dteam	M_ID-	0	LEVEL-	0	<input checked="" type="radio"/> edit	<input type="radio"/> delete	Submit
Object-	lxsra3004.cern.ch	Object type-	host	A_TYPE-	1	VO-	All	M_ID-	0	LEVEL-	3	<input checked="" type="radio"/> edit	<input type="radio"/> delete	Submit
Object-	f01-015-109-e.gridka	Object type-	host	A_TYPE-	1	VO-	All	M_ID-	0	LEVEL-	20	<input checked="" type="radio"/> edit	<input type="radio"/> delete	Submit
Object-	dteam	Object type-	VO	A_TYPE-	3	VO-	All	M_ID-	0	LEVEL-	50	<input checked="" type="radio"/> edit	<input type="radio"/> delete	Submit

If M-id = 0 it means all mistake

A_TYPE: 1 - if amount > level; 2 - if amount(t)-amount(t-1) > level (increase from the last time);
3 - if % of failurt > level (allowed only for channel or VO)

Рис. 10. Интерфейс изменения триггеров оповещения в системе мониторинга

вычисления, следует всего лишь обратиться к нужной таблице в БД.

Особый класс ошибок — “GENERAL_FALURES” — представляет для администраторов FTS наибольший интерес, поскольку означает неисправности в работе самого сервиса, поэтому при любых настройках механизма оповещений при возникновении подобных ошибок администраторы получают сообщения от механизма оповещений, а на всех страницах web-интерфейса возникает соответствующее предупреждение.

3. Применение технологии экспертных систем для автоматизации исходной системы.

Одной из основных задач систем мониторинга является снабжение пользователя данными для формирования представлений о том, в каком состоянии находятся объекты системы, а также система в целом. Однако простому пользователю зачастую довольно сложно получить подобное представление и, тем более, определить пути исправления сложившейся ситуации, поскольку для решения данной задачи требуется значительный опыт. В свою очередь, эксперт способен на основе некоторых характеристик объектов и их значений выдвинуть предположение либо точно определить состояние, в котором находится объект, а также определить меры по изменению этого состояния. Поскольку работа эксперта — достаточно ресурсоемкий процесс, представляется целесообразным отделить знания от эксперта для автоматизации процесса определения состояний объектов и необходимых действий для разрешения проблемных ситуаций. Для решения подобных задач разрабатываются различные экспертные системы. ЭС — компьютерные программы, способные заменить эксперта в решении проблемных ситуаций. Теоретические и практические вопросы организации ЭС описаны в [11, 12].

В области мониторинга подобные системы способны значительно упростить определение проблемных ситуаций, минимизировать труд операторов и облегчить понимание проблем. В результате использования ЭС пользователям не приходится анализировать исходные данные, поскольку они получают на доступном для них языке информацию о состоянии объектов и системы в целом, объяснения относительно того, почему данные объекты находятся в этом состоянии, а также список рекомендуемых действий, которые необходимо выполнить для устранения неблагоприятных ситуаций. Естественно, при достаточной “мощности” подобной системы роль пользователя/администратора при разрешении возникающих проблем может быть полностью нивелирована, так как уровень автоматизации позволит гарантировать необходимую надежность работы системы. Таким образом, ЭС, опирающаяся на данные системы мониторинга, способна стать ключевым элементом управления исходной системы.

Object type-	Field-	Case-	Value-	<input checked="" type="radio"/> edit	<input type="radio"/> delete	submit
channel/vo	N_TRANSFER	>	100			
Object type-	Field-	Case-	Value-	<input checked="" type="radio"/> edit	<input type="radio"/> delete	submit
channel/vo	globus_ftp_client_size: Connection timed out	>	50			

Рис. 11. Интерфейс задания правил

Создание ЭС является достаточно сложной задачей, вследствие чего в рамках нашей работы был разработан лишь модуль-прототип экспертной системы для FTS, призванный продемонстрировать потенциальные возможности использования данной технологии. Система состоит из четырех блоков: правила — > знания — > состояния — > действия. На каждом из этапов используется опыт экспертов, что, в конечном итоге, позволяет любому пользователю получить представление о состоянии отдельных объектов и системы в целом, а также произвести некоторые действия для изменения этих состояний.

Базовые правила (condition) задаются для определенного класса объектов (каналы, виртуальные организации, сайты, хосты) в виде отношений (больше, меньше, равно и т.д.) к определенному значению.

На рис. 11 представлены два правила, которые могут быть использованы для любого канала или виртуальной организации. Первое считается верным при количестве ошибок передачи данных более 100, второе — при количестве определенных системой GridFTP ошибок более 50.

Знания (knowledge) задаются для определенного объекта или класса объектов и определяют их состояние в зависимости от значения составляющих правил.

Parent knowledge- none	Object- any	Object type- channel	Condition- channel/vo: N_TRANSFER>100	State- channel stuck	<input checked="" type="radio"/> edit <input type="radio"/> delete	submit
→Parent knowledge- channel: any-channel stuck	Object- any	Object type- channel	Condition- channel/vo: globus_ftp_client_size: Connection timed out>50	State- channel stuck	<input checked="" type="radio"/> edit <input type="radio"/> delete	submit

Рис. 12. Интерфейс задания знаний

На рис. 12 представлен пример задания знания, устанавливающего состояние “channel stuck” у любого канала при условии выполнения двух описанных выше правил.

Состояния (state) — различные состояния, которые характеризуют объекты системы и которые способны значительно облегчить понимание как конкретных проблем, так и общего состояния системы.

Действия (action) — меры, которые необходимо предпринять при условии нахождения конкретного объекта или любого представителя класса объектов в некотором состоянии.

Object- any	Object type- channel	State- channel stuck	Action- restart transfer-agent	<input checked="" type="radio"/> edit <input type="radio"/> delete	submit
----------------	-------------------------	-------------------------	-----------------------------------	--	--------

Рис. 13. Интерфейс задания действий

На рис. 13 представлен пример задания действия (перезапуск агентов передач), которое необходимо выполнить при нахождении любого из каналов в состоянии “channel stuck”.

Таким образом, работа экспертов по заполнению базы знаний сводится к трем этапам:

- задание правил для всевозможных объектов;
- определение состояний, которые характеризуются теми или иными правилами;
- определение действий при различных состояниях объектов.

Подобная организация базы знаний предоставляет достаточно гибкий механизм, позволяющий перекрывать множество различных объектов и их характеристик, а также использовать доступные для понимания описания состояний объектов и действий, необходимых для их изменения.

Результатом работы системы является список объектов, их состояний и рекомендуемых действий, а также таблица истории изменений состояния объектов.

Time	Object type	Name	State	Action
2008-10-23 14:38:26	Site	SARA-V22	down time	open ticket
2008-10-23 14:38:26	Channel	CERN-INDIACMS	channel stuck	restart transfer-agent
2008-10-23 14:38:26	Site	CERN-PROD	lost files	open ticket for Castor

Рис. 14. Пример отчета в модуле “Expert System”

На рис. 14 представлен отчет, вторая строка которого является результатом работы описанных выше правил и знаний.

Таким образом, даже человек, не имеющий значительного опыта работы с системой, сможет без труда не только определить текущие неисправности, но и выполнить определенные шаги для их устранения.

Представленный модуль является лишь прототипом, призванным раскрыть потенциальные возможности подобного подхода. Естественно, для разработанного модуля существует множество путей совершенствования: расширение возможностей задания правил и знаний, усовершенствование методов объяснения, разработка четкой и ясной классификации состояний, определение наиболее эффективных действий и т.д. Кроме того, представляется логичным путь полной либо частичной автоматизации работы операторов. Однако если в случае действий, требующих работы лишь с настройками, хранящимися в БД, или

при оповещении сторонних участников это не представляет особых трудностей, то работа с агентами, файлами, правами пользователей и т.д. является непростой задачей.

В рамках же рассматриваемой работы модуль призван продемонстрировать, что ЭС в нашем случае способна описать состояние системы на доступном для пользователя языке и при необходимости объяснить, вследствие каких причин объект находится в текущем состоянии, какие действия необходимо предпринять для исправления сложившейся ситуации и т.д. Обычно пользователи хотят посвящать анализу исходных данных как можно меньше времени. Все, что они хотят, — чтобы их система работала надежно и с высокой производительностью. Мы полагаем, что созданная ЭС поможет пользователям в данном вопросе. В настоящее время ведется активное тестирование представленной ЭС.

Заключение. Представлена стратегия (основные требования) создания средств мониторинга, которая может быть использована при развитии широкого класса web-сервисов.

В соответствии с этой стратегией разработана новая версия системы мониторинга FTS. Система предоставляет “гибкий” и надежный способ получения и анализа информации, а также расширенные возможности по управлению как собственным состоянием, так и настройками сервиса передачи данных FTS. Дана информация о структуре и архитектурных особенностях, а также об общем устройстве системы. Описаны основные интерфейсы, отчеты и возможности системы мониторинга.

Благодаря гибким настройкам и конструктору правил (триггеров) оповещения, предложенный для системы мониторинга механизм оповещения позволяет значительно упростить работу операторов FTS и администраторов сайтов.

Для решения проблемы определения причин возникновения новых ошибок в системе предоставлена возможность проведения корреляционного анализа ошибок.

В качестве первого шага на пути автоматизации работы операторов предложен прототип модуля экспертной системы, позволяющий интерпретировать состояние системы в понятных пользователям терминах и при необходимости объяснять, вследствие каких причин тот или иной объект находится в определенном состоянии и какие действия необходимо предпринять для исправления сложившейся ситуации.

В настоящее время система мониторинга активно используется в EGEE/WLCG как для поддержания работоспособности, так и для оптимизации работы FTS.

СПИСОК ЛИТЕРАТУРЫ

1. LHC home page (www.lhc.web.cern.ch/lhc).
2. LCG home page (www.lcg.web.cern.ch/LCG).
3. EGEE home page (www.public.eu-eg ee.org).
4. FTS twiki page (<https://twiki.cern.ch/twiki/bin/view/EGEE/FTS>).
5. Кореньков В.В., Ужинский А.В. Архитектура сервиса передачи данных в grid // Открытые системы. 2008. № 2. 52–56.
6. GridView home page (<http://gridview.cern.ch/>).
7. Кореньков В.В., Ужинский А.В. Система мониторинга сервиса передачи данных (FTS) проекта EGEE/WLCG // Сообщения ОИЯИ. Дубна, 2008. P11-2008-80.
8. Кореньков В.В., Ужинский А.В. Статистический анализ сбоев, возникающих при пересылке данных в глобальной грид-инфраструктуре EGEE/WLCG // Сообщения ОИЯИ. Дубна, 2008. P11-2008-82.
9. Joyce J., Lomow G., Slind K., Unger B. Monitoring distributed systems // ACM Transactions on Computer Systems (TOCS). 1987. 5, N 2. 121–150.
10. Zanolis S., Sakellariou R. A taxonomy of grid monitoring systems // Future Generation Computer Systems. 2005. 21, N 1. 163–188.
11. Jackson P. Introduction to expert systems. Reading: Addison Wesley, 1998.
12. Giarratano J.C., Riley G.D. Expert systems: principles and programming. Boston: PWS Publ. Co., 2004.

Поступила в редакцию
05.02.2009