Р11-2008-80

А. В. Ужинский, В. В. Кореньков

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

Ужинский А. В., Кореньков В. В. Р11-2008-80
Система мониторинга сервиса передачи данных (FTS)
проекта EGEE/WLCG

В настоящее время очень важной является проблема обеспечения высокого уровня надежности и качества grid-сервисов. В данной статье описываются структура и функциональные свойства системы мониторинга сервиса FTS (File Transfer Service — сервис передачи файлов — один из сервисов gLite).

Система разработана в рамках проекта EGEE/WLCG и нацелена на обнаружение ошибок и нестандартных ситуаций при переносе данных по FTS-каналам. Система состоит из трех компонент, которые реализованы различным образом: извлечение информации — Perl и Shel, хранение данных — Mysql и представление данных — PHP и XHTML. Даны краткие сведения об отчетных формах. Работа выполнена сотрудниками ОИЯИ и НИИЯФ МГУ совместно со специалистами ЦЕРН.

Работа выполнена в Лаборатории информационных технологий ОИЯИ.

Сообщение Объединенного института ядерных исследований. Дубна, 2008

Uzhinskiy A. V., Korenkov V. V. P11-2008-80
FTS Monitoring System of EGEE/WLCG Project

Nowadays it is very important to provide a high level of Grid Services performance and reliability. The work describes FTS monitoring system structure and functionality (File Transfer Service, one of the gLite services). The system was developed in the framework of the EGEE/WLCG project. It is aimed to detect errors and nonstandard situations at data transfer through FTS channels. The system has three components implemented in various techniques: data mining — Perl and Shel, data storing — Mysql and data representation — PHP and XHTML. User interface and summary reports are shortly considered. The work has been carried out by JINR and SINP MSU researchers in collaboration with CERN.

The investigation has been performed at the Laboratory of Information Technologies, JINR.

Communication of the Joint Institute for Nuclear Research. Dubna, 2008

ВВЕДЕНИЕ

Большой адронный коллайдер (LHC — Large Hadron Collider, http://lhc.web.cern.ch/lhc/), строительство которого в настоящее время завершается в ЦЕРН, является самым масштабным научным проектом на планете, и после запуска он будет производить около 15 петабайт данных ежегодно. Когда ускоритель заработает в нормальном режиме потребуется обеспечить доступ к экспериментальным данным 5000 ученых из более чем 500 исследовательских институтов и университетов всего мира. Для этого создается и будет поддерживаться распределенная инфраструктура для хранения и анализа данных, что является крайне сложной задачей, в решении которой особую роль занимают grid-технологии. В проекте, получившем название LHC Computing GRID (LCG) (в дальнейшем проект стал называться WLCG, Worldwide LHC Computing GRID, http://lcg.web.cern.ch/LCG/), решаются вопросы построения распределенной иерархической архитектуры системы региональных центров, в которых и будет храниться и обрабатываться информация с ускорителя. Основная задача другого крупного grid-проекта — EGEE (Enabling Grids for E-sciencE, http://public.eu-egee.org/), тесно связанного с WLCG, — организовать мировые компьютерные ресурсы в единую однородную среду для решения научных задач. Главным пользователем инфраструктуры EGEE является сообщество ученых, занимающихся физикой высоких энергий (ФВЭ). Поэтому часто эту интегрированную инфраструктуру называют EGEE/WLCG. В рамках проекта EGEE/WLCG разрабатывается программное обеспечение промежуточного слоя (middleware) для построения grid-систем под названием gLite (http://glite.web.cern.ch/glite/). Основным понятием в grid являются grid-сервисы — это сервисы высокого уровня, необходимые пользователям и виртуальным организациям и реализованные в сервисно-ориентированной архитектуре (Service Oriented Architecture), основанной, главным образом, на web-сервисах с соблюдением рекомендаций по интероперабельности (Web Services Interoperability (WS-I)). Существуют grid-сервисы для следующих областей: безопасность, информация и мониторинг, работа с данными, управление заданиями, помощь пользователям и т. д.

Наиболее приоритетным направлением в развитии grid-сервисов на данный момент является увеличение их производительности и надежности, поскольку пользователи хотят иметь стабильный доступ к ресурсам 24 часа 7 дней в неделю, вне зависимости от праздников и выходных. Системы мониторинга могут помочь в данном вопросе, предоставляя как общую информацию о производительности и функционировании сервисов, так и информацию об ошибках, нестандартных ситуациях и «узких» местах.

В соответствии с определенной в рамках проекта WLCG схемой, необработанные (так называемые «сырые») данные будут храниться на центрах (сайтах)* Tier-0 (CERN) и Tier-1, а их обработка будет производиться на центрах Tier-2. Все эти центры территориально разнесены, поэтому вопросы надежного хранения и перемещения данных между различными сайтами имеют огромное значение. В соответствии с требованиями проекта EGEE/WLCG каждый сайт должен иметь систему хранения данных Castor (http://www.castor.org/), dCache (http://www.dcache.org/) или DPM (http://www.gridpp.ac.uk/wiki/Disk_Pool_Manager) и систему управления хранилищами SRM (Storage Resource Manager, http://www.gridpp.ac.uk/wiki/SRM). На физическом уровне данные в gLite передаются посредством либо GridFTP (http://dev.globus.org/wiki/GridFTP) — протокола, разработанного в рамках проекта Globus (http://www.globus.org/), либо его модификаций. Сервис передачи данных FTS (http://egee-jra1-dm.web.cern.ch/egee-jra1-dm/FTS/default.htm, https://twiki.cern.ch/twiki/ bin/view/EGEE/FTS) [1] должен быть установлен только на сайтах Tier-0 и Tier-1. Предоставляя надежный способ передачи данных типа точка–точка, FTS организует и контролирует работу всех остальных элементов. Основные обязанности FTS — обеспечение надежных и удобных механизмов передачи файлов типа точка–точка, контроля и мониторинга передач, распределения ресурсов сайта между экспериментами, а также управления запросами пользователей различных виртуальных организаций. Ежедневно десятки тысяч файлов передаются между различными сайтами, а объемы передач составляют десятки терабайт в день. При подобной интенсивности потоков данных мониторинг является одной из самых приоритетных задач. Общие вопросы мониторинга и grid-мониторинга в частности рассматриваются в [2, 3].

До недавнего времени набор средств мониторинга FTS включал в себя следующие возможности: анализ объемов данных на каналах с использованием системы GridView (http://gridview.cern.ch/GRIDVIEW/index.php); получение отчета о проценте ошибок первого и второго рода по каналам за прошедший день и неделю — FTS Report (https://fts102.cern.ch/ftsstats/prod-

*Tier с соответствующим номером — это вычислительный центр определенного уровня в иерархии центров, принятой для построения grid-инфрастуктуры WLCG.

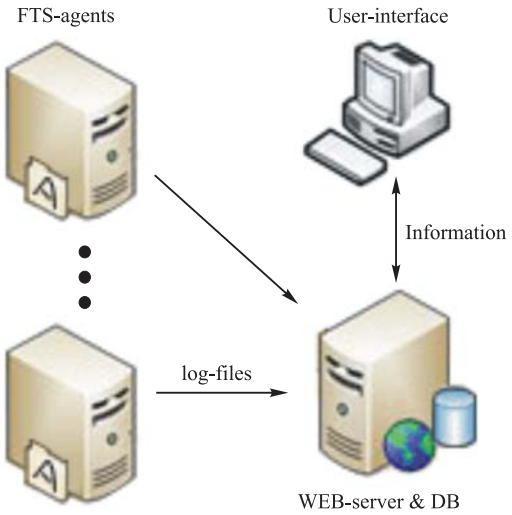


Рис. 1. Общая схема системы мониторинга

fts-ws/index.html), а также некоторые дополнительные возможности, разработанные на Tier-1-сайтах, для получения информации о текущем состоянии системы — агентах, настройках каналов и т. д. — через web-интерфейс.

Данная система мониторинга давала недостаточно полную информацию о проблемах, возникающих на каналах, а также не обеспечивала оперативное предоставление информации. Сложившаяся ситуация инициировала создание системы мониторинга «Spider» [4], основное предназначение которой — отображение информации об ошибках, возникающих на каналах передачи данных. Работы в данном направлении проводились сотрудниками ОИЯИ и МГУ в рамках проекта LCG.

Сервис FTS построен таким образом, что информация о передачах данных хранится на серверах, где установлены «агенты передач» (transfer agent), в виде og-файлов. Каталоги с данными файлами очищаются ежесуточно (в полночь). Для получения информации о проблемах на каналах требуется обрабатывать log-файлы, сохранять полученные данные, а также предоставлять доступный интерфейс для работы с ними. Таким образом, для системы мониторинга была выбрана трехзвенная архитектура: извлечение данных — набор скриптов, ответственных за перемещение и обработку log-файлов (реализация — Perl, shell), хранение данных — база данных системы, реализованная в MySQL, и представление данных — доступ к информации из БД через web-интерфейс (реализация — PHP, XHTML). Общая схема системы представлена на рис. 1.

ИЗВЛЕЧЕНИЕ ДАННЫХ (PERL + SHELL)

Первым шагом работы скриптов уровня извлечения данных является перемещение log-файлов с серверов, на которых установлены агенты передач, на узел, где установлена система мониторинга. Очевидно, что обработка log-файлов может производиться и непосредственно «на месте», но для уменьшения нагрузки на серверы было решено производить все работы на сторонних машинах. За временной интервал для снятия показаний системы был выбран один час, так как это позволяет своевременно реагировать на возникновение внештатных ситуаций и в то же время не создает чрезмерной нагрузки на ресурсы.

Следующим этапом является определение количества разнообразных ошибок, возникших на канале с 12 часов ночи. Следует заметить, что FTS взаимодействует с различными элементами хранения и передачи данных (Castor, dCache, DPM, SRM, GridFTP), и поэтому спектр возникающих ошибок, приводящих к неудачному завершению передачи данных, довольно велик, что привело к необходимости предусмотреть возможность «безболезненного» добавления новых ошибок в систему. Поскольку в диагностической информации об ошибке обычно содержится уникальная персонифицированная информация — время, дата, название файла или атрибуты пользователя (DESTINATION error during PREPARATION phase: [FILE_EXISTS] at Tue Jul 24 13:18:28 CEST 2007 state Failed : file exists), то неизбежно использование механизма «шаблонов», т. е. определенных устойчивых частей, однозначно характеризующих конкретную ошибку. В результате для хранения различных ошибок была использована отдельная таблица в базе данных, в которой содержатся полный пример ошибки, ее три основные составляющие части и прочая информация.

Упрощенный алгоритм работы скрипта выглядит примерно так: скрипт сопоставляет сообщения об ошибках в тексте очередного файла с шаблонами из базы данных, и если ошибка не будет опознана, то она определяется как ранее неизвестная, и ее текст направляется для рассмотрения администратору. В случае же совпадения с шаблоном количество ошибок подобного типа на канале увеличивается на единицу. В конце работы программы мы имеем количество ошибок каждого типа на определенном канале, и остается только записать эту информацию в БД.

ХРАНЕНИЕ ДАННЫХ (MYSQL)

В базе данных хранится информация об известных ошибках — примеры, шаблоны (паттерны), типы и т. д. Отдельная таблица отведена для хранения информации об отслеживаемых каналах. Существует «историческая»

таблица, в которой записываются случаи появления ошибок на каналах с описанием вызвавших их причин. Информация о количестве разнообразных ошибок, возникающих на каналах, хранится по временным интервалам в некоторой основной таблице.

ПРЕДСТАВЛЕНИЕ ДАННЫХ (PHP + XHTML)

Web-интерфейс системы (доступный только пользователям ЦЕРН) представлен на рис. 2 и может быть найден по адресам http://pcitgd26.cern.ch/channel.php и http://lxb2012.cern.ch/.

Основной блок программы «Monitoring tools» имеет следующие настройки: канал, временной период (может быть предоставлена самая последняя информация, информация за последние 24 часа либо за временной период более суток) и форма вывода результата. Информация может быть представлена в виде таблиц, графиков, диаграмм, а также агрегирована по типу ошибок. Примеры отчетов в системе приведены на рис. 3.

Кроме приложения, вызвавшего ошибку: Castor, SRM, GridFTP и т.д., часто удается установить, где именно произошла ошибка: на сайте-источнике, сайте назначения или в процессе передачи. В результате анализа различных ошибок было выделено 16 различных классов:

«fts» — ошибки, возникающие из-за неисправности самого сервиса FTS; «globus» — ошибки с gridftp; «user» — ошибки, возникающие по вине пользователей; «tcp» — ошибки подключения к сети; «t_dcache», «s_dcache», «d-cache» — ошибки dCache («t_» и «s_» обозначают, на чьей стороне произошла ошибка, на сайте-источнике или сайте назначения: t-target, s-source),

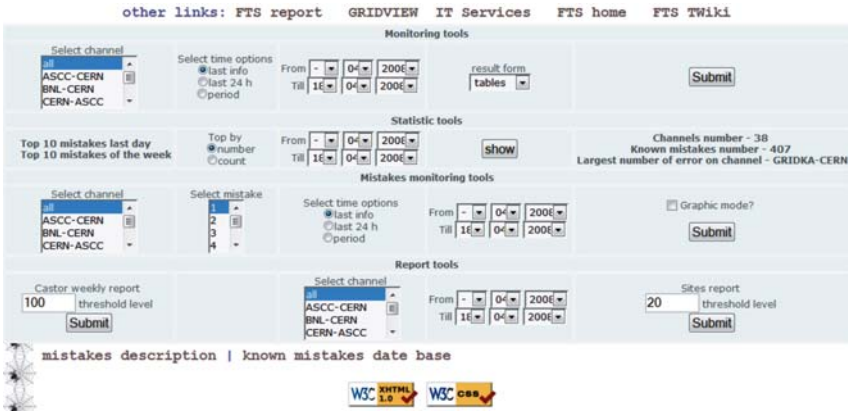


Рис. 2. Общий вид интерфейса системы

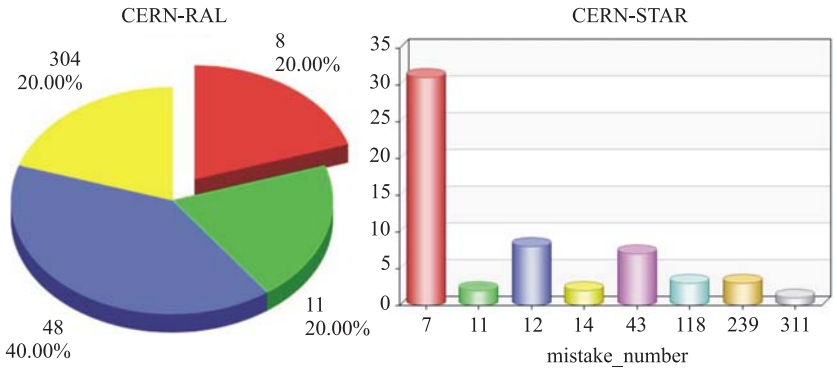


Рис. 3. Примеры отчетов в системе

CERN-IHEP

date	time	all	lost	fts	globus	user	tcp	dcach	castor	t_dcach	s_dcach	t_castor	s_castor	dpm	t_dpm	s_dpm	t_srm	s_srm	srm
2008-04-18	13:13:26	519	0	0	96	0	0	0	0	0	0	0	0	0	0	0	10	393	0

Рис. 4. Аналитический отчет о канале CERN–IHEP

«t_castor», «s_castor», «castor» — ошибки Castor, «t_dpm», «s_dpm», «dpm» — ошибки DPM, и «t_srm», «s_srm», «srm» — кроме ошибок непосредственно самого SRM в данный класс входят и ошибки, у которых не удалось выяснить приложение, повлекшее их возникновение. Использование подобной классификации было необходимо для упрощения анализа результатов на начальных этапах работы, так как количество шаблонов ошибок достаточно велико (более 350), и намного удобнее сначала определить класс проблем, возникающих на канале, а затем переходить к более детальным исследованиям. Тип представления результата, при котором ошибки на канале агрегированы по классам, в системе называется аналитический отчет. Пример аналитического отчета представлен на рис. 4.

Для решения проблем на канале зачастую необходимо знать конкретный тип ошибки, поэтому в системе предусмотрена возможность детализировать аналитический отчет по любому его пункту, т. е. перейти от общего класса ошибок к конкретным ошибкам. Пример подобного отчета представлен на рис. 5.

Channel	2	3	14	16	60	98	129	190	225	226	244	245	288	295	305	311	322	331	333
CERN-IHEP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	122	0	0	0

Рис. 5. Предыдущий отчет, развернутый по классу ошибок «t_castor»

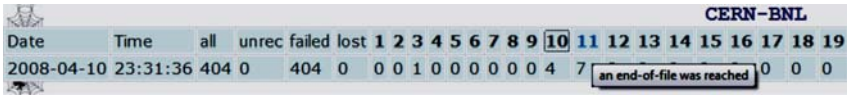


Рис. 6. Пример всплывающей подсказки с текстом ошибки

Для характеристики ошибок в системе используются их уникальные идентификационные номера id. Список всех ошибок с их полным текстом и уникальными номерами доступен по адресу http://pcitgd26.cern.ch/mistakes.php, а информация о случаях проявления ошибок на каналах и вызвавших их причинах может быть найдена по адресу http://pcitgd26.cern.ch/mkdb.php. Для получения полного текста ошибки пользователю достаточно просто навести курсор на ее номер, если тип предоставления результатов — «таблицы» или «аналитика», либо нажать на него, что повлечет перемещение на страницу с детальным описанием ошибки (см. рис. 6).

Довольно часто при отладке приложений интересен не весь спектр возникающих ошибок, а только некоторые из них — для подобных задач и был разработан дополнительный модуль «Mistake monitoring tools», позволяющий получать информацию по конкретным ошибкам, отсеивая все остальные. Набор настроек у него примерно такой же, как и у описанного выше модуля, с некоторыми изменениями: добавлена возможность выбирать конкретные ошибки, а представление результатов возможно только в виде таблиц и графиков. Пример отчета о нескольких ошибках представлен на рис. 7.

Модуль «Reports tool» создан для упрощения создания отчетности, он позволяет создать недельный отчет по ошибкам, возникающим из-за проблем с

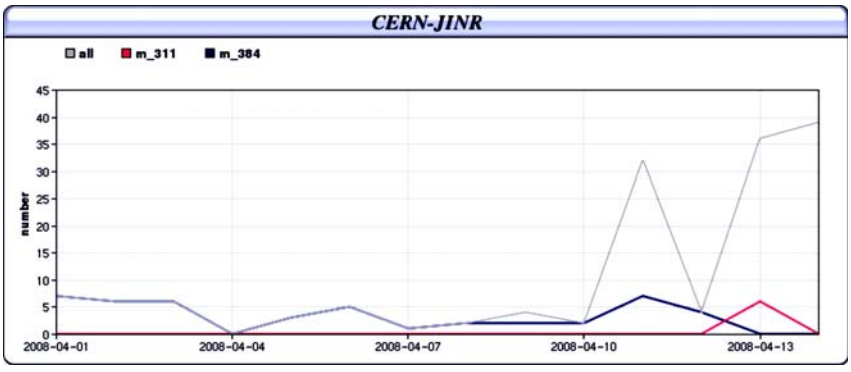
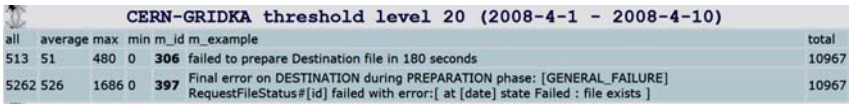


Рис. 7. Отчет о 311 и 384 ошибках на канале CERN–JINR за период с 01.04.08 по 14.04.08



CERN-GRIDKA threshold level 20 (2008-4-1 - 2008-4-10)

all	average	max	min	m_id	m_example	total
513	51	480	0	**306**	failed to prepare Destination file in 180 seconds	10967
5262	526	1686	0	**397**	Final error on DESTINATION during PREPARATION phase: [GENERAL_FAILURE] RequestFileStatus#[id] failed with error:[at [date] state Failed : file exists]	10967

Рис. 8. Отчет о наиболее распространенных ошибках на канале CERN-GRIDKA с 1.04.2008 по 10.04.2008

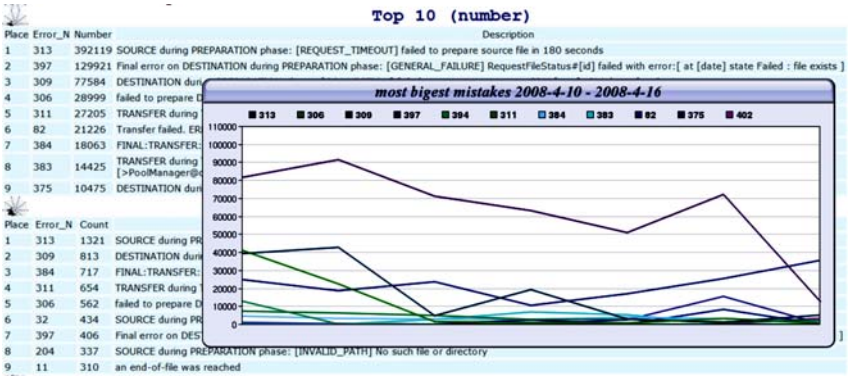


Рис. 9. Рейтинг ошибок за наделю и наиболее часто встречающиеся ошибки с 10.04.08 по 16.04.08

CERN Castor, а также получить отчет о наиболее распространенных ошибках на остальных сайтах с учетом порога чувствительности за определенный временной интервал. Используя этот модуль, администратор может получить список ошибок на его сайте, отбросив редко встречающиеся ошибки и заняться выяснением причин их возникновения. Пример отчета представлен на рис. 8.

Последний модуль «Statistic tools» используется для отображения общей информации о системе — количестве каналов и ошибок в системе, канала с наибольшим текущим количеством ошибок и т. д. Также этот модуль позволяет получить рейтинги ошибок по частоте либо по суммарному количеству на каналах за последний день, неделю или иной определенный временной интервал. Пример отчетов данного модуля представлен на рис. 9.

ЗАКЛЮЧЕНИЕ

Предложена общая концепция системы мониторинга сервиса передачи данных (FTS), предназначенная для обнаружения ошибок на каналах передачи данных. Рассмотрена трехуровневая архитектура системы. Представлено краткое описание интерфейсов и отчетных форм. Данная система в на-

стоящий момент активно используется в ЦЕРН для поддержания работоспособности FTS-каналов. Также следует отметить, что данные, получаемые в результате работы разработанной системы мониторинга, неоднократно способствовали выявлению программных ошибок в различных приложениях.

ЛИТЕРАТУРА

1. *Кореньков В., Ужинский А.* Архитектура сервиса передачи данных в grid // Открытые системы. 2008. № 2.

2. *Joyce J., Lomow G., Slind K., Unger B.* Monitoring Distributed Systems // ACM Transactions on Computer Systems (TOCS). 1987. No. 5(2). P. 121–150,

3. *Zanikolas S., Sakellariou R.* A Taxonomy of Grid Monitoring Systems // Future Generation Computer Systems. 2005. No. 21(1). P. 163–188.

4. *Uzhinskiy A.* FTS monitoring. WLCG Service Reliability Workshop, November 2007. [http://indico.cern.ch/getFile.py/access?contribId=21&sessionId=1&resId= 2&materialId=slides&confId=20080].

Получено 28 мая 2008 г.

Редактор *М. И. Зарубина*

Подписано в печать 21.10.2008.
Формат 60 × 90/16. Бумага офсетная. Печать офсетная.
Усл. печ. л. 0,75. Уч.-изд. л. 0,94. Тираж 310 экз. Заказ № 56389.

Издательский отдел Объединенного института ядерных исследований
141980, г. Дубна, Московская обл., ул. Жолио-Кюри, 6.
E-mail: publish@jinr.ru
www.jinr.ru/publish/