

УДК 004.415.2, 004.51, 004.65

РАЗРАБОТКА ВЕБ-СЕРВИСА ДЛЯ МОДЕЛИРОВАНИЯ СИСТЕМ ХРАНЕНИЯ И ОБРАБОТКИ ДАННЫХ ФИЗИЧЕСКИХ ЭКСПЕРИМЕНТОВ

**Нечаевский Андрей Васильевич¹, Пряхина Дарья Игоревна²,
Ужинский Александр Владимирович³**

¹*Инженер-программист;*

Объединенный институт ядерных исследований,

Лаборатория информационных технологий;

Россия, 141980, Московская обл., г. Дубна, ул. Жолио-Кюри, д. 6;

e-mail: nechav@jinr.ru.

³*Аспирант;*

ГБОУ ВО МО «Университет «Дубна»,

Институт системного анализа и управления;

Россия, 141980, Московская обл., г. Дубна, ул. Университетская, д. 19;

e-mail: pry-darya@yandex.ru.

³*Ведущий программист;*

Объединенный институт ядерных исследований,

Лаборатория информационных технологий;

Россия, 141980, Московская обл., г. Дубна, ул. Жолио-Кюри, д. 6;

e-mail: zalexandr@list.ru.

В статье представлен веб-сервис для моделирования систем хранения и обработки данных физических экспериментов. Описана схема работы приложения и приведен пример его использования.

Ключевые слова: веб-сервис, моделирование, грид.

WEB-SERVICE DEVELOPMENT FOR THE PHYSICAL EXPERIMENTS DATA STORAGE AND PROCESSING SIMULATION

Nechaevskiy Andrey¹, Pryahina Darya², Uzhinskiy Alexander³

¹*Software engineer;*

Joint institute for nuclear researches,

Laboratory of Information Technologies;

Russia, 141980, Moscow reg., Dubna, 6 Joliot-Curie st.;

e-mail: nechav@jinr.ru.

²*Postgraduate student;*

Dubna State University,

Institute of the system analysis and management;

Russia, 141980, Moscow reg., Dubna, 19 University st.;

e-mail: pry-darya@yandex.ru.

³*Lead programmer;*

Joint institute for nuclear researches,

Laboratory of Information Technologies;

Russia, 141980, Moscow reg., Dubna, 6 Joliot-Curie st.;

e-mail: zalexandr@list.ru.

Web-service for the physical experiments data storage and processing simulation are presented. The described scheme of application work and an example of its use is given.

Keywords: web-service, simulation, grid.

Введение

Современные научные эксперименты в различных областях науки (физики высоких энергий, астрофизики, биологии, науках о Земле) требуют интенсивной обработки данных. Для этого существуют вычислительные инфраструктуры различного масштаба. Наибольший интерес для исследования представляют грид, облачные, а также грид-облачные системы, обрабатывающие сверхбольшие объемы данных.

В настоящее время в Объединенном институте ядерных исследований (ОИЯИ, г. Дубна) планируется создание ускорительного комплекса *NICA*. Предполагается, что количество данных, поступающих с детекторов будет на уровне 20 ПБ в год [1]. Прежде, чем приступить к созданию такой установки необходимо решить, какова будет архитектура системы хранения и обработки данных, сколько ресурсных центров будет использовано и прочее, для этого также необходимо моделирование. Для вновь разрабатываемых комплексов обработки информации с помощью моделирования необходимо определить основные параметры, структуру и проверить предлагаемые технические решения.

В Лаборатории информационных технологий (ЛИТ) ОИЯИ разрабатываются инструменты имитационного моделирования систем хранения и обработки данных крупных научных экспериментов, а именно программа *SyMSim* (*Synthesis of Monitoring and Simulation* – Синтез Мониторинга и Моделирования), которая реализует идею синтеза процессов мониторинга и моделирования [2, 3]. Для хранения данных мониторинга грид систем, различных параметров моделирования, результатов работы программы и управления вычислительными экспериментами спроектирована и разработана специальная база данных (БД), а также создано программное окружение и удобный интерфейс взаимодействия с ней.

1. Проектирование и разработка базы данных

Для обеспечения совместимости с системами мониторинга, схема данных *SyMSim* была реализована на основе БД системы управления потоком задач *PanDA* эксперимента *ATLAS* на Большом Адронном Коллайдере (БАК) [4]. Такое решение обусловлено тем, что при определении входных данных для модели нужно будет проанализировать результаты мониторинга потоков заданий существующей установки. Такая совместимость дает возможность использовать результаты мониторинга потока без изменения его параметров.

В качестве системы управления базами данных (СУБД) используется *PostgreSQL*. БД включает в себя таблицы с информацией о пользователях и экспериментах; с описанием параметров для запуска модели; с описанием объектов моделируемой системы и связей между ними; и, главное, таблицу с описанием потока заданий для модели и таблицу с результатами моделирования.

2. Разработка веб-интерфейса для взаимодействия с программой моделирования

Упрощенно схема разработанной системы представлена на рис. 1. Для взаимодействия с программой моделирования был создан веб-сервис.

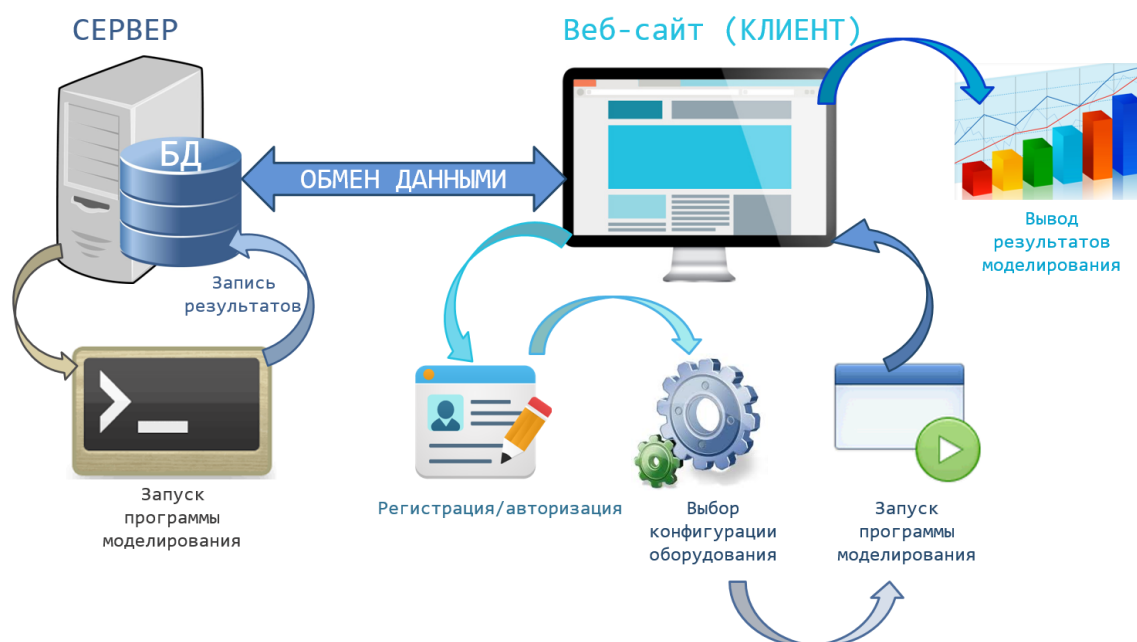


Рис. 1. Схема работы приложения

Пользователю необходимо пройти процедуру регистрации на веб-сайте, указав логин и пароль для доступа к программе, адрес электронной почты, а также выбрать название эксперимента моделирования, в котором он планирует участвовать. В дальнейшем, пройдя процедуру аутентификации, пользователь получает возможность осуществлять запуски программы, моделировать работу различных систем с использованием существующей конфигурации оборудования или добавлением новой.

Средство описания вычислительной структуры реализовано посредством веб-сервиса. БД содержит описание грид-структуры, каждого ее узла, связей между узлами, информацию о запущенных заданиях, временах исполнения, результаты мониторинга работы различных подсистем грид, а также результаты моделирования. Описанию присваивается идентификатор, который указывается в параметрах запуска модели. Модель считывает информацию из БД и строит описание вычислительной структуры. Характеристики потока задач, которые подлежат обработке, задаются через веб-интерфейс [5].

На основе заданных характеристик запускается набор утилит, который позволяет статистически проанализировать результаты мониторинга и сформировать поток заданий, аналогичный, или отличающийся от проанализированного на управляемое воздействие пользователя, в виде упорядоченной по времени последовательности записей в таблице БД.

Запуск программы моделирования осуществляется на сервере с помощью командной строки. Результатом ее работы служит последовательность записей в соответствующей таблице БД, отражающая все события, происходящие в системе. Все события описываются в едином формате. Запись привязывается к внутреннему времени.

Параметры и результаты моделирования сохраняются в БД после каждого запуска программы, чтобы предоставить возможность пользователю сравнить результаты работы систем с использованием различных конфигураций оборудования.

Пользователь может отобразить результаты моделирования в виде графиков и проводить дальнейший анализ.

Разработка процесса взаимодействия системы моделирования с БД осуществлялась на языке программирования *Java* в среде *NetBeans*. Был создан *Java*-проект, включающий класс, который реализует подключение к БД, создание курсоров, отправку запросов типа *SELECT* и *INSERT*, вывод результатов запросов. Взаимодействие приложения с СУБД *PostgreSQL* реализовано с помощью платформенно-независимого промышленного стандарта *JDBC*.

3. Пример использования приложения

Рассмотрим пример использования разработанной системы для моделирования важных процессов хранения экспериментальных данных проектируемого в ОИЯИ центра обработки данных ускорительного комплекса *NICA*.

Пройдя процедуру авторизации на веб-сайте, пользователь попадает на страницу, где графически представлена структура моделируемой грид системы для выбранного при регистрации эксперимента (например, см. рис. 2) и результаты предыдущего запуска модели. Для каждой грид системы уже существует начальная конфигурация оборудования.

Моделируемая инфраструктура

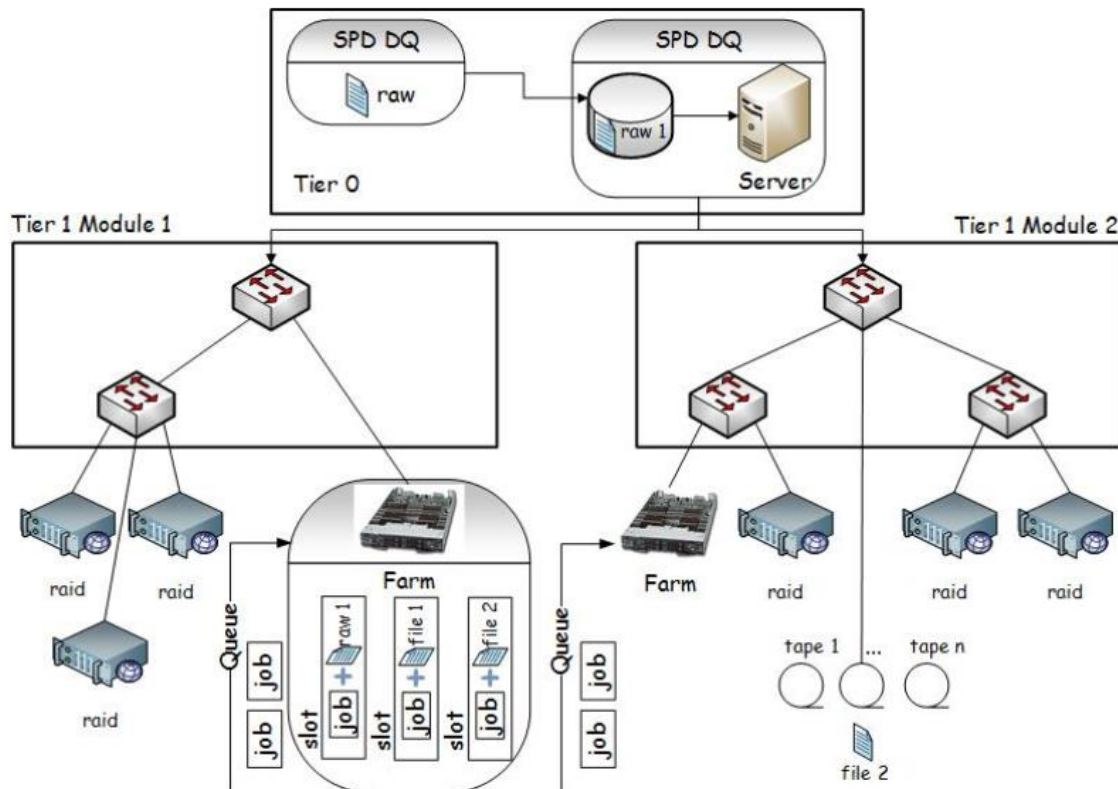


Рис. 2. Структура моделируемой грид системы [5]

Прежде, чем запустить программу моделирования, следует задать параметры входного потока заданий. Для этого необходимо нажать на ссылку «Задать другие параметры входного потока и перезапустить модель». Откроется форма для заполнения (см. рис. 3). В данном примере задается число задач различных типов, генерируются потоки задач для сохранения «сырых данных» с эксперимента (*DQ*), для моделирования (*MC*), экспресс анализа (*EA*), обработки (*PR*). То есть поток задач состоит из четырех подпотоков. Время прихода задачи в каждом из потоков распределено по нормальному закону. Величина потребляемых ресурсов (время процессора, объем ввода-вывода) также является случайной. Параметры распределений зависят от решаемых физических задач, алгоритмов обработки и т.д. Значения этих параметров можно получить из результатов мониторинга структур, обрабатывающих данные аналогичной предметной области. Значения потока данных и заданий для данного примера получены из результатов мониторинга системы обработки данных эксперимента *ATLAS* и прогноза значений для проекта *NICA* [1].

Задать другие параметры входного потока и перезапустить модель

Число задач DQ в час от:	680	Параметры: время цпу - 10, память - 2, входной файл - да
Число задач DQ в час до:	720	
Число задач MC в час от:	45	Параметры: время цпу - 3000, память - 4, входной файл - нет
Число задач MC в час до:	55	
Число задач EA в час от:	220	Параметры: время цпу - 300, память - 4, входной файл - да
Число задач EA в час до:	260	
Число задач PR в час от:	150	Параметры: время цпу - 12000, память - 4, входной файл - да
Число задач PR в час до:	250	

Рис. 3. Форма для задания параметров потока заданий

Затем необходимо выбрать объекты, информация о которых будет записываться в БД (см. рис. 4). В моделируемой системе объектами являются файлы (*FILE*), задачи (*JOB*), буфер (*RESO*), диски (*HDISK*), ленточные катушки (*REEL*), архивы (*ARCH*), ленточный робот (*ROBOT*), коммуникационные узлы (*CHAN*), ленточные диски (*DRIVE*).

FILE: JOB: RESO: HDISK: REEL: ARCH: ROBOT: CHAN: DRIVE:

Перезапустить модель

Рис. 4. Список объектов моделируемой системы

Теперь можно запустить программу моделирования. По нажатию на кнопку «Перезапустить модель» будет сгенерирован идентификатор эксперимента и поток задач для него. Также будут сформированы различные параметры, необходимые для запуска программы моделирования, в том числе значение, отвечающее за типы объектов, статистика о работе которых необходима. Все данные сохранятся в БД и запустится работа модели.

Модель запускается на сервере. Для этого был написан скрипт, который получает необходимую информацию из БД, а затем с помощью командной строки запускает модель (см. рис. 5).

```
#run the program with -exp simulation_id, -obj object_parametr (the rest is default)
java -jar dist/Alea.jar -art 1.0 -comp 1.0 -dest 1 -dt 10 -exp $simulation_id -obj $object_parametr -tz 60
```

Рис. 5. Команда для запуска модели

При запуске указываются следующие параметры для модели:

- идентификатор эксперимента (флаг *-exp*) – значение, полученное из БД (в текущем примере значение равно 101);
- идентификатор объектов системы, информация о работе которых будет записываться в БД (флаг *-obj*) — значение, полученное из БД;
- компрессор времени (флаг *-comp*) – вещественное число, больше 0;
- уровень подавления потока данных от номинала (флаг *-art*) – вещественное число, от 0 до 1;
- идентификатор места сохранения отчета моделирования: база данных или текстовый файл (флаг *-dest*).

Во время работы модели на веб-сервис отображает текущее состояние модели, а также количество записей в результирующей таблице БД для текущего эксперимента (см. рис. 6).

Текущее состояние модели: Идет расчет, подождите. Число записей в логе: 2294

Остановить модель

Рис. 6. Описание текущего состояния модели

По окончании работы модели обновится текущее состояние модели (см. рис. 7). Также можно самостоятельно остановить модель, нажав на соответствующую кнопку (см. рис. 6).

Текущее состояние модели: Расчет закончен

Рис. 7. Описание текущего состояния модели по окончании ее работы

После завершения моделирования можно посмотреть результаты. Для этого следует выбрать номер проводимого эксперимента из выпадающего списка (в рассматриваемом примере – 101). На странице появятся графики, описывающие различные характеристики моделируемой системы (см. рис. 8). Пользователь может экспортировать графики и проводить дальнейший анализ.

Результаты расчета модели

Выберите эксперимент: 101

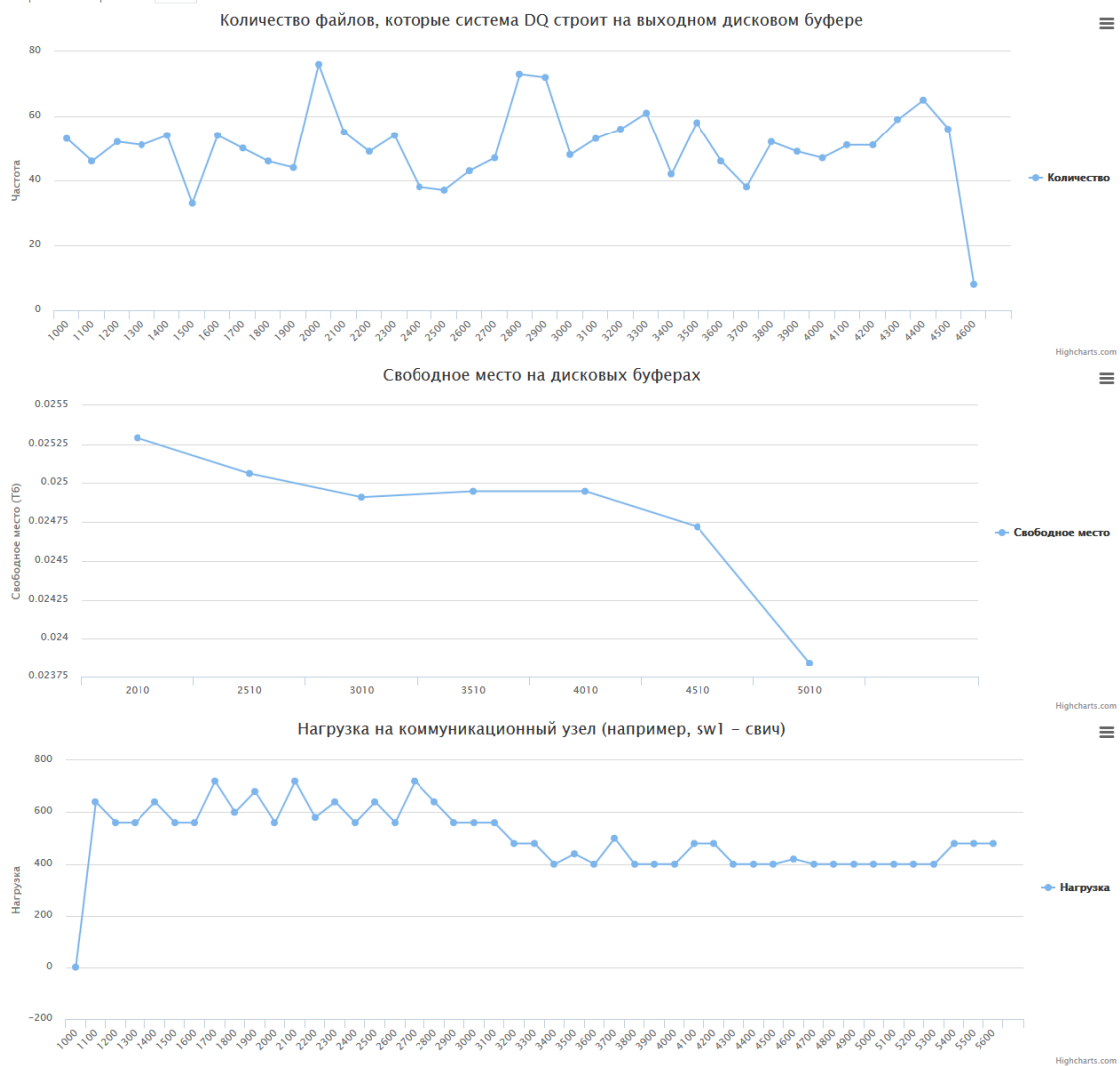


Рис. 8. Результаты моделирования

Заключение

Таким образом была создана система, реализующая идею синтеза процессов мониторинга и моделирования систем хранения и обработки больших массивов данных. Разработанная система была апробирована на важной для ОИЯИ задаче прогнозного моделирования распределенного хранения и передачи данных эксперимента *NICA*.

Пользователь с помощью удобного и понятного веб-сервиса может осуществлять запуски программы, моделировать работу различных грид-систем с использованием существующей конфигурации оборудования или добавлением новой. А потребность в организации хранилища решена путем создания такой БД, которая содержит не только входные данные для моделирования и результаты работы программы, но и предусматривает совместимость с реальными системами мониторинга.

Поэтому этап проектирования и разработки БД, а также создание программного окружения и удобного интерфейса, является очень важным в реализации программы *SyMSim*.

Благодарности

Авторы выражают свою признательность профессору В.В. Коренькову, профессору Г.А. Ососкову, В.В. Трофимову за ценные консультации и помощь в работе.

Список литературы

1. Baskakov A., Bazylev S., Filippov I., Kuklin S., Minaev Yu., Shchipunov A., Shutov A., Sidorenko V., Slepnev I., Slepnev V., Tarasov N., Terletskiy A. MPD Data Acquisition System. Technical Design Report. Discussion Paper // Laboratory of High Energy Physics, Joint Institute for Nuclear Research, Dubna, RUSSIA, 2015.
2. Кореньков В.В., Нечаевский А.В., Ососков Г.А., Пряхина Д.И., Трофимов В.В., Ужинский А.В. Моделирование грид-облачных сервисов проекта *NICA*, как средство повышения эффективности их разработки // Компьютерные исследования и моделирование. – 2014. – Т. 6. – № 5.
3. Кореньков В.В., Нечаевский А.В., Ососков Г.А., Пряхина Д.И., Трофимов В.В., Ужинский А.В. Синтез процессов моделирования и мониторинга для развития систем хранения и обработки больших массивов данных в физических экспериментах // Компьютерные исследования и моделирование. – 2015. – Т. 7. – № 3.
4. Maeno T., et all for the ATLAS Collaboration. Evolution of ATLAS PanDA System // CHEP2012 contribution id 275, 2012.
5. Korenkov V., Nechaevskiy A., Ososkov G., Pryahina D., Trofimov V., Uzhinskiy A., Balashov N. Web-Service Development of the Grid-Cloud Simulation Tools // Procedia Computer Science. – 2015.