ФИЗИКА ЭЛЕМЕНТАРНЫХ ЧАСТИЦ И АТОМНОГО ЯДРА
2020. Т. 51. ВЫП. 6. С. 1175–1303

МЕТОДЫ ОБРАБОТКИ СВЕРХБОЛЬШИХ ОБЪЕМОВ ДАННЫХ В РАСПРЕДЕЛЕННОЙ ГЕТЕРОГЕННОЙ КОМПЬЮТЕРНОЙ СРЕДЕ ДЛЯ ПРИЛОЖЕНИЙ В ОБЛАСТИ ФИЗИКИ ВЫСОКИХ ЭНЕРГИЙ И ЯДЕРНОЙ ФИЗИКИ

А. А. Климентов *

Брукхейвенская национальная лаборатория, Аптон, США

Исследования в области физики высоких энергий и ядерной физики невозможны без использования значительных вычислительных мощностей и программного обеспечения для обработки, моделирования и анализа данных. Создание и запуск таких установок, как LHC и NICA, потребовали новых подходов и алгоритмов при создании систем обработки и управления данными. Одновременно происходят интеграция гетерогенных вычислительных ресурсов в единую киберинфраструктуру, разработка концепции «озеро научных данных». В данной работе рассмотрена эволюция систем обработки данных и компьютерных решений для экспериментов в области физики элементарных частиц.

Research in High Energy and Nuclear Physics requires significant computing resources and sophisticated software for data processing, handling, and analysis. A large increase in data volume and data complexity at the LHC created a shortage of computing cycles, and HPC and cloud computing systems stepped in to help the LHC to achieve its physics goals. Already now the managed data volume of the LHC experiment is close to 0.5 EB. Data increase and complexity required new approaches and algorithms to handle data, to process and to analyze them. The ATLAS and other LHC experiments have designed a new generation of data and workload management system to cope with the LHC challenge. The above challenges are not specific for the LHC or NICA, it is also a hot topic for new facilities, such as LSST, DUNE or SKA. The HENP experiments have launched several R&D projects to address High-Luminosity LHC challenge, and one of them is a “scientific data lake”. JINR and many Russian Universities actively participate in it and play a vital role to implement a “data lake” prototype. In this paper, we will describe the HENP computing model evolution and new developments towards HL-LHC challenges.

PACS: 29.85.-c

*E-mail: alexei.klimentov@cern.ch

ВВЕДЕНИЕ

Исследования в области физики высоких энергий (ФВЭ) и ядерной физики (ЯФ) невозможны без использования значительных вычислительных мощностей и программного обеспечения для обработки, моделирования и анализа данных. Это определяется рядом факторов:

— большими объемами информации, получаемыми с установок на современных ускорителях;

— сложностью алгоритмов обработки данных;

— статистической природой анализа данных;

— необходимостью (пере)обрабатывать данные после уточнения условий работы детекторов и ускорителя и/или проведения калибровки каналов считывания;

— необходимостью моделирования условий работы современных установок и физических процессов одновременно с набором и обработкой «реальных» данных.

Введение в строй Большого адронного коллайдера (LHC) [1], создание и запуск установок такого масштаба, как ATLAS, CMS, ALICE [2–4], новые и будущие проекты класса мегасайенс (FAIR [5], XFEL [6], NICA [7]), характеризующиеся сверхбольшими объемами информации, потребовали новых подходов, методов и решений в области информационных технологий. Во многом это связано:

— со сложностью современных детекторов и количеством каналов считывания, например, детектор ATLAS размером 44×25 м и массой 7000 т имеет 150 млн датчиков для считывания первичной информации;

— со скоростью набора данных (до 1 ПБ/с);

— с международным характером современных научных сообществ и требованием доступа к информации для тысяч ученых из десятков стран (в научные коллаборации на LHC входят более 8000 ученых, сравнимое количество ученых будет работать в проектах FAIR и NICA, и в последние годы эксперименты в области астрономии и астрофизики, такие как LSST (обсерватория им. Веры Рубин) и AMS-02 (Alpha Magnetic Spectrometer), становятся сопоставимыми с ускорительными экспериментами в области элементарных частиц по объемам данных и количеству участников);

— с высокими требованиями к обработке данных и получению физических результатов в относительно короткие сроки.

Научный прорыв 2012 г. — открытие бозона Хиггса [8] — стал триумфом исследований на LHC. В последующие годы в экспериментах на LHC исследовали свойства новой частицы, одновременно были увеличены светимость и энергия коллайдера. В настоящее время (2020 г.) идет подготовка к третьему этапу работы LHC и начаты научно-исследовательские проекты в области детекторов, программного обеспечения и компьютинга для этапа

супер-LHC (High Lumi LHC — этапа работы с высокой светимостью). Современные эксперименты работают с данными в эксабайтном диапазоне и являются заметными «поставщиками» так называемых больших данных и методов работы с ними. Как и в случае со Всемирной паутиной (WWW), технологией, созданной в ЦЕРН для удовлетворения растущих потребностей со стороны ФВЭ к обмену информацией между учеными и совместному доступу к ней, что вызвало бурное развитие информационных технологий и систем связи в конце XX в., технологии больших данных начинают влиять на исследования и в других научных областях, включая нанотехнологии, астрофизику, биологию и медицину. Большие данные часто являются связующим звеном, объединяющим разработки из различных областей науки в единый мегапроект [9]. В речи, произнесенной всего за несколько недель до того, как был потерян в море недалеко от калифорнийского побережья в январе 2007 г., Джим Грей, пионер программного обеспечения для баз данных и исследователь, работавший в Microsoft, изложил в общих чертах аргументы, которые показывают, что «эксапоток» научной информации существенно преобразует практику науки [10]. Доктор Грей назвал это изменение «четвертой парадигмой» [11,12].

Проблемы, которые ставятся в связи с развитием областей науки с большими объемами данных (а это не только физика элементарных частиц, но и вычислительная химия и биология, науки о Земле и климате), многочисленны. Данные эксабайтного масштаба, как правило, распределены и должны быть доступны для больших международных сообществ. Для управления большими массивами данных и их обработки необходимы многоуровневые интеллектуальные системы, системы управления потоками данных, контроля и мониторирования, а также системы хранения информации.

Вопросы разработки компьютерной модели, архитектуры распределенных и параллельных вычислительных систем для обработки данных, основополагающие принципы и модели таких систем, анализ алгоритмов параллельных вычислений обсуждаются в работах начала XXI в. Э. Таненбаума, М. ван Стеена [13] и Вл. В. Воеводина [14]. Следует отметить, что во второй половине XX в. классические работы Н. Н. Говоруна [15] о применении ЭВМ для обработки и анализа данных в области физики частиц, совпавшие по времени с запуском новых ускорителей в СССР (У-10, У-70), ЦЕРН (PS, SPS) и США (AGS, SLAC), оказали большое влияния на развитие методики обработки данных в области ФВЭ и ЯФ и во многом заложили основу для будущих компьютерных моделей обработки данных.

Уже на этапе создания архитектуры и компьютерной модели для экспериментов на Большом адронном коллайдере (1998–2001 гг.) стало очевидным, что хранение и обработка данных не могут быть выполнены в одном центре, даже таком крупном, как ЦЕРН. Следует отметить, что это понимание было вызвано техническими, финансовыми и социологическими причинами, в том

числе и отсутствием на начало XXI в. решений, предложенных десятилетием позже ведущими коммерческими ИТ-компаниями.

Большой адронный коллайдер — уникальный ускоритель, в котором каждые 50 нс происходит столкновение протонов при энергии 13 ТэВ с рождением около 1600 заряженных частиц, каждая из них регистрируется и анализируется триггером высокого уровня. В результате работы триггера около 1000 событий ежесекундно отбираются для дальнейшей обработки и анализа. Управляемый объем данных современного физического эксперимента близок к 500 ПБ. С 2014 г. физики международного сотрудничества ATLAS обрабатывают и анализируют более 1,4 ЭБ данных в год. Беспрецедентный объем информации, поступавший во время второй фазы работы LHC (2015–2018 гг.), и ожидаемое возрастание объема информации на следующих этапах работы коллайдера, как и требования к вычислительным комплексам на современных и будущих установках (FAIR, XFEL, NICA), повлияли на необходимость разработки новой компьютерной модели, методики и методов управления загрузкой, создания новых систем для обработки данных. Также необходимым условием для своевременной обработки данных и получения физического результата в короткие сроки (в течение года) стал переход от использования гомогенной вычислительной среды (грид) к гетерогенной вычислительной инфраструктуре с применением суперкомпьютеров (СК), академических и коммерческих центров «облачных вычислений», «волонтерских» компьютеров и отдельных вычислительных кластеров.

Еще на раннем этапе развития компьютерной модели LHC (2000-е гг.) было принято решение объединить существующие и вновь создаваемые вычислительные центры (более 200) в распределенный центр обработки данных и сделать это таким образом, чтобы физики из университетов и научных организаций участвующих стран имели равные возможности для анализа информации. В результате работы физиков, ученых и инженеров в области ИТ была создана система, известная сегодня как WLCG (Worldwide LHC Computing Grid) [16]. На настоящее время WLCG — самая большая академическая распределенная вычислительная сеть в мире, состоящая из около 300 вычислительных центров в 42 странах мира. Более 8000 ученых использовали эти центры для анализа данных коллайдера в поисках новых физических явлений (на рис. 1 показана карта вычислительных центров и проектов, входящих в консорциум WLCG).

Грид-технологии были предложены в конце прошлого века Я. Фостером и К. Кессельманом. Основная концепция грид представлена в их книге «The grid: a blueprint to the new computing infrastructure» [17]. Именно приложения в области ФВЭ и ЯФ привели к широкому использованию грид-технологий и потребовали существенных изменений и развития информационно-вычислительных комплексов в составе физических центров.

Рис. 1. Вычислительные центры и проекты, входящие в консорциум WLCG (май 2020 г.)

В WLCG ежедневно выполняется до 3 млн физических задач, общее пространство хранения данных превышает 1 ЭБ, результаты обработки данных архивируются, распределяются между центрами обработки и анализа данных и поступают непосредственно на «рабочее место» физика. Подобную систему можно сравнить с огромным вычислительным комплексом, узлы которого соединены высокоскоростным интернетом. Объемы передачи данных между центрами составляют до 10 ГБ/с (среднее значение в течение суток). Создание системы заняло около 10 лет и потребовало не только вложений в инфраструктуру вычислительных центров во многих странах мира, но и развития сетевых средств. Для обмена данными между центрами WLCG были созданы две компьютерные сети, ориентированные на задачи LHC: LHCOPN (LHC Optical Private Network) [18] и LHCONE (LHC Open Network Environment) [19]. (Отметим, что этот опыт оказался настолько удачным, что в конце 2019 г. было начато обсуждение создания аналогичной сети DUNEONE, призванной объединить вычислительные центры для обработки данных нейтринных экспериментов.) Создание WLCG стало возможно в результате совместной работы тысяч ученых и специалистов и больших финансовых вложений.

Доктор Фабиола Джанотти (руководитель эксперимента ATLAS в 2008–2013 гг., директор ЦЕРН с 2014 г.) на семинаре, посвященном открытию новой частицы, сказала: «Мы наблюдаем новую частицу массой около 126 ГэВ. Мы не смогли бы провести обработку и анализ данных так быстро, если бы не

использовали грид. Центры во всех странах-участницах эксперимента были задействованы в обработке данных LHC, практически это был стресс-тест для вычислительных мощностей, и грид показал себя высокоэффективной и надежной системой».

Можем ли мы сказать, что LHC и WLCG выполнили поставленную задачу? Если говорить об открытии новой частицы, то да. Ни ускоритель тэватрон (в Национальной ускорительной лаборатории им. Э. Ферми, США), ни большой электрон-позитронный коллайдер LEP (в ЦЕРН) за десятилетия работы не смогли зарегистрировать предсказанную в 1964 г. частицу. Однако более важно получить ответ на следующие вопросы. Достаточно ли классическое решение грид, реализованное в рамках проекта WLCG, для решения задач на следующих этапах работы коллайдера? Как должна развиваться компьютерная модель для этапа супер-LHC (2027–2036 гг.), а также для новых комплексов, таких как FAIR, XFEL, NICA? Ответить на эти вопросы невозможно без понимания логики создания проекта WLCG и тех условий, в которых была разработана и реализована первая компьютерная модель распределенных вычислений для LHC. Необходимо проанализировать ограничения компьютерной модели и понять, насколько они имеют фундаментальный характер, почему потребовалось создание новой компьютерной модели и распределенной системы обработки данных для второго и последующих этапов работы LHC, применима ли новая компьютерная модель для экспериментов на установках класса мегасайенс в «эпоху больших данных» [20].

Работы по созданию концепции и архитектуры систем для распределенной обработки данных экспериментов в области ФВЭ и ЯФ, а также астрофизики были начаты в конце XX в. Создание программного пакета Globus Toolkit [21], который на десятилетия стал основным набором инструментов для построения грид-инфраструктуры, — важнейший этап в развитии концепции грид. Тогда же были разработаны и реализованы первые сервисы для обнаружения ошибок и защиты информации, сервисы управления данными и ресурсами, сформулированы требования по взаимодействию сервисов внутри грид-систем. Следует отметить пионерские работы по развитию и созданию грид в России, в первую очередь в ЛИТ ОИЯИ и НИИЯФ МГУ [22], а также работы НИВЦ МГУ, в которых рассмотрены вопросы эффективности работы суперкомпьютерных центров и проблемы их интеграции [23]. Многие из предложенных идей повлияли на развитие архитектур вычислительных систем и систем обработки и управления данными, а также на развитие компьютерной модели современных физических экспериментов.

Важным этапом развития систем для обработки данных явилось обоснование принципов построения и архитектуры системы, а также разработка методов планирования выполнения заданий. Это позволило создать принципиально новое программное обеспечение, необходимое для управления данными

и заданиями в распределенной среде, разработать методы оценки эффективности функционирования систем управления загрузкой и работы вычислительных центров (в рамках грид-инфраструктуры) и методы распределения задач обработки данных с целью оптимального использования вычислительного ресурса.

Компьютерная модель обработки данных физического эксперимента прошла в своем развитии много этапов: от модели централизованной обработки данных, когда все вычислительные ресурсы были расположены в одном месте (как правило, там же, где находилась экспериментальная установка), до разделения обработки и анализа, которые по-прежнему велись централизованно, и моделирования данных, проводившегося в удаленных центрах. В эпоху LHC была предложена и реализована иерархическая компьютерная модель MONARC [24]. Следующим этапом стала модель равноправных центров внутри однородной грид-инфраструктуры — «смешанная компьютерная модель» [25, 26]. В настоящее время компьютерная модель предполагает равноправное использование центров грид и интегрированных с грид ресурсов облачных вычислений и суперкомпьютерных центров в рамках единой гетерогенной среды. Для дальнейшего развития компьютерной модели для этапа супер-LHC и комплексов FAIR, XFEL, NICA потребовались разработки концепции и архитектуры единой федеративной киберинфраструктуры в гетерогенной вычислительной среде [27], а также развитие концепции «озеро научных данных» [28].

Для обработки и управления большими массивами данных необходимы многоуровневые интеллектуальные системы и системы управления потоками заданий. Создание таких систем имеет свою эволюцию, сравнимую по количеству этапов с развитием компьютерной модели физических экспериментов: от набора программ, написанных на скриптовых языках и имитирующих работу планировщика в рамках одного компьютера, до систем пакетной обработки, таких как LSF [29] или PBS [30], с последующей разработкой пакетов программ управления загрузкой промежуточного уровня грид (HTCondor [31]), а на последнем этапе развития — до разработки и создания высокоинтеллектуальных систем управления загрузкой (AliEN, Dirac, PanDA [32–34]). Эти системы способны управлять загрузкой и позволяют обрабатывать данные одновременно в сотнях вычислительных центров. Практическое использование систем управления загрузкой показало их ограничения по параметрам масштабируемости, стабильности, возможности использования компьютерных ресурсов вне грид. Выявились трудности при интегрировании информации глобальных вычислительных сетей с информацией об имеющемся вычислительном ресурсе, быстроте «захвата» этого вычислительного ресурса (что стало особенно заметно при переходе от модели MONARC к «смешанной компьютерной модели», а также при использовании суперкомпьютеров и коммерческих ресурсов облачных вычислений). Другой существен-

ной проблемой стала реализация способа разделения вычислительного ресурса между различными потоками заданий, таких как обработка, моделирование и анализ данных, а также предоставление вычислительного ресурса для задач эксперимента («виртуальной организации»), отдельных научных групп и ученых в рамках установленных квот использования вычислительного ресурса.

Таким образом, запуск Большого адронного коллайдера и создание новых ускорительных комплексов класса мегасайенс, характеризующихся сверхбольшими объемами информации и многотысячными коллективами ученых, обусловили новые требования к информационным технологиям и программному обеспечению. В эти же годы произошло качественное развитие информационных технологий, появились коммерческие вычислительные мощности, превышающие возможности крупнейших вычислительных центров в области ФВЭ и ЯФ, резко повысилась пропускная способность глобальных вычислительных сетей. Требования по обработке данных на LHC и развитие ИТ привели к необходимости решения фундаментальной проблемы по разработке систем нового поколения для глобально распределенной обработки данных и новой компьютерной модели физического эксперимента, позволяющей объединять различные вычислительные ресурсы и включать их (например, интегрировать ресурсы грид и суперкомпьютеры в единую вычислительную среду).

1. РАЗВИТИЕ ВЫЧИСЛИТЕЛЬНОЙ МОДЕЛИ ЭКСПЕРИМЕНТОВ В ОБЛАСТИ ФИЗИКИ ЭЛЕМЕНТАРНЫХ ЧАСТИЦ И АСТРОФИЗИКИ

В данном разделе кратко рассмотрим компьютерные модели наиболее значимых экспериментов в области физики элементарных частиц и ядерной физики на ускорителях и коллайдерах в последние десятилетия, а также астрофизических экспериментов AMS и AMS-02 на Международной космической станции (МКС). В силу специфики эксперимента AMS-02 в нем была предложена и реализована одна из первых компьютерных моделей для распределенной обработки данных. Более подробно рассмотрим иерархическую компьютерную модель MONARC для экспериментов на LHC и эволюцию моделей обработки данных в последние годы.

1.1. Этапы развития компьютинга в области физики высоких энергий, ядерной физики и астрофизики. ***1.1.1. Компьютерные модели обработки данных в физике элементарных частиц до запуска Большого адронного коллайдера.*** В табл. 1 представлены сравнительные характеристики экспериментов в области физики частиц за последние 60 лет. Первым прорывом явилось использование компьютеров для онлайн- и офлайн-обработки дан-

ных и магнитных лент для архивирования информации с последующей обработкой данных на машинах серий ЕС и IBM. Развитие вычислительной техники в конце 1980-х гг., появление поколения машин серий СМ, PDP, VAX, а также сетевого протокола DECNET наряду с созданием сегментов скоростной локальной сети Ethernet, функционирующей с пропускной способностью 10 Мбит/с, впервые позволили перейти от изолированных ЭВМ к кластерам из нескольких машин и рабочих станций (как правило, аппаратно-совместимых с основной ЭВМ), а также «связать» обработку в реальном масштабе времени (онлайн) с постобработкой (офлайн) и передачей данных между центрами онлайн- и офлайн-обработки. Такие работы практически одновременно были выполнены в ЦЕРН (эксперименты UA1 и UA2) [35], ОИЯИ и ИТЭФ (адронный калориметр установки L3) [36]. Работы в ИТЭФ явились

Таблица 1. **Характеристики экспериментов в области физики частиц за последние 60 лет**

Год	Число сотрудников эксперимента	Объем данных / технология хранения и обработки данных и информации
Конец 1950-х	2–3	Килобиты / записи в рабочих журналах
1960-е (У-7)	10–15	Килобайты / перфокарты, бумажные носители
1970-е (У-10, У-70, PS, AGS)	~ 35	Мегабайты / магнитные ленты Онлайн-обработка: PDP 8; офлайн-обработка: ЕС, IBM 360
1980-е (SPS, У-70)	~ 100	Гигабайты / магнитные ленты и диски Онлайн-обработка: Caviar, PDP 70, VAX, СМ4; офлайн-обработка: ЕС, IBM 370, БЭСМ 6, VAX 8800
1990-е (LEP, SLAC, тэватрон, RHIC)	700–800	Терабайты / магнитные ленты, диски Онлайн-обработка: VAX, специальные процессоры; офлайн-обработка: ЕС, IBM 370, VAX 8800, Appollo, SGI, Sun
После 2010-го (LHC, AMS, Belle, LSST, SKA)	~ 3000	Петабайты / магнитные ленты, диски Онлайн-обработка: кластеры, графические процессоры; офлайн-обработка: грид
После 2030-го (HL-LHC, EIC, DUNE)	Более 3000	Эксабайты / магнитные ленты (?), диски, «облачные» ресурсы Онлайн-обработка: специальные процессоры, кластеры; офлайн-обработка: «облачные» ресурсы, грид, суперкомпьютеры

важным этапом в развитии систем обработки данных на коллайдере LEP [37], на котором в рамках эксперимента L3 в 1993 г. была впервые реализована распределенная обработка данных. Локальной сетью Ethernet (с пропускной способностью 100 МБ/с) были связаны компьютеры, работающие в точке пересечения пучков $N2$ (ВЦ для онлайн-обработки) и центром обработки данных эксперимента L3 в ЦЕРН (ВЦ для офлайн-обработки), находившегося в 10 км от источника исходных данных (на рис. 2 схематично показан классический подход по использованию вычислительных мощностей в физическом эксперименте на всех этапах управления данными, а на рис. 3 — управление потоком данных от эксперимента к центру обработки до принятия концепции грид). Это решение позволило использовать центр онлайн-обработки в период заполнения ускорителя, а также во время его плановых остановок совместно с центром офлайн-обработки. Для конечного пользователя данная система выглядела как единый вычислительный комплекс. Кроме того, это позволило перестать ежедневно перевозить магнитные ленты, так как данные передавались по мере набора в центр офлайн-обработки и архивировались в нем. Дальнейшее развитие ВТ, ИТ и увеличение пропускной способности глобальной вычислительной сети (WAN) позволили создать распределенную обработку для эксперимента AMS.

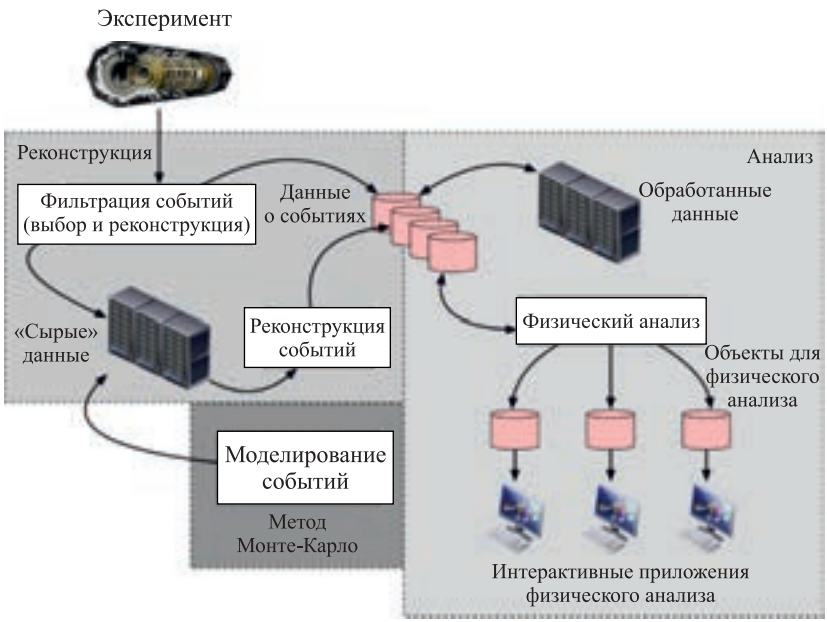

Рис. 2. Роль информационных технологий в физике высоких энергий и ядерной физике

Для эксперимента AMS существуют следующие потоки информации:

— команды, поступающие со станций контроля работы установки;

— телеметрия от станции;

— информация о состоянии детектора (в ФВЭ и ЯФ имеющая название slow control: показания датчиков, измеряющих температуру, напряжение, давление и т. д.; в NASA эти данные называются H&S — health and status);

— научные данные с установки.

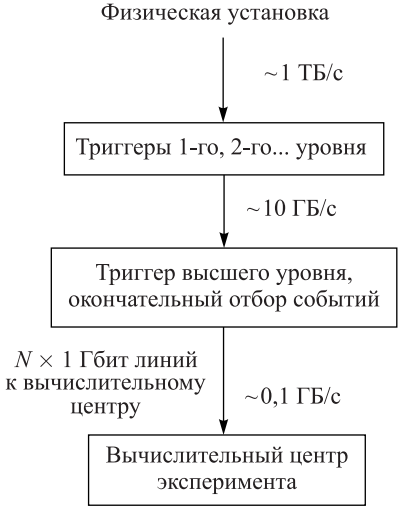

Рис. 3. Поток данных физического эксперимиента

В AMS-02 перечисленная выше информация с МКС поступает в центр NASA (Marshall Space Flight Center — MSFC, Алабама, США), буферизуется на серверах AMS-02, установленных в MSFC. Телеметрия и информация H&S передаются в POCC (Payload Operations and Control Center — центр контроля работы AMS-02), а научные данные передаются в SOC (Science Operations Center — центр научной обработки данных), оба центра находятся в ЦЕРН. Первичная обработка данных проходит в ЦЕРН, после чего данные распределяются между всеми центрами AMS-02 в Европе, США и Азии для физического анализа. Моделирование методом Монте-Карло ведется более чем в 10 центрах по всему миру. Следует отметить, что наряду с основным центром контроля и управления (POCC) в ЦЕРН существуют сателлитные центры во многих странах, они имеют функции контроля и мониторирования (похожая концепция была применена позже в эксперименте CMS на LHC, когда основной центр управления находится в ЦЕРН, рядом с экспериментальной установкой, а сателлитные центры — в ОИЯИ (Дубна, Россия) и Национальной ускорительной лаборатории им. Э. Ферми (США)).

Концепция и архитектура системы управления данными, а также основное программное обеспечение были изначально реализованы для моделирования физических процессов, а после начала работы детектора на МКС — и для обработки данных эксперимента [38, 39]. В те же годы (1997–2002 гг.) группы, работавшие на ускорителях SLAC (эксперимент BaBar) и тэватроне (Национальная ускорительная лаборатория им. Э. Ферми, эксперименты D0,

CDF), пытались найти оптимальное решение для организации обработки данных как в центральном ВЦ (находящемся географически в той же точке, где и установка, т. е. источник данных), так и в удаленных ВЦ (в том числе и в других странах).

В лучшем случае удаленные ВЦ использовались для моделирования событий и/или работы детектора методом Монте-Карло, и результаты моделирования в каждом случае были доступны для пользователей после их передачи (через WAN или на магнитных носителях) в SLAC или Национальную ускорительную лабораторию им. Э. Ферми. Долгие годы AMS был единственным экспериментом в области физики частиц, в котором обработка данных и установка находились в разных географических точках. Разработка и создание системы распределенной обработки данных эксперимента AMS стало первой попыткой предложить новый архитектурный подход к реализации глобальной системы обработки данных на распределенных вычислительных ресурсах.

К середине 1990-х гг. стало очевидным, что дальнейшее развитие систем обработки данных для будущих ускорителей невозможно без создания новых информационных технологий, а также кардинального пересмотра компьютерной модели для экспериментов в области ФВЭ и ЯФ. Следует отметить, что в то время развитие коммерческой индустрии ИТ было еще не таким стремительным, как в последующие десятилетия, а после военных приложений и приложений по исследованию климата, физики частиц являлось одним из наиболее информационноемких. Также следует отметить, что изначально запуск ускорителя LHC планировался в 2000–2004 гг., когда не существовало нынешних гигантов ИТ-индустрии (в те годы Google разрабатывал поисковики, а Amazon занимался продажей книг через интернет), что требовало разработки новых информационных технологий (а также их апробации) в сравнительно короткие сроки.

1.1.2. Распределенная иерархическая компьютерная модель для обработки данных Большого адронного коллайдера. Еще на раннем этапе развития компьютерной модели LHC (конец XX в.) было принято решение объединить существующие и вновь создаваемые вычислительные центры (более 300 центров в настоящее время) в распределенный центр обработки данных и сделать это таким образом, чтобы физики из университетов и научных организаций участвующих стран имели равные возможности для анализа информации. Для такого решения было несколько причин.

• Экономические и социологические:

— даже предварительная оценка будущего объема данных LHC не позволяла просто расширить существующий ВЦ, даже такой крупный, как в ЦЕРН, и использовать его для хранения, обработки и анализа данных. Требовались капитальные вложения в инфраструктуру, и в случае использования централизованной модели взнос стран-участниц в бюджет организации мог

существенно возрасти, при этом ЦЕРН должен был одновременно обеспечить строительство самой «машины» и сопутствующей инфраструктуры;

— количество ученых, участвовавших в экспериментах на LHC, уже на первом этапе заявок было близко к 5 тыс. (в настоящее время около 9 тыс.), и они представляли более чем 50 стран мира. В случае централизованного решения анализ данных в ВЦ ЦЕРН создавал неравноправные условия для стран, находящихся на значительном расстоянии от Женевы (таких как Россия, США, Япония, Австралия, Канада), доступ к данным для них был бы не столь эффективен, как для стран Западной Европы;

— многие страны, университеты, исследовательские институты имели значительные вычислительные мощности и были заинтересованы в их развитии и использовании;

— экономическая ситуация во многих странах мира требовала вложений в национальные проекты и создания рабочих мест в странах ЕС, поэтому идея дополнительного финансирования компьютерных мощностей ЦЕРН не была поддержана экспериментами. Одновременно идея о расширении национальных ВЦ для потребностей LHC была воспринята позитивно мировым сообществом.

• Технические:

— ни ЦЕРН, ни другие центры ФВЭ и ЯФ не имели опыта строительства ВЦ для обработки данных в мультипетабайтном диапазоне и одновременного доступа к данным тысяч пользователей;

— характеристики будущего центра в части потребляемой мощности и систем охлаждения не могли быть реализованы на территории ЦЕРН в Швейцарии и Франции без изменения двухсторонних соглашений организации с этими странами;

— использование суперкомпьютера (или нескольких СК) для проведения централизованной обработки данных не позволяло решить вопрос анализа данных, не говоря уже о стоимости такого решения;

— ПО физических экспериментов (а это 4 млн инструкций кода) не было оптимизировано для суперкомпьютеров и, в частности, для параллельных вычислений и графических процессоров;

— существующие в то время технологии иерархического гибридного хранения данных (диск и лента), например CASTOR [40], не позволяли эффективно перемещать файлы между постоянным (лента) и временным (диск) хранилищами с частотой и объемами, требуемыми для обработки будущих данных LHC в одном центре;

— оценка и прогнозирование возможностей WAN не гарантировали эффективнного удаленного доступа к данным;

— требования к вычислительному и дисковому ресурсу значительно менялись в течение подготовки экспериментов на LHC. В табл. 2 приведены данные, по которым видно, как менялась оценка необходимого ресурса для

Таблица 2. **Изменение оценки вычислительных ресурсов, необходимых для эксперимента ATLAS**

Год	Дисковый ресурс, ТБ	Вычислительный ресурс, MIPS	Комментарий
1995	100	10^7	Техническое предложение по ПО и вычислительным мощностям эксперимента
2001	1900	$7 \cdot 10^7$	Рецензирование требований экспериментов на LHC по ПО и вычислительным мощностям
2005	70 000	$55 \cdot 10^7$	Окончательное техническое предложение ATLAS
2011	83 000	$100 \cdot 10^7$	Оценка по результатам первого года работы

эксперимента ATLAS. Разница в оценке необходимых мощностей составила три порядка для дискового ресурса и два порядка для вычислительного ресурса между временем подачи меморандума о создании эксперимента и этапом начала работы ускорителя.

Ниже рассмотрим компьютерную модель для распределенной обработки данных применительно к экспериментам на LHC, для которых это проявилось наиболее явно. Эксперименты на KEK (Belle II), LSST, RHIC и будущих комплексах FAIR, NICA, астрофизический эксперимент AMS-02 и нейтринный эксперимент DUNE приняли аналогичную компьютерную модель.

В 1998 г. был создан проект MONARC (Models of Networked Analysis at Regional Centres for LHC Experiments) под общим руководством профессора Калифорнийского технологического института Харви Ньюмана.

Задачей проекта являлась разработка компьютерной модели для экспериментов на LHC, а также пакета моделирования, позволяющего провести анализ необходимых аппаратно-программных средств и компонентов (пропускной способности WAN, количества и характеристик линий передачи данных) для реализации такой модели. Перечислим основные результаты проекта MONARC:

— предложение распределенной компьютерной модели для обработки и анализа данных LHC;

— концепция иерархии центров обработки, моделирования и анализа данных (введение определения трех уровней центров: Tier-0 (T0), Tier-1 (T1), Tier-2 (T2); строгое определение функций для центров каждого уровня);

— создание пакета программ MonALISA [41], который изначально позволял проводить моделирование распределенной работы центров при условии, что архитектура отвечает критериям, определенным в этом проекте (по мере

изменения компьютерной модели для экспериментов в области ФВЭ и ЯФ функции пакета стали более ограниченными, и в настоящее время он используется в одном из экспериментов на LHC для мониторирования работы ВЦ, входящих в инфраструктуру грид).

Основным аргументом при выборе иерархической компьютерной модели послужило предположение о том, что пропускная способность WAN не позволит передавать данные в объемах, необходимых для физического анализа между всеми центрами обработки. На рис. 4 схематично представлена компьютерная модель, предложенная проектом MONARC. Следует отметить, что в последние годы XX в. Калифорнийский технологический институт (КТИ) и Лаборатория стэнфордского линейного ускорителя (SLAC, США) были «законодателями моды» в вопросах архитектуры систем обработки и управления данными и в вопросах создания ПО для физических экспериментов. Это было связано не только с тем, что Х. Ньюман (КТИ) и Р. Моунт (SLAC) руководили в то время компьютингом самого крупного эксперимента в области ФВЭ — BaBar, но и фактом, что исторически сильные позиции российских физиков и ИТ-специалистов были подорваны экономической ситуацией в стране, когда многие из них перестали заниматься наукой. Кроме того, ЦЕРН не смог провести широкой дискуссии по вопросам компьютерной модели для экспериментов на LHC. Отметим, что решения «западного побережья США»: иерархическая компьютерная модель для обработки данных, строгое определение функций центров на каждом уровне и использование объектно-ориентированной базы данных Objectivity — стоили больших денег и принесли много проблем, преодоление которых заняло годы. Таким образом, в начале XXI в. сообществом ФВЭ и ЯФ была принята концепция, предложенная MONARC, и было решено использовать грид-технологии для ее реализации. В конце 2001 г. было предложено создать консорциум, который отвечал бы за компьютинг для LHC, — World LHC Computing Grid (WLCG). Создание WLCG стало важным шагом в развитии компьютинга для экспериментов в области физики частиц, потому что ни один ВЦ (каким бы большим он ни был) не имел более монополии на принятие решений. Возглавил проект д-р Лес Робертсон (ЦЕРН), им были определены два основных направления работы консорциума:

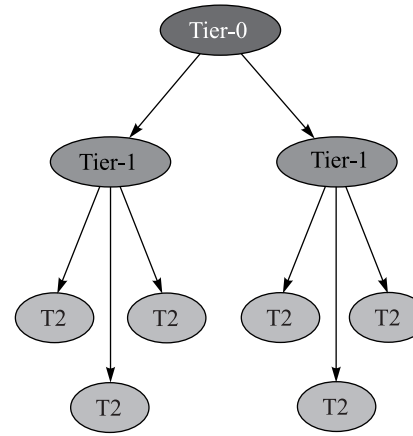

Рис. 4. Иерархическая компьютерная модель проект (MONARC)

— вычислительные ресурсы (центры по всему миру, которые должны стать единым распределенным центром обработки данных LHC);

— программное обеспечение (которое необходимо разработать и использовать, чтобы скрыть сложности инфраструктуры и предоставить «прозрачный» доступ к вычислительным ресурсам).

Кроме того, д-р Робертсон в своем письме руководителям ведущих экспериментов признал необходимость широкой дискуссии и подчеркнул, что проект MONARC должен продолжать работы по развитию компьютерной модели для экспериментов на LHC, но форумом для обсуждения и принятия решений становятся рабочие совещания консорциума WLCG. В октябре 2003 г. состоялось рабочее совещание, на которое были вынесены следующие вопросы:

— обсуждение компьютерной модели для будущих экспериментов в области физики частиц;

— создание единого распределенного центра для обработки данных LHC (каким образом и с использованием каких технологий сотни центров должны быть организованы в «единый» распределенный центр обработки);

— организация и финансирование центров обработки данных (как будет организована работа центров, кем и как они будут финансироваться);

— разработка ПО для управления данными экспериментов и потоками заданий для их обработки в случае принятия концепции распределенной компьютерной модели.

На рис. 5 показано, как должно было измениться управление потоком данных для экспериментов на LHC по сравнению с экспериментами на LEP и коллайдерах RHIC, SLAC, тэватрон. Для новой компьютерной модели не существовало вычислительной инфраструктуры, ее необходимо было создать. Для реализации грид-инфраструктуры в рамках ЕС в 2004 г. был начат проект EGEE (Enabling Grid for E-Science in Europe) [42], который имел три этапа и завершился в 2010 г. созданием глобальной грид-инфраструктуры (параллельно с EGEE были реализованы национальные проекты для стран, не входящих в ЕС: Open Science Grid (США), NorduGrid (Норвегия, Дания, страны Балтии, Украина, Швейцария), Russian Data Intensive Grid (RDIG, Россия и ОИЯИ). Руководителем проекта RDIG стал профессор МГУ В. А. Ильин; во многом благодаря его инициативе российские центры и разработки стали заметными участниками WLCG, аналогичные проекты были реализованы в Австралии, Индии, Китае, Тайване и Японии. В проектах участвовали около 40 стран, более 100 организаций (университетов, исследовательских институтов, национальных лабораторий). Как будет показано далее, создание трех версий промежуточного программного обеспечения (EGEE, OSG, NorduGrid) привело к определенным сложностям при разработке систем для обработки и управления данными. Создание грид-инфраструктуры для LHC и ФВЭ и ЯФ в целом стало важным этапом в развитии компьютинга, были определены

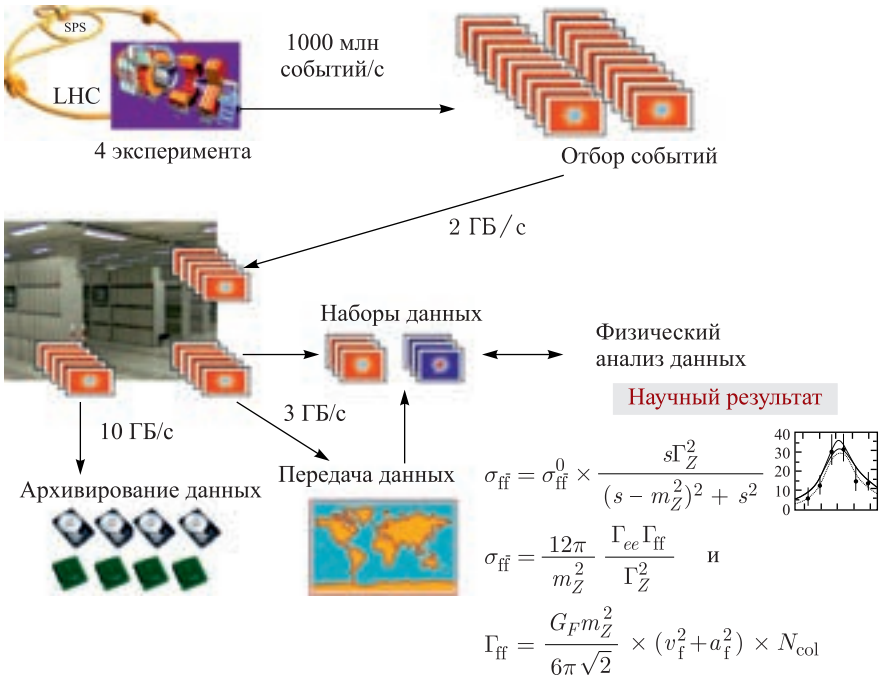

Рис. 5. Потоки данных экспериментов на LHC

11 центров первого уровня (T1), более 100 центров и федераций второго уровня (T2), центром T0 стал ВЦ ЦЕРН. Было определено соотношение разделения вычислительного ресурса между уровнями: T0 — 15 %, T1 — 40 %, T2 — 45 % (центры уровня T3 могли предоставлять вычислительный ресурс на добровольной основе и в ограниченные периоды работы экспериментов). Были подписаны более 40 меморандумов о взаимопонимании в рамках WLCG, в которых финансирующие организации стран брали обязательства обеспечить работу центров первого и второго уровня на срок не менее трех лет.

В рамках проекта MONARC было завершено описание вычислительной модели и определены функции центров каждого из уровней:

— Tier-0 (ЦЕРН) — первичная реконструкция событий, калибровка, постоянное хранение и архивирование полного набора «сырых» и моделируемых данных. Местонахождение основных сервисов (СУБД, репозитории программ) и копий баз данных в случае, если БД находится вне ЦЕРН;

— Tier-1 (11 центров) — архивирование второй копии «сырых» (неприведенных) данных, распределенной между всеми центрами T1, переобработка данных после уточнения калибровок, моделирование методом Монте-Карло, постоянное хранение копий данных, используемых для анализа;

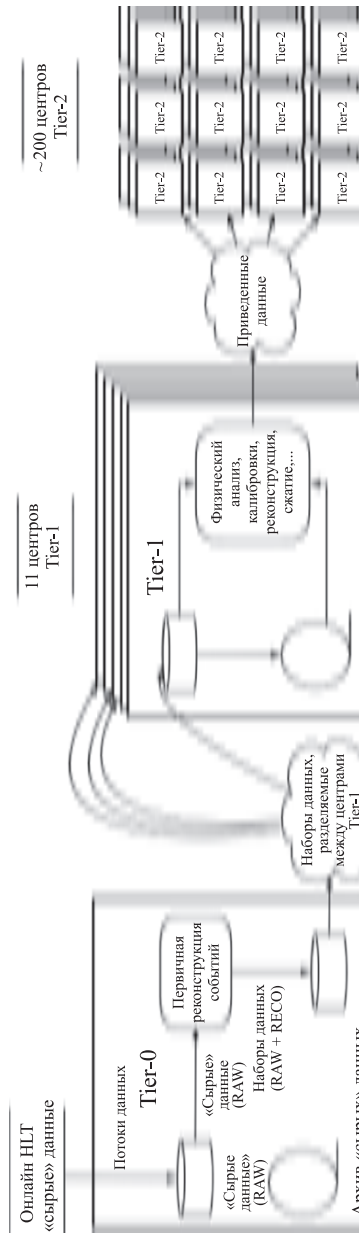

Рис. 6. Компьютерная модель во время первого этапа работы LHC

— Tier-2 (около 100 центров) — временное хранение наборов данных, используемых для анализа, моделирование данных и детекторов, физический анализ;

— Tier-3 (около 50 центров) — университетские кластеры, или центры, предоставляющие ресурсы на добровольной основе, физический анализ данных.

Схематично компьютерная модель для первого этапа работы экспериментов на LHC представлена на рис. 6.

1.1.3 Концепция грид. Грид-технологии были предложены в конце прошлого века Я. Фостером и К. Кессельманом, основная концепция грид представлена в книге «The grid: a blueprint to the new computing infrastructure» (1997 г.). Появление технологии грид совпало по времени с поиском новой компьютерной модели для обработки данных ФВЭ и ЯФ, а также вводом в строй коллайдеров в США (тэватрон, RHIC) и подготовкой к запуску LHC. Как и в случае технологии Всемирной паутины (WWW), созданной в ЦЕРН для удовлетворения растущих потребностей со стороны ФВЭ к обмену информацией между учеными и совместному доступу к ней, вызвавшей бурное развитие информационных технологий и систем связи в конце XX в., когда технологии WWW обеспечили бесшовный доступ к информации, хранящейся на миллионах географически распределенных веб-сайтах, в случае грид предполагалось обеспечить бесшовный доступ к вычислительным мощностям и дисковому ресурсу в ВЦ по всему миру. Из многочисленных существующих определений грид остановимся на следующем: «координированное совместное использование ресурсов и решение проблем в динамичных многопрофильных виртуальных организациях» (Я. Фостер [17]). Таким образом, в концепции грид отсутствовал централизованный контроль над ресурсами, вводилось понятие «виртуальная организация» как совокупность институтов, университетов, групп, объединенных для решения общей задачи в режиме скоординированного использования распределенных вычислительных ресурсов, выделенных для данного проекта. С точки зрения конечного пользователя концепция грид выглядела очень притягательно:

— вычислительный ресурс используется пользователем по потребности;

— ресурс может принадлежать неизвестному владельцу и/или находиться в неизвестном месте;

— владелец ресурса гарантирует компьютерную безопасность ресурса, данных и ПО пользователя;

— программа пользователя будет выполнена на грид-ресурсе.

С точки зрения владельца ресурса концепция грид представлена следующим образом:

— «мой» вычислительный ресурс может быть использован любым авторизованным пользователем;

— «авторизация» не связана административно с организацией, которой принадлежит ресурс;

— ресурс предоставляется не бесплатно.

Важнейшей частью концепции грид-технологий явилось введение понятия «промежуточное программное обеспечение» (ППО, middleware). Первым проектом по созданию ППО явился проект globus [43], который был инициирован Я. Фостером и К. Кессельманом, ими же были определены основные компоненты (и подсистемы) грид-архитектуры. Кратко рассмотрим основные компоненты грид-архитектуры (более подробно они описаны в работах Я. Фостера и К. Кессельмана [17, 18], здесь остановимся на том, что важно для обсуждения развития компьютерной модели и систем для глобально распределенной обработки данных).

Вычислительный элемент (Computing Element, CE) — это вычислительный ресурсный узел грид. На CE выполняются задания пользователей и происходит управление заданиями (запуском, остановкой по ошибке и/или по запросу пользователя или по истечению ресурса, например, оперативной памяти). Состояние и описание ресурсов всех CE публикуется в центральном сервисе (информационной системе) и доступно для всех авторизованных пользователей. Управление загрузкой CE производится через единую систему управления загрузкой.

Элемент хранения (Storage Element, SE) — это узел грид, где хранятся результаты выполнения заданий пользователей на CE. Управление данными, хранимыми на SE, осуществляется через систему управления данными. Состояние и описание ресурсов всех SE публикуется в центральном сервисе (информационной системе) и доступно для всех авторизованных пользователей.

Система управления данными (Distributed Data Management, DDM) — система управления данными, включая хранение, передачу и удаление данных. DDM работает с данными на уровне файлов, конкретные реализации DDM высокого уровня часто предполагают использование набора данных (dataset) как единицы управления данными (например, передача между центрами грид производится на уровне dataset). Контроль доступа к данным основан на понятии группы пользователей. Такую группу составляет как вся виртуальная организация, так и ее отдельные члены.

Система управления загрузкой (Workload Management System, WMS) — система управления пользовательскими заданиями и распределения их для выполнения на грид-ресурсах. WMS выбирает ресурс в соответствии с параметрами задания, находит оптимально подходящий по параметрам грид-ресурс (память, свободное дисковое пространство для промежуточного хранения, географическое расположение входных данных, требование, куда должны быть помещены выходные данные).

Система протоколирования (Logging and Bookkeeping, LB) отслеживает выполняющиеся в грид-инфраструктуре шаги выполнения заданий, хранит

информацию о времени, затраченном на каждый шаг выполнения задания (запуск, инициализацию . . .). Иногда часть функций LB интегрирована с системой управления заданиями, как будет показано ниже, в новейших системах эта информация используется для предсказания поведения WMS и обнаружения аномалий в ее работе.

Система информационного обслуживания (Grid Information System, GIS — информационная система, ИС) отвечает за хранение информации о ВЦ, входящих в грид-инфраструктуру, включая информацию о вычислительной мощности узлов сайта, планируемой остановке ВЦ (например, на профилактическое обслуживание).

Система мониторирования и учета (аккаунтинга) работы грид — основная система для контроля стабильности и эффективности работы грид-инфраструктуры, которая сохраняет информацию о качестве работы грид-сайта на протяжении всего времени его функционирования и одновременно предоставляет информацию в режиме реального времени о состоянии грид-сайта сотрудникам, осуществляющим эксплуатацию ВЦ.

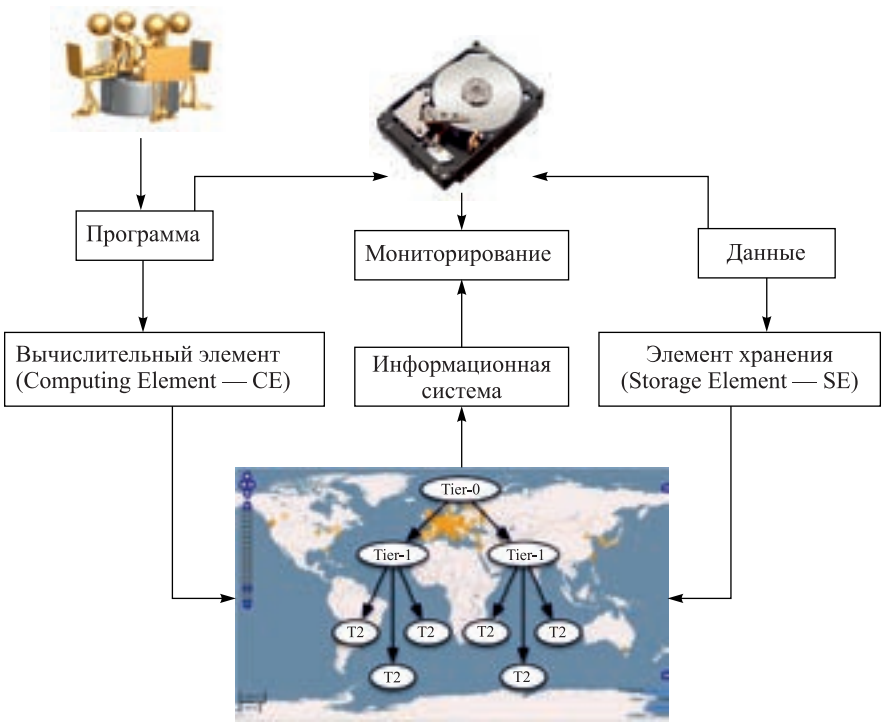

Рис. 7. Компоненты грид-инфраструктуры и их взаимодействие между собой

Система компьютерной безопасности (СБ) должна защитить доступ к грид-инфраструктуре и данным от несанкционированного доступа. СБ рассматривается как средство защиты веб-сервисов и, как правило, реализуется в виде отдельных модулей для сервисов Apache, Axis, Tomcat.

Таким образом, в архитектуре грид основным элементом становится грид-сайт, состоящий из набора «вычислительных элементов» и «элементов хранения». Описание сайта хранится в информационной системе грид, и на сайте установлено промежуточное программное обеспечение, которое дает доступ к элементам CE и SE для авторизованных пользователей через системы управления загрузкой и данными (WMS и DDM соответственно).

На рис. 7 схематично показано взаимодействие различных компонентов грид-инфраструктуры.

1.2. Реализация иерархической компьютерной модели распределенной обработки данных на первом этапе работы Большого адронного коллайдера. Таким образом, к 2004 г. были определены компьютерная модель для экспериментов на LHC и метод реализации компьютерной инфраструктуры на основе технологии грид.

Иерархическая модель MONARC предполагала статическое соответствие $1{:}n$ между центрами уровней T1 и T2 (T3) в предположении, что такие «связки» будут созданы по географическому и/или национальному признаку и сформируют группы ВЦ на постоянной основе. В силу причин политического характера центры T2 в Японии и Китае оказались в «связке» с центром T1 в Лионе (Франция), а не T1 в Тайпее (Тайвань). В силу причин социологического характера центры T2 России, Израиля и Турции оказались в одной «связке» с центром в Амстердаме (так как Нидерландам «не хватило» центров в ЕС). Швейцария оказалась в «связке» с Норвегией. Модель MONARC не допускала нарушения иерархии и отсутствия предопределенного соотношения T1:nT2. Реализация модели была завершена в 2009 г. На начало запуска LHC были определены и зафиксированы обязательства всех центров в рамках консорциума WLCG. Все центры консорциума предоставляли «выделенный» ресурс в течение всего времени участия в WLCG, что, как будет показано далее, не является оптимальным для использования мощностей конкретного вычислительного центра и всего ресурса, доступного для физики частиц в целом. На рис. 8 показана реализация модели MONARC на начало запуска LHC (2009 г.), а также схематично сайты уровня T1 и T2 и их основные функции.

Реализация модели распределенных вычислений, предложенная проектом MONARC, стала значительным шагом в развитии компьютинга в области физики частиц. Более 200 центров, финансируемых более чем 60 агентствами по всему миру, вошли в консорциум WLCG, был получен первый опыт по распределенной обработке данных. Вычислительный ресурс распределялся следующим образом: 15 % находилось в ЦЕРН (уровень T0), 40 % распределялось между 11 центрами уровня T1, 45 % — между центрами уровня T2.

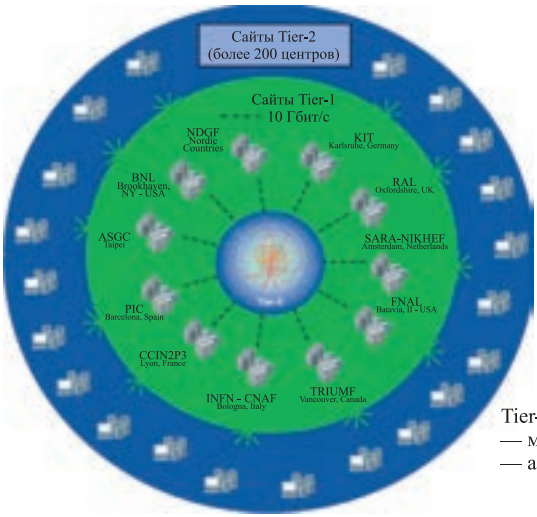

Рис. 8. Организация грид-сайтов WLCG на начало запуска LHC

Следует отметить, что реализованная модель успешно работала в течение первого этапа работы коллайдера, но поддержание ее в рабочем состоянии требовало больших ресурсов как со стороны ВЦ (инфраструктуры), так и со стороны научных коллабораций для поддержания работы сервисов управления данными и загрузкой. Кроме того, ожидание, что гиганты индустрии ИТ (Google, Amazon, Yandex, Microsoft) будут использовать технологии грид и тем самым способствовать развитию промежуточного программного обеспечения, не подтвердилось, поэтому необходимо было проанализировать первый опыт по реализации иерархической модели распределенной обработки данных и определить дальнейшее развитие компьютерной модели для физики частиц в целом исходя из реалий и появления новых игроков на поле ИТ. Этого требовала подготовка как ко второму этапу работы ускорителя LHC, так и к экспериментам на будущих комплексах: KEK (эксперимент Belle II), LSST, FAIR и NICA.

1.3. Ограничения иерархической компьютерной модели MONARC. Реализация модели распределенных вычислений, предложенная проектом MONARC, стала значительным шагом в развитии компьютинга в области физики частиц. В то же время уже на первом этапе работы LHC проявились существенные ограничения данной модели. Перечислим основные из них.

• Определение вычислительного ресурса как совокупности вычислительных узлов, дискового пространства и систем архивирования информации без учета пропускной способности WAN и качества линий связи.

• Статическая методика распределения данных между центрами: а) изначально определено, какой объем данных (реальных и моделируемых) бу-

дет находиться в каждом центре; б) изначально определено, сколько копий данных каждого типа («сырых» и приведенных) будет распределено между центрами обработки.

• Отсутствие понятия «популярность» (востребованность) для данных и групп данных.

• Предложенная методика обработки данных при статическом характере организации вычислительного ресурса и распределения данных между центрами грид-инфраструктуры. Наиболее точно ее можно определить слоганом «задачи обработки идут к данным». Такой подход привел к задержке при обработке и моделировании данных, так как требовал одновременного наличия данных и свободного вычислительного ресурса в одном и том же ВЦ.

• Вычислительный ресурс центров был ориентирован на среднюю загрузку. В результате это привело к недостатку вычислительного ресурса в периоды пиковой нагрузки (работы коллайдера с повышенной светимостью) при анализе и обработке данных и к неоптимальному использованию вычислительного ресурса во время плановых остановок коллайдера, праздников и т. д.

• Ограничения самой модели, предполагающей гомогенность используемого ресурса, наличие ПО промежуточного уровня (middleware) во всех центрах обработки данных.

Основной проблемой стали реализация идеи иерархии ВЦ и статический характер связки центров 1:T1–n:T2, когда любой сбой в работе центра первого уровня (T1) практически останавливал работу всех связанных с ним центров второго уровня (T2), в результате эксперименты лишались мощностей от 1 до 10 центров одновременно. Кроме того, многие центры уровня T2 были мощнее и стабильнее центров уровня T1, но ресурс уровня T2 не использовался оптимальным образом, так как в рамках модели результаты выполнения заданий всегда должны быть переданы в центр уровня T1 и только после этого могло быть создано дополнительное количество копий. Сама передача данных между центрами T2 включала в себя создание до двух промежуточных копий, временного хранящихся в центрах уровня T1 (в случае нестабильной работы конечного центра T2 время хранения таких копий могло достигать нескольких недель).

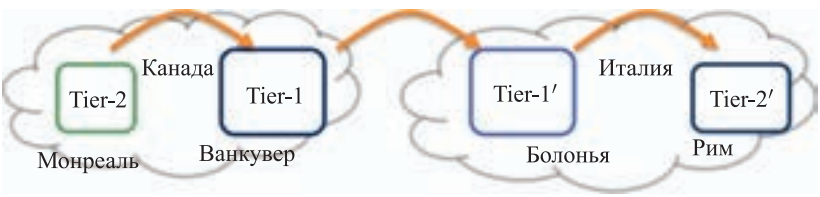

Рис. 9. Многоступенчатая передача данных между центрами уровня T2 при реализации модели MONARC

Эти причины послужили мотивацией для разработки новой концепции распределенной обработки данных и новой модели компьютинга для второго и последующих этапов работы LHC. На рис. 9 схематично показано, как происходила передача данных между двумя центрами уровня T2, находящимися в разных странах, при реализации модели MONARC. Из этого рисунка следует, что требовались, по крайней мере, две дополнительные передачи с промежуточным хранением данных в двух центрах.

1.4. Разработка новой компьютерной модели для распределенной обработки данных. Переход от иерархической модели обработки к смешанной модели в рамках грид-инфраструктуры. При разработке новой модели были введены следующие определения:

— популярность данных (насколько часто задачи обработки, анализа или моделирования обращаются к данным определенного типа, насколько данные популярны у ученых и научных групп, как часто поступают запросы на копирование данных);

— «температура» данных (как со временем меняется частота обращения к определенному набору данных);

— вычислительная среда (вычислительный ресурс, дисковый ресурс и ресурс архивирования, пропускная способность и стабильность глобальной вычислительной сети, т. е. было предложено рассматривать ресурс глобальной вычислительной сети (WAN) совместно с вычислительным ресурсом и ресурсом хранения данных);

— отсутствие предопределения функций центров внутри среды (центры уровня T2 могут выполнять те же функции, что и центры уровня T1, кроме архивирования данных);

— оценка стабильности работы центров и, как результат, решение об использовании их дискового ресурса в качестве постоянного или временного хранилища данных, независимо от уровня центра в классификации WLCG (также стабильность работы центра стала влиять на выбор центра для выполнения высокоприоритетных задач, например задач триггера высшего уровня, которые должны быть выполнены в течение 12 ч). Для этого была разработана методика постоянной проверки стабильности работы центров грид-инфраструктуры (она подробно рассмотрена в п. 1.4.3) и на основании результатов проверки была введена классификация стабильности центров.

1.4.1. Методика определения популярности данных. Классификация данных. Шаги обработки данных в ФВЭ и ЯФ показаны на рис. 10. Единичным объектом при обработке данных является событие, созданное на этапе набора информации с экспериментальной установки и прошедшее сито отбора согласно меню триггеров разных уровней (состав и количество меню соответствуют научной программе эксперимента). Такое событие называется «сырым» или «неприведенным» (RAW). Все события являются независимыми,

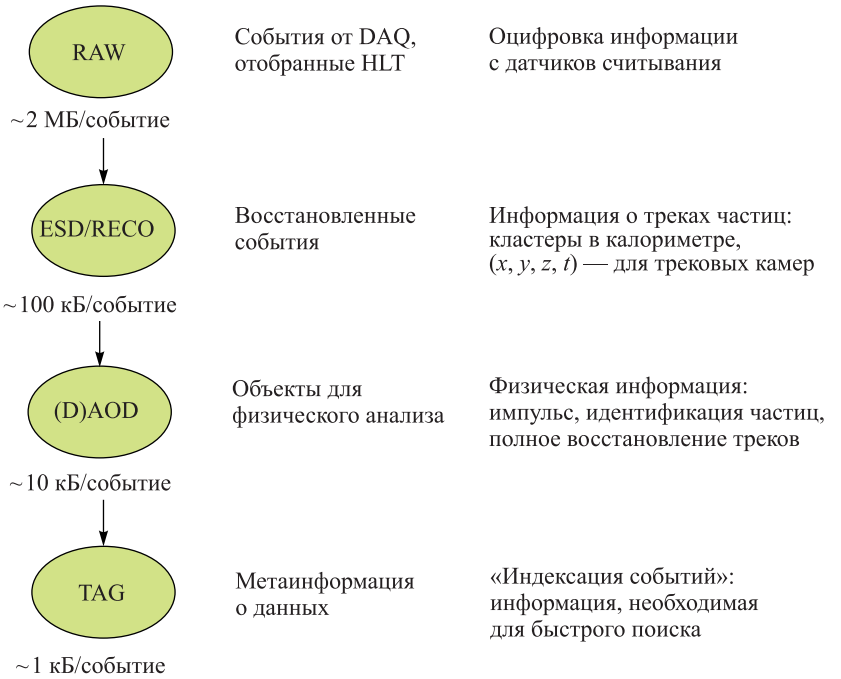

Рис. 10. Шаги обработки данных в ФВЭ и ЯФ

что позволяет применить тривиальный параллелизм при их обработке (пример части «сырого» события показан на рис. 11).

На первом этапе обработки проводится реконструкция события, когда восстанавливаются треки частиц, определяются масса, заряд, импульс и другие параметры, на этом же этапе учитываются калибровочные параметры и неэффективность работы отдельных элементов экспериментальной установки (результат работы программы реконструкции помещается во временное хранилище в формате ESD — Event Summary Data).

На втором этапе обработки создаются данные в формате, необходимом для проведения физического анализа (AOD — Analysis Object Data), окончательным этапом является дополнительный отбор событий и запись информации в табличном виде (формат NTUP, DAOD), удобном для программ анализа данных (например, ROOT [44]). На рис. 12 показана последовательность шагов обработки для реальных событий.

Для этапа моделирования методом Монте-Карло необходимы генерация событий (результат записывается в формате EVGEN), оцифровка событий и имитация «реальных сырых событий» (формат HITS).

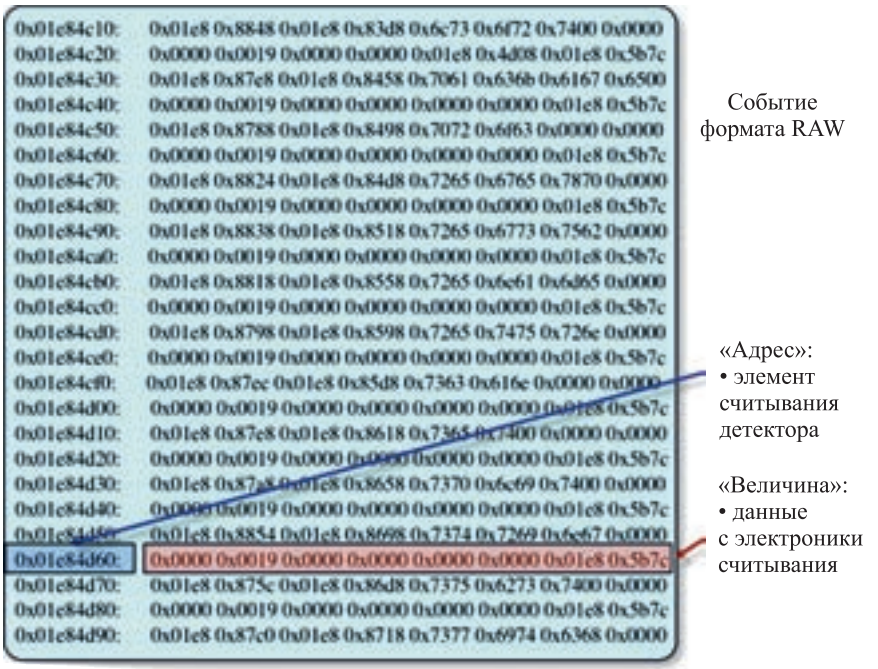

Рис. 11. Пример содержимого неприведенного («сырого») события

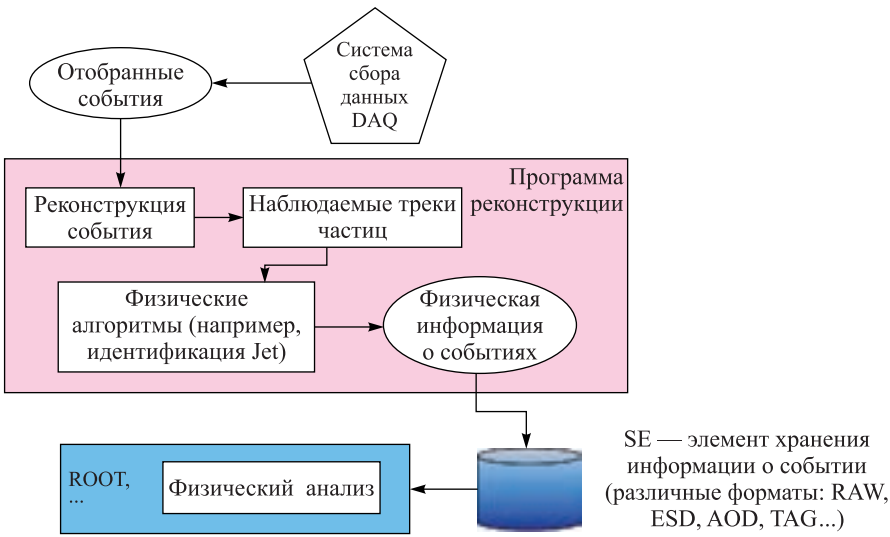

Рис. 12. Последовательность шагов обработки для реальных данных

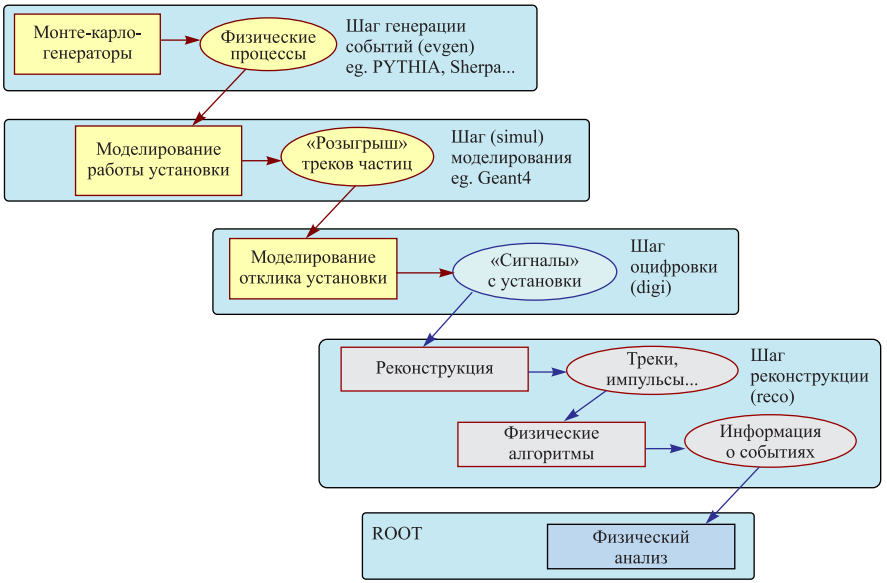

Рис. 13. Последовательность шагов обработки для моделируемых событий

Введено понятие «класс данных». Состав класса определялся следующими параметрами: тип события (моделированные или реальные данные), шаг обработки, формат события (RAW, ESD, AOD, NTUP. . .), версия программного обеспечения, используемого для его обработки, ценность информации и затраты на ее восстановление (на рис. 13 показана последовательность шагов при моделировании событий, важной особенностью является «разбиение» на этапы и запоминание (иногда временное) результатов работы каждого этапа, что позволяет существенно уменьшить время, необходимое для моделирования работы установки и физических процессов, что более подробно рассмотрено в разд. 3).

Так, моделирование событий (этап EVGEN) требует значительного вычислительного ресурса (на рис. 14 показано астрономическое процессорное время в процентах, затраченное на обработку различных типов данных эксперимента ATLAS, видно, что этап моделирования методом Монте-Карло требует наибольшего ЦПУ-ресурса).

1.4.2. Иерархическая модель данных. Предложена следующая модель данных.

Событие — результат моделирования или полученная в результате «слияния» данных с физической установки информация со всех систем считывания для единичного столкновения ускорителя. Событие может быть «сырым» или реконструированным.

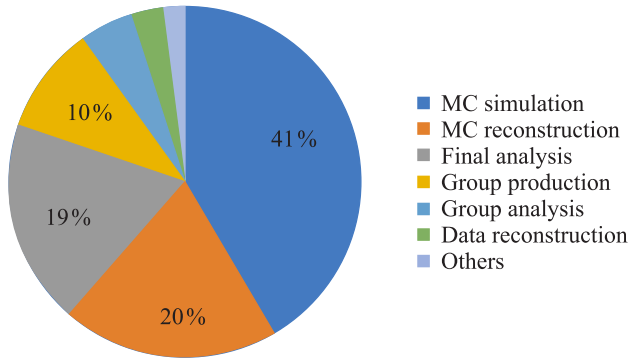

Рис. 14. Астрономическое время, затраченное на различных этапах на обработку и моделирование данных физического эксперимента

Файл — группа событий, как правило, набранных или обработанных последовательно.

Датасет (dataset) — набор данных. Для лучшей гранулярности и организации введено понятие «набор данных», в состав датасета входят файлы с событиями одного формата, созданные в одной версии ПО, произведенные за определенный промежуток времени (один этап набора статистики при одинаковых условиях и параметрах отбора (run) при одном заполнении коллайдера (fill).

Контейнер содержит наборы данных (датасеты) одинакового формата, созданные/обработанные одинаковой версией ПО, например, для работы ускорителя определенного периода (с одинаковой энергией и светимостью). Эта модель была применена для организации данных эксперимента ATLAS.

По модели MONARC все классы данных имели определенное количество копий и распределялись между центрами грид согласно меморандуму о взаимопонимании (MoU), заключенному всеми ВЦ в рамках WLCG. При этом дисковый ресурс центров уровня T2 не мог быть использован для длительного (месяцы) или постоянного хранения данных. Такая концепция привела к тому, что к середине 2012 г. дисковое пространство было заполнено данными формата ESD в предположении, что именно эти данные будут наиболее востребованы. Диски центров уровня T1 были переполнены при значительном свободном дисковом пространстве в центрах уровня T2.

Классификация наборов данных и их номенклатура были выполнены в рамках эксперимента ATLAS. Данная работа [45] до сих пор является основным документом, в котором описывается номенклатура данных эксперимента и сформулированы базовые определения классов данных. После введения классификации данных необходимо определить их значимость и популярность. Так, «сырые» события составляют отдельную группу и являются наи-

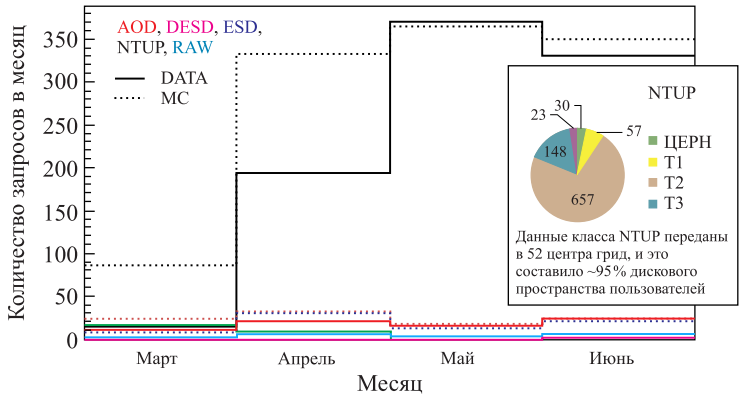

Рис. 15. Количество запросов пользователей на создание дополнительных копий наборов данных в распределенной инфраструктуре

более ценными, их утрата по любой из причин не может быть восполнена. Поэтому для «сырых» данных была выбрана следующая модель: полная копия «сырых» данных архивируется в ЦЕРН, а вторая копия распределяется и архивируется между центрами уровня T1 (таким образом, всегда существуют две копии «сырых» данных). Одна копия «сырых» данных для последнего периода работы ускорителя (как правило, 2 мес.) распределена между центрами T1 и T2 (выбор центров основан на стабильности их работы и имеющемся дисковом ресурсе) и используется для изучения работы детектора и/или экспресс-обработки. Для остальных классов данных было введено понятие популярности данных. Для этого были классифицированы методы доступа к данным: копирование за пределы распределенной системы обработки, чтение информации программами анализа индивидуальных ученых (при этом учитывались количество запросов, география и количество индивидуальных ученых), доступ к данным для отдельных физических групп. Центральная обработка и моделирование данных не учитывались как доступ, но эта информация использовалась на следующем этапе для определения «температуры» данных (на рис. 15 показано количество запросов пользователей на дополнительные копии данных). Введена система весов, когда общее количество запросов нормировалось на количество ученых, требовавших доступ к данным.

Таким образом, в результате анализа информации относительно популярности всех классов данных для реальных и моделированных событий было определено, что наиболее популярными являются данные классов AOD и NTUP, а не ESD, и именно для них необходимо наибольшее количество копий с учетом географического распределения пользователей. На рис. 16 показано количество обращений задач анализа данных к различным классам данных в центрах уровней T1 и T2 [49]. Из графика видно, что форматы

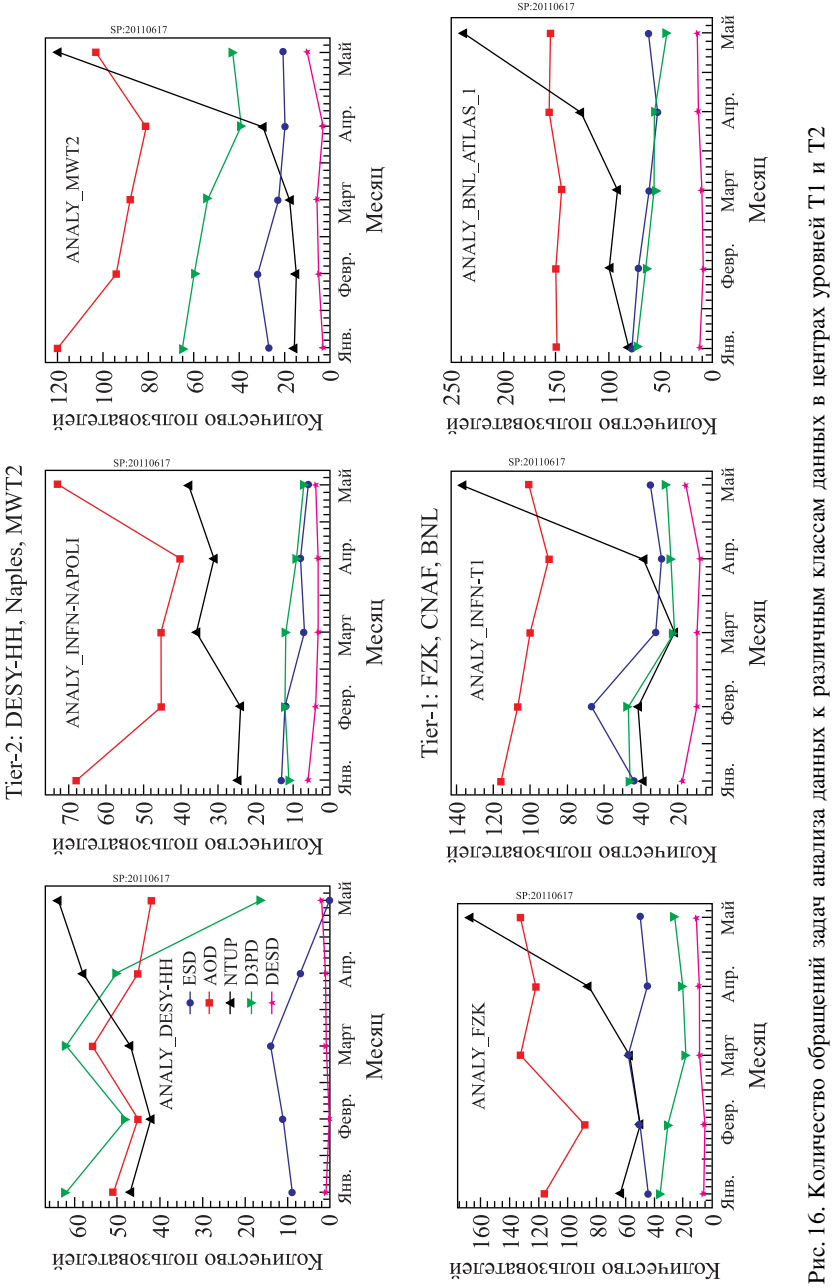

Рис. 16. Количество обращений задач анализа данных к различным классам данных в центрах уровней T1 и T2

AOD и NTUP являются наиболее популярными для данного класса задач. Одновременно достаточно одной копии данных формата ESD, так как эти данные используются только группами, централизованно изучающими работу детектора. Разработанная методика определения популярности данных сохранилась до сих пор, в 2016–2017 гг. к методам вероятностного анализа [46] были добавлены методы «машинного обучения» для определения популярности данных в системах управления данными и управления потоком загрузки [47]. Следующим этапом стала работа по анализу популярности данных внутри класса, для этого была предложена термодинамическая модель данных, которая будет рассмотрена в п. 1.4.2.

1.4.3. Термодинамическая модель данных. Целью создания модели явились потребности экспериментов на LHC определить методику управления данными (распределение наборов данных между центрами грид, удаление данных, архивирование «непопулярных» данных). Реализация методики привела к созданию автоматической системы управления данными для экспериментов в области ФВЭ, использующих грид-инфраструктуру.

В модели под набором данных подразумеваются как моделированные данные, так и полученные с экспериментальной установки. Единицей управления данными является датасет, как это было определено ранее.

Базовые предположения сформулированы следующим образом.

• Детальная классификация типов данных: физические эксперименты имеют различные категории/классы данных — «сырые» данные, получаемые непосредственно с экспериментальных установок (RAW); события, моделируемые методом Монте-Карло (RDO); оцифрованные события RDO, называемые «срабатываниями» (HITS); реконструированные данные (ESD); производные данные (AOD, DPD, NTUP, TAG), используемые для физического анализа.

• Каждый класс данных может иметь различное время «жизни» (от «хранить вечно» для данных класса RAW до нескольких месяцев для данных класса ESD).

• Метод управления данными зависит от их класса.

• Все дисковое пространство, принадлежащее эксперименту в рамках грид-инфраструктуры (для экспериментов на LHC такой инфраструктурой является WLCG), рассматривается как единый ресурс (это стало принципиальным отличием от модели MONARC, в которой при иерархии центров ресурсы уровней T0, T1 и T2 рассматривались изолированно).

• Вводится понятие «центр хранения данных», таким центром может быть центр уровня T0, T1 или T2 при условии его стабильной работы и наличия ресурса WAN для скоростной передачи данных:

— скоростная передача данных была определена необходимостью подключения центра к выделенной сети LHCOPN или LHCONE, но не ограничена ими;

— «стабильная работа» определялась согласно результатам системы учета обращений к данным, принятой в грид-инфраструктуре, с допустимым отклонением от параметров работы, описанным в меморандуме о взаимопонимании, не более чем на 1 %, а также постоянным мониторированием стабильности работы WAN для каждого центра с использованием системы perfSONAR [48]. Информация о стабильности центра автоматически обновлялась в информационной системе каждые 12 ч;

— расширение функций центров второго уровня (T2) было принципиальным изменением концепции MONARC и первой попыткой начать эволюционный переход от иерархической модели к «смешанной компьютерной модели» грид, кроме того, большую роль в использовании ВЦ начинала играть стабильность и производительность WAN; социологически это стало очень важным шагом по привлечению руководства центров уровня T2 к работе над новой компьютерной моделью для LHC. При таком подходе роль наиболее стабильных центров уровня T2 возрастала и их участие становилось более заметным.

• Данные класса RAW распределяются для архивирования на основе меморандума о взаимопонимании, в котором определены параметры каждого центра уровня T1 в рамках WLCG, полная версия данных хранится в ЦЕРН.

• Первичная копия приведенных данных (классы ESD, AOD...) всегда хранится в центре, где она была произведена.

• Удалению первичной копии данных должно предшествовать архивирование этой копии (за исключением централизованно удаляемых наборов данных в случае обнаружения ошибки при их создании или уточнения калибровочных констант).

• Удаление копий данных всегда осуществляется централизованно, действия протоколируются, протокол управления данными хранится в течение всего времени «жизни» эксперимента.

• «Важность» и «популярность» данных не являются синонимами:

— «важность» данных определяется физической программой эксперимента;

— «популярность» данных «измеряется» на уровне набора данных (датасет) по количеству обращений к данным с использованием средств обработки, анализа, репликации и т. д. и с учетом частоты обращения, географии и временного интервала.

Введена температурная шкала для всех данных. Согласно показаниям шкалы состояние (температура T) данных может быть: «горячей», «теплой», «холодной», «замороженной», «устаревшей» (слово «устаревшие» было выбрано по этическим соображениям, чтобы не называть значение температуры данных «мертвой»).

«Горячие» данные широко используются учеными и физическими группами эксперимента для проведения физического анализа и исследования ра-

боты детектора. К таким данным всегда относятся: данные класса RAW для последнего периода набора данных (2–3 мес.) и приведенные данные последнего года. Для «горячих» приведенных данных всегда существует несколько копий. Копии размещаются в грид-инфраструктуре согласно географии доступа к данным (например, все копии не могут быть размещены на одном континенте). Первичная реплика «горячих» данных не подлежит удалению. Дополнительные копии «горячих» данных создаются автоматически по мере необходимости (этот метод будет рассмотрен в п. 1.4.3).

«Теплые» данные используются отдельными физическими группами и учеными для физического анализа. Предполагается, что уже имеется достаточное количество копий этих данных в грид-инфраструктуре. Дополнительные копии можно запросить через интерфейс запроса передачи данных, запросы будут одобрены согласно стандартной процедуре утверждения передачи данных. Для «теплых» данных гарантировались, по крайней мере, две копии. Температура «теплых» данных (T) всегда может быть повышена, количество копий может быть увеличено автоматически. При уменьшении количества копий для «теплых» данных первоначально удаляются данные на уровне T1 и в ВЦ, где отмечена наименьшая частота доступа к ним. Копии «теплых» данных удаляются в случае запроса на свободное дисковое пространство, например, перед «началом» переобработки данных.

«Холодные» данные используются отдельными физиками или рабочими группами. Полная копия «холодных» данных распределена между центрами грид-инфраструктуры, дополнительные копии могут быть созданы только на дисках, принадлежащих отдельным группам/пользователям, но не могут быть размещены на дисках, предназначенных для всего эксперимента. Температура «холодных» данных всегда может быть повышена, при этом количество копий увеличивается автоматически. Массовое удаление данных (уменьшение количества копий) внутри инфраструктуры автоматически начинается при понижении T с «горячей/теплой» до «холодной».

«Замороженные» данные практически не используются. Сохраняется одна полная копия (в центре, где данные были произведены или повторно обработаны). Это может быть копия на диске или магнитной ленте. Дополнительные копии «замороженных» данных могут быть созданы только на дисках, принадлежащих отдельным группам/пользователям, но не могут быть размещены на дисках, предназначенных для всего эксперимента. Данные класса RAW являются исключительным случаем, для них всегда сохраняются две архивируемые копии (причем одна из них находится в ЦЕРН, а вторая — в центрах уровня T1).

«Устаревшие» данные не могут быть использованы для физического анализа или других исследований в эксперименте, они подлежат удалению со всех сайтов и из всех каталогов. Примером «устаревших» данных являются приведенные данные, произведенные при ошибке, найденной в ПО, или с

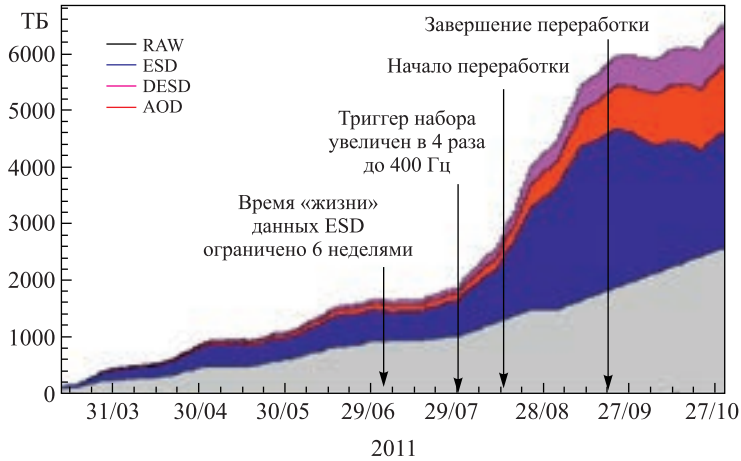

Рис. 17. Использование дискового ресурса ATLAS в 2011 г. до и после введения термодинамической модели

неправильными калибровками. Данные класса RAW не могут быть признаны «устаревшими».

Следующим этапом стала разработка концепции динамического увеличения количества копий датасетов в зависимости от популярности данных. Реализован механизм автоматического контроля количества копий при изменении количества запросов. Количество запросов анализировалось ежедневно, количество копий рассчитывалось каждые три дня. Оба интервала были выбраны исходя из оценки среднего времени ожидания заданий анализа данных и времени их выполнения. Этот метод получил название «динамическое распределение данных». В качестве места для новой копии выбирался сайт со свободным дисковым пространством и вычислительным ресурсом из числа центров хранения данных. Такой подход также привел к конкуренции между грид-центрами WLCG, так как теперь уровень загрузки был напрямую связан со стабильной работой центра.

На рис. 17 показано, как введение термодинамической модели позволило оптимально использовать дисковый ресурс и увеличить скорость набора данных в 4 раза (до 400 Гц), а также провести переобработку всех данных. Это было достигнуто исключительно за счет введения понятия «популярность» данных и оптимального использования дискового пространства грид-центров.

1.4.4. Методика определения стабильности работы центров WLCG при создании «смешанной компьютерной модели» грид-инфраструктуры. Переход к «смешанной компьютерной модели» для экспериментов на Большом адронном коллайдере. Одним из основных ограничений модели MONARC явилось предопределение функций центров грид на основе их клас-

сификации по уровням T0, T1, T2 и T3. Вместе с тем оказалось, что многие центры уровня T2 имеют бóльшие мощности и часть центров по стабильности работы сравнима (или даже превосходит) центры уровня T1. Для перехода к «смешанной компьютерной модели» необходимо было выделить наиболее мощные и стабильные центры и использовать их ресурс оптимальным образом. Введены три независимые метрики (стабильность центра при обработке и анализе данных, стабильность центра при обмене данными с другими центрами, пропускная способность центра) и четыре градации для классификации центров: альфа, бета, чарли, дельта (от англ. A, B, C, D). На первом этапе все центры уровня T2 были причислены к группе альфа.

Стабильность центров при выполнении заданий обработки, анализа и моделирования. Определены тестовые образцы заданий трех типов (это было сделано совместно с физическими группами, чтобы задания соответствовали реальным заданиям и имели аналогичные требования по времени обработки событий, оперативной памяти и вводу/выводу информации).

Для каждого типа заданий были подготовлены тестовые наборы данных, а их копии помещены во все центры уровня T2. Для всех типов заданий были определены необходимые версии ПО (например, версия программы реконструкции событий). Версии ПО были доступны в каждом из центров. Ежедневно каждый центр получал запрос на выполнение тестовых заданий, при этом тип задания выбирался случайным образом. Результаты выполнения тестовых заданий хранились в базе данных, и статистика была доступна как компьютерным специалистам центров, так и представителям экспериментов на LHC.

Стабильность центра при обмене данными с другими центрами. Были подготовлены тестовые наборы данных. Эти наборы были распределены между всеми центрами уровня T2 и отличались размерами файлов, которые соответствовали создаваемым при обработке, анализе и моделировании файлам (в первом приближении их можно разделить по размеру на «большие» (2,5 ГБ), «средние» (1 ГБ) и «малые» (0,5 ГБ)). Четыре раза в день генерировался автоматический запрос на передачу данных между всеми центрами уровня T2, а также между всеми центрами уровня T2 к центрам уровня T1. По результатам тестовых передач создавалась матрица «все против всех», и по ней определялись центры уровня T2, имеющие стабильную передачу данных со всеми центрами уровня T1. При наличии 100 %-го выполнения заданий, описанных в п. 1.4.2, эти центры используются также как центры уровня T1, за исключением архивирования данных (такие центры получили название T2D). Созданная матрица также используется системой управления загрузкой при определении наилучшей комбинации центров для обработки данных.

Пропускная способность центра и стабильность его подключения к WAN. Как было изложено ранее, модель MONARC была статической и в качестве

вычислительного ресурса в ней учитывались только вычислительные мощности центров WLCG и их дисковый ресурс. При переходе к «смешанной компьютерной модели» необходимо было учитывать также ресурс WAN. Для этого инфраструктура каждого центра была дополнена системой мониторирования perfSONAR, которая позволяла постоянно мониторировать состояние WAN при передаче данных. Для всех пар сайтов (пункт отправки/пункт назначения) собирается следующая информация:

— количество файлов, переданных за последний час;

— количество файлов в очереди на передачу;

— средняя пропускная способность согласно метрикам FTS (File Transfer System) для последнего часа, дня и недели;

— информация от системы perfSONAR: задержки при передаче, число потерянных пакетов, пропускная способность.

Эта информация собирается специальным сервисом NWS (Network Weather Service) и хранится в специальном хранилище, агрегированная информация хранится в информационной системе. ИС запрашивает информацию от NWS каждые 15 мин. В разд. 3 будет рассмотрено, как информация о WAN используется для выбора вычислительного ресурса при выполнении заданий обработки, анализа или моделирования данных. На рис. 18 представлена матрица, показывающая стабильность работы центров первого уровня при передаче данных между собой, а на рис. 19 — результаты передачи данных в центры, отобранные в результате созданной методики для «постоянного» хранения данных.

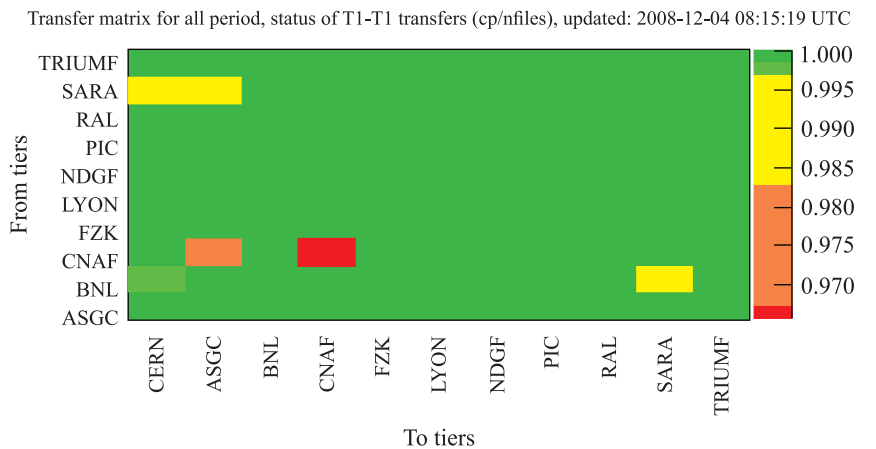

Рис. 18. Матрица эффективности работы центров уровня T1 при передаче данных

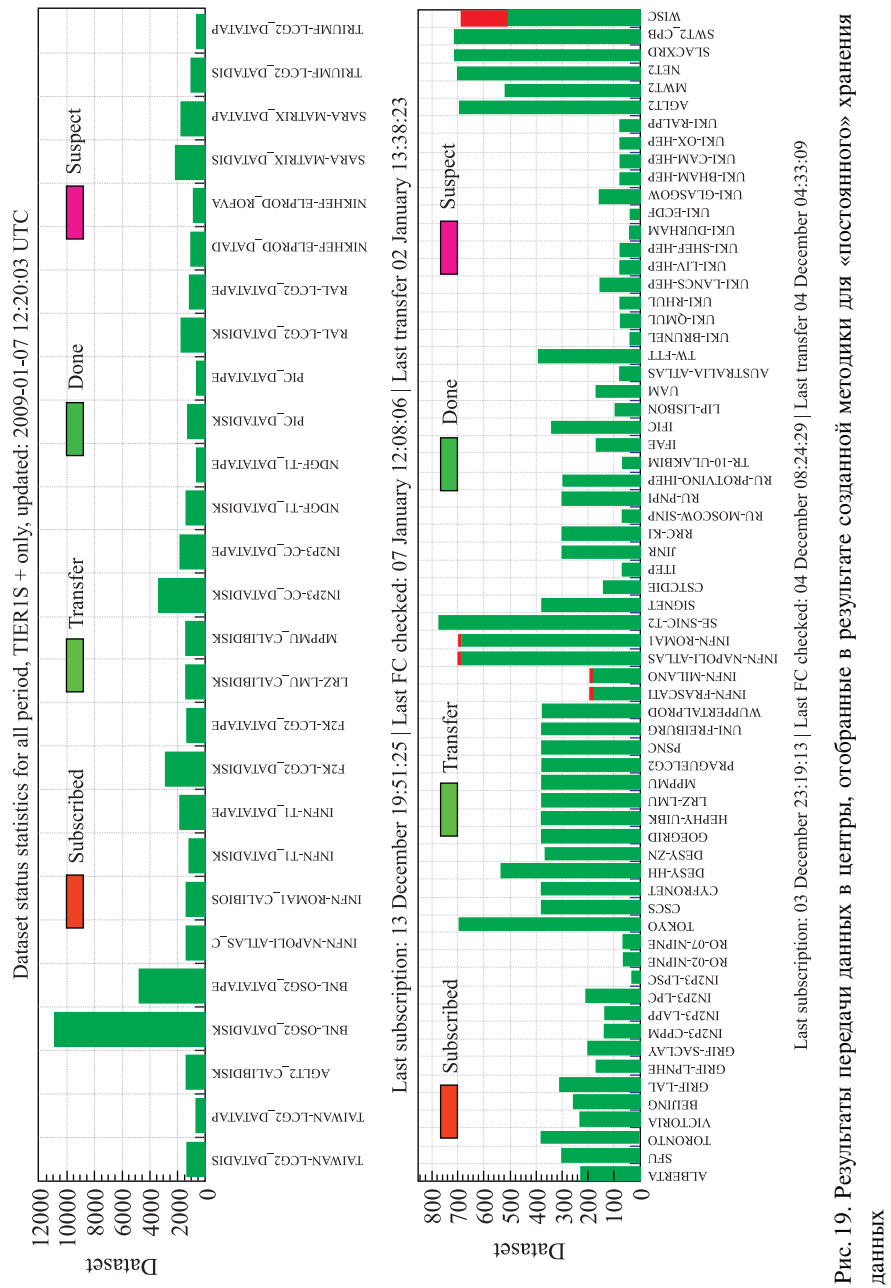

Рис. 19. Результаты передачи данных в центры, отобранные в результате созданной методики для «постоянного» хранения данных

Рис. 20. Результаты тестирования глобальной сети между центрами грид для двух типов тестов: perfsonar и тестов передачи данных ATLAS

На рис. 20 показаны результаты тестирования глобальной сети между центрами грид для двух типов тестов: perfsonar и тестов передачи данных ATLAS. На рис. 21 показано, как изменилось количество центров второго уровня, используемых для хранения и обработки данных на тех же «правах», что и центры первого уровня, после применения методики определения стабильности центров. Как отмечалось ранее, большинству центров T2 это дало шанс существенно расширить свои функциональные возможности и предоставить гораздо бóльшие ресурсы для «виртуальной организации». Многие центры (например, ОИЯИ, ИФВЭ НИЦ КИ) предоставляют свои ресурсы более чем одной коллаборации, и изменение режима их использования экспериментом ATLAS привело к изменению статуса центров в экспериментах ALICE и CMS. Кроме того, введение параметра WAN привело к необходимости для центров учитывать пропускную способность глобальной вычислительной сети и, в конечном итоге, созданию WAN, известной ныне как LHCONE. На рис. 21 показано, как изменилось количество центров, используемых для передачи данных «все-ко-всем» (центры T2D), и как увеличилось число «стабильных центров» за 6 мес. после использования методики опре-

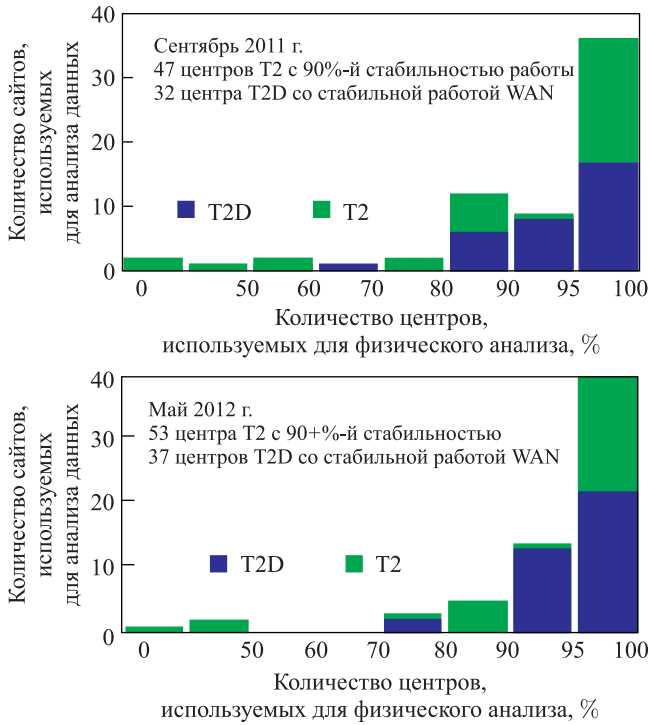

Рис. 21. Количество центров уровня T2, используемых для хранения данных

деления стабильности центров. В настоящее время проверка сайтов согласно описанной методике полностью автоматизирована и реализована через автоматическую систему проверки ВЦ под названием HammerCloud [50].

Следует отметить, что переход к «смешанной компьютерной модели» потребовал разработки концепции и создания новой информационной системы хранения информации [51]. Концепция такой системы и ее архитектура были изначально предложены и реализованы для эксперимента ATLAS. Система получила название AGIS (ATLAS Grid Information System) и позволила аккумулировать информацию о центрах WLCG, а также дополнить ее информацией, собираемой в результате проверки стабильности работы центров.

Разработанная концепция ИС оказалась гибкой и динамичной, что позволило на следующем этапе создания модели компьютинга дополнить ее информацией о суперкомпьютерных центрах и центрах облачных вычислений. Дальнейшее развитие ИС и ее использование в эксперименте CMS привели к разработке ее второго поколения, известной как CRIC (Computing Resource Information Catalog) [52].

Таким образом, на первом этапе развития компьютерной модели был совершен переход от иерархической модели к «смешанной компьютерной модели» (рис. 22), были «нарушены» иерархия и предопределение функций центров грид, предложенных в модели MONARC.

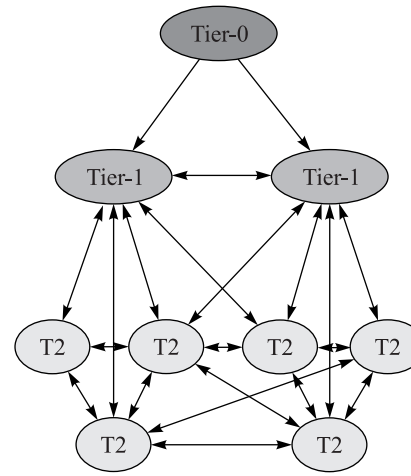

Рис. 22. «Смешанная компьютерная модель» для экспериментов LHC

Были созданы необходимые программные средства (система запросов на передачу данных, первая версия принципиально новой информационной системы), исследована работа и роль WAN при создании вычислительной инфраструктуры, разработана и реализована методика определения стабильной работы центров WLCG, а также методика определения популярности данных и метод динамического распределения данных между центрами WLCG. Все это создало предпосылки к разработке новой компьютерной модели и созданию гетерогенной компьютерной киберинфраструктуры для следующего этапа работы LHC и нового поколения экспериментов в области физики частиц.

1.4.5. Метод динамического распределения данных с использованием информации о популярности данных. Слоган, описывающий модель MONARC, — «задачи идут к данным» (т. е. задачи анализа, обработки, мо-

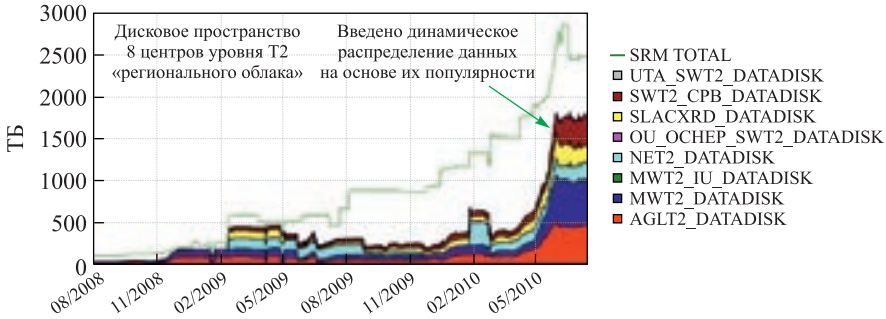

Рис. 23. Рост объема данных после начала работы LHC

делирования выполняются в центрах, где находятся исходные данные). При таком подходе не только создавались задержки с выполнением задач, но и искусственно увеличивалось количество копий данных. На рис. 23 показано, как после начала работы коллайдера резко возросло количество данных на сайтах T2 за счет многочисленных копий.

Подробно ограничения модели MONARC рассмотрены в п. 1.3, здесь покажем, как переход к «смешанной компьютерной модели» и введение термодинамической модели и методики определения наиболее стабильных центров уровня T2 позволили более эффективно использовать ресурсы WLCG. На первом этапе были определены наиболее популярные форматы данных для задач физического анализа, ими оказались данные форматов AOD и NTUP. Задачей второго этапа стала разработка метода динамического увеличения копий наиболее популярных наборов данных, а также использование дискового пространства центров второго уровня для кэширования дополнительных копий.

Рассмотрим более подробно предложенный алгоритм.

• Статическое распределение данных для центров уровня T2 прекращается.

• Если задача пользователя обращается к набору данных (датасет) и нет копии датасета в центрах уровня T2, то первая такая задача выполняется в центре первого уровня (T1), где всегда есть копия данных, одновременно автоматически посылается запрос в систему управления данными для создания дополнительной копии набора данных на одном из центров уровня T2.

• Таким образом, для первого обращения к данным не существовало задержки, связанной с выполнением задачи пользователя, а время передачи данных и создания дополнительной копии датасета занимало несколько часов. Метод выбора T2 был основан на использовании нескольких метрик (методика управления потоками заданий и работы брокера задач подробно рассмотрены в разд. 3):

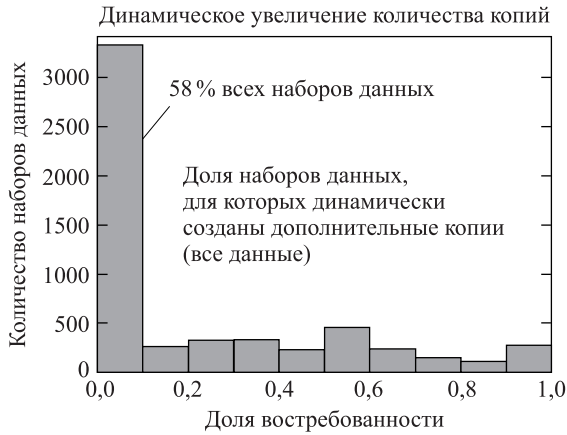

Рис. 24. Популярность данных в эксперименте ATLAS, частота использования наборов данных

— размер свободного дискового пространства;

— количество задач в очереди на выполнение к данному сайту;

— планируемое время остановки сайта;

— количество файлов в очереди на передачу к данному сайту (общее количество копий данных основывается на статистике обращений к ним задач пользователей и увеличивается логарифмически по мере роста количества обращений, т. е. 10, 100, 1000 ... обращений соответствуют $1, 2, 3 \ldots$ дополнительным копиям данных).

• Данные в форматах, не используемых для физического анализа, за исключением формата EVNT (входные события для этапа моделирования), например датасеты в формате RAW, исключаются из рассмотрения.

На рис. 24 показано, насколько эффективно используются наборы данных и их дополнительные копии. Из графика видно, что в 58 % случаев было достаточно только одной основной копии. Доля востребованности набора данных вычислялась как

$$accessFraction = dynamicReplicaAccess/totalDatasetAccess,$$

где *dynamicReplicaAccess* — количество задач анализа, обратившихся к дополнительному набору данных; *totalDatasetAccess* — общее количество обращений задач анализа к наборам данных.

На рис. 25 показано, какие классы данных наиболее популярны для динамического увеличения количества копий, а также продемонстрированы популярность копий, повторное (и более) использование копий. Графики подтверждают правильность выбранной методики, по ним видна популярность данных форматов NTUP и AOD, используемых для физического анализа.

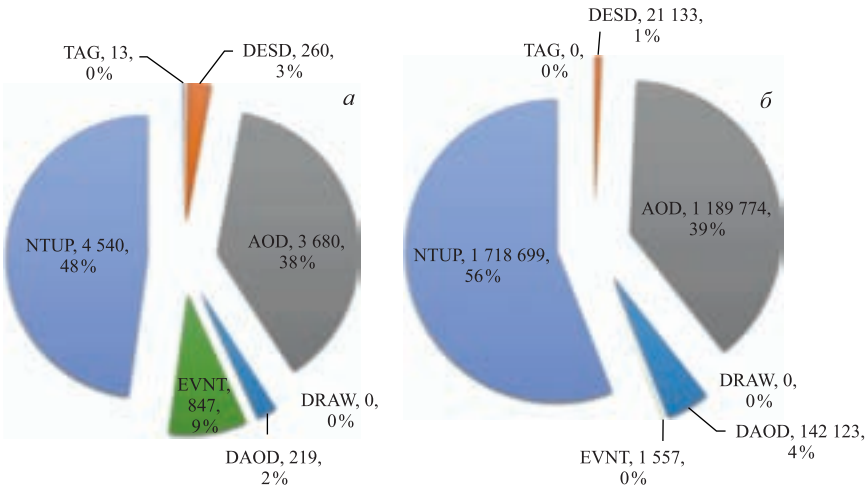

Рис. 25. Выбор данных для динамического увеличения наборов, популярность второй и последующих копий наборов данных в зависимости от класса данных: *а*) количество наборов данных, для которых динамически было увеличено количество копий; *б*) использование более двух раз дополнительной копии для разных классов данных

Важной особенностью является введение понятия «кэширование данных», а также реализация термодинамической модели и понимание, что копии данных на дисках должны иметь время «жизни», а по истечении интереса к набору данных датасет должен автоматически архивироваться и «мигрировать» на ленту. На рис. 26 показано, как популярность данных менялась со временем. Из графика видно, что популярность данных резко уменьшается примерно через 45 дней. Это привело к созданию специального сервиса в системе управления данных (deletion service) [55] для удаления копий датасетов, не востребованных в физическом анализе. Таким образом, был сделан важный шаг в развитии методики управления данными и переход от заранее планируемого распределения данных на сайтах грид — к динамическому распределению данных.

Следует отметить, что реализация данного метода не могла быть отложена до времени плановой остановки коллайдера и она проводилась непосредственно в период набора и обработки данных. Переход к динамической системе распределения данных позволил использовать слоган «данные и задачи анализа идут туда, где есть свободные ресурсы».

Таким образом, фундаментальными ограничениями иерархической модели обработки данных являлись:

— иерархия ВЦ и статический характер связки 1:T1–n:T2, когда любой сбой в работе центра первого уровня (T1) практически останавливал работу

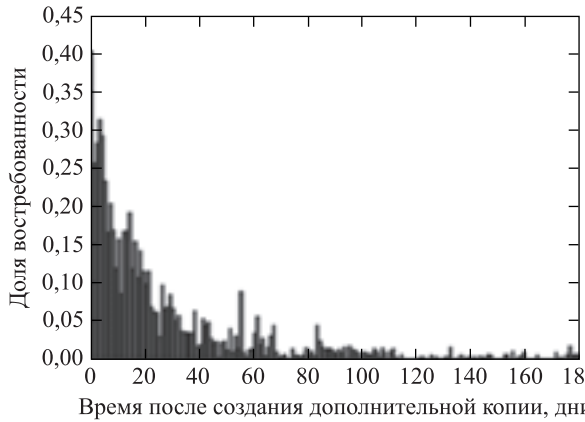

Рис. 26. Изменение популярности набора данных в зависимости от времени

всех связанных с ним центров второго уровня (T2), в результате эксперименты лишались мощностей от 1 до 10 центров одновременно;

— недооценка роли глобальной вычислительной сети (WAN).

Итак, в разд. 1 для разработки концепции и новой компьютерной модели распределенной обработки данных необходимо было обосновать переход от иерархической компьютерной модели к «смешанной компьютерной модели». Это стало возможным после разработки методик определения популярности классов и наборов научных данных. На основе этих методик был предложен и реализован метод динамического распределения данных между центрами грид-инфраструктуры. Кроме того, был обоснован вывод о необходимости учитывать ресурс WAN наряду с дисковым и компьютерным ресурсами при оценке возможностей центров обработки физических данных. В результате это позволило отказаться от модели MONARC и гибко использовать ресурс всех центров WLCG для хранения и обработки данных на последующих этапах работы LHC.

2. ТРЕБОВАНИЯ К ВЫЧИСЛИТЕЛЬНОЙ ИНФРАСТРУКТУРЕ ДЛЯ ОБРАБОТКИ, МОДЕЛИРОВАНИЯ И АНАЛИЗА ДАННЫХ. РОЛЬ СУПЕРКОМПЬЮТЕРОВ ДЛЯ ПРИЛОЖЕНИЙ В ОБЛАСТИ ФИЗИКИ ВЫСОКИХ ЭНЕРГИЙ И ЯДЕРНОЙ ФИЗИКИ

В данном разделе рассматриваются требования к вычислительной инфраструктуре на втором и последующих этапах работы Большого адронного коллайдера. Обосновывается необходимость развития компьютерной модели в области физики частиц в целом и для экспериментов на LHC в частно-

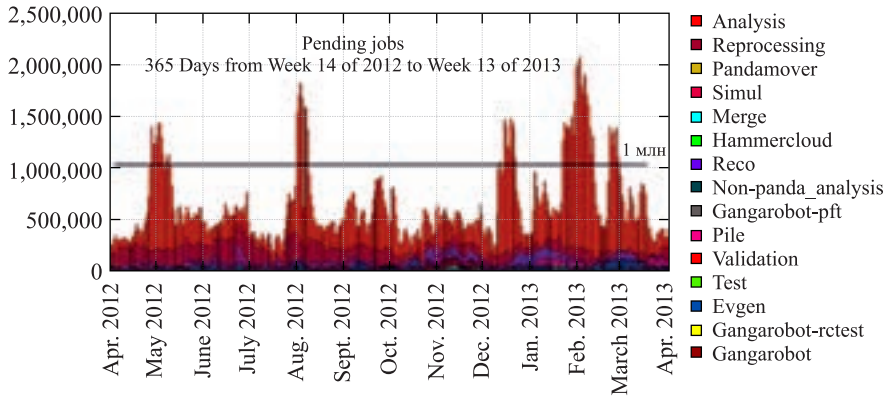

Рис. 27. Количество задач в очереди на выполнение из-за отсутствия вычислительного грид-ресурса в 2012–2013 гг. (статистика LHC Run1)

сти. Обоснован переход от однородной вычислительной среды грид к гетерогенной вычислительной среде, включающей ресурсы облачных вычислений, суперкомпьютеры и грид-инфраструктуру. Рассматриваются вопросы использования суперкомпьютеров для приложений в области ФВЭ и ЯФ и интеграции суперкомпьютеров с системой высокопропускной обработки данных (грид).

К началу второго этапа работы LHC (2014–2018 гг.) стало очевидно, что имевшийся компьютерный ресурс использован полностью. На рис. 27 показано количество задач, ожидавших выполнения из-за отсутствия грид-ресурсов. Хорошо видно, что в случае пиковых нагрузок, как правило, предшествовавших основным научным конференциям и этапам массовой переобработки данных, очередь могла достигать 1,5 млн задач. Беспрецедентная производительность LHC на втором этапе его работы и увеличение объемов данных потребовали компьютерных мощностей, бóльших, чем мог предоставить консорциум WLCG (на рис. 28 приведены графики интегральной светимости в 2011–2018 гг.). Уже в 2016 г. светимость была на 60 % больше запланированной, а 2018 г. стал рекордным по объемам полученных данных.

Возрастающая «множественность», связанная с ростом энергии пучков и светимости ускорителя, ведет к увеличению размера события и времени, необходимого для его обработки. Возрастание необходимого вычислительного и дискового ресурса для эксперимента ATLAS (в том числе на этапе супер-LHC — см. на рис. 29 и 30).

В силу финансовых и технических причин невозможно увеличить вычислительный и дисковый ресурсы центров грид в несколько раз или создать новые центры. Финансирующие организации согласились в лучшем случае сохранить существующий фонд финансирования ИТ для экспериментов на

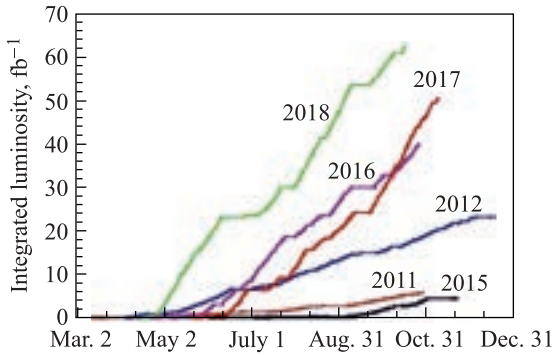

Рис. 28. Интегрированная светимость LHC в 2011–2018 гг.

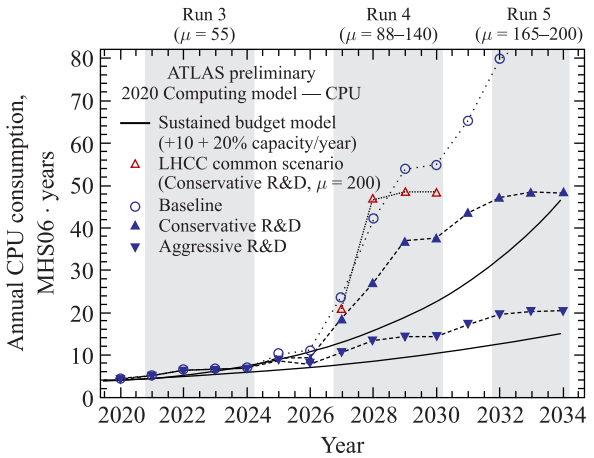

Рис. 29. Потребности в вычислительном ресурсе эксперимента ATLAS на третьем и четвертом этапах работы LHC для различных сценариев обработки и моделирования данных

LHC. Технически на создание новых центров и их интеграцию в общую инфраструктуру потребовались бы годы. Кроме того, использование гомогенной киберинфраструктуры не выглядело больше как единственное правильное решение.

Конечно, существует возможность консервации компьютерной модели (только грид-инфраструктура) в пределах доступного финансирования. Ценой такого решения будет уменьшение количества набираемых данных и невозможность обрабатывать данные в течение года, а в результате — замедление в получении фундаментальных знаний об окружающем нас мире. Кроме того, появился ресурс, которого не было в начале XXI в., а именно ресурс, пре-

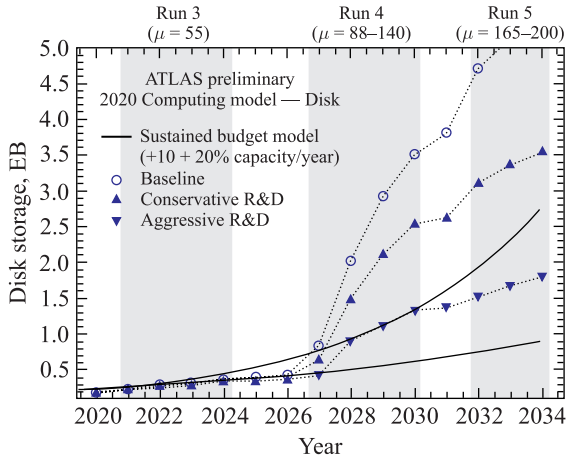

Рис. 30. Потребности в дисковом ресурсе эксперимента ATLAS на третьем и четвертом этапах работы LHC для различных сценариев обработки и моделирования данных

доставляемый крупными коммерческими ИТ-компаниями (Google, Amazon, Yandex). Рассмотрим, как эти факторы повлияли на эволюцию компьютерной модели в области физики частиц.

Первый руководитель и организатор проекта WLCG д-р Лес Робертсон в своем докладе на конференции по компьютингу в области ФВЭ и ЯФ в Праге [54] задал нериторический вопрос: «Нами построен грид. Что дальше?» Он также постулировал, что вызовы, стоящие перед научным сообществом, требуют изменений в концепции обработки и анализа данных. В частности, д-р Робертсон сказал, что, пока строили грид, компьютерный ландшафт изменился и надо быть готовым адаптироваться к новым реалиям. В некотором смысле это выступление стало его напутствием, так как в следующем году он покинул пост руководителя WLCG.

Работы по развитию новой концепции и методов обработки данных были начаты в ЦЕРН, университетах и национальных лабораториях в США (BNL, Фермилабе, Ратгерском и Техасском университетах), Европе (Университете Осло, DESY), России (ОИЯИ и НИЦ «Курчатовский институт»). Действительно, с момента начала проектов EGEE и globus ландшафт компьютерных ресурсов к началу второго десятилетия XXI в. изменился. Он стал огромен и разнообразен, сложен и разнороден, способен к решению многих задач и скорее выглядит как большой архипелаг, чем как группа материков. Наряду с центрами коллективного пользования, которые подразумевают использование ресурса группами ученых, часто работающих в разных областях наук, узковедомственными и/или узкоспециализированными ВЦ, существуют суперкомпьютеры, грид-консорциумы, коммерческие и академические ресурсы

облачных вычислений, университетские вычислительные кластеры. Такова тенденция организации вычислительных мощностей (киберинфраструктуры) во всем мире. В настоящее время распределенная киберинфраструктура и ее составляющие используются в лучшем случае индивидуально, а чаще — как изолированные ресурсы. Аристотель утверждал, что «целое больше, чем сумма его частей» [55], поэтому федеративная организация распределенной киберинфраструктуры и создание «озер научных данных» позволят использовать компьютерные ресурсы более эффективно, что будет выгодно как «владельцам» ресурса, так и пользователям. Проблема интеграции разнородных ресурсов и создание единой федеративной киберинфраструктуры стали основополагающей идеей при разработке новой компьютерной модели для экспериментов в области ФВЭ и ЯФ.

2.1. Общие проблемы создания федеративной киберинфраструктуры. Существующие проблемы создания федеративной распределенной киберинфраструктуры (ФРКИ) характеризуются:

— недостатком блоков, из которых можно построить такую федерацию;

— точечными решениями, не выходящими за пределы немедленного использования и конкретной задачи (и/или центра), как правило, это прикладные решения с низким уровнем абстракции для интерфейсов и модулей программного обеспечения;

— тем, что вопросы интеграции распределенной киберинфраструктуры рассматриваются только после создания вычислительной инфраструктуры, а не на этапе разработки архитектуры системы;

— использованием характеристик дискового и вычислительного ресурса как параметров, определяющих мощность вычислительного комплекса, без учета мощности глобальных компьютерных сетей, скорости передачи данных и возможного использования удаленного доступа к ним (например, при создании грид-инфраструктуры и ПО для экспериментов на LHC не была предусмотрена возможность использовать ресурс университетских кластеров, суперкомпьютерных центров и центров облачных вычислений, а также удаленный доступ к данным).

Таким образом, требуется решить все четыре проблемы и обеспечить доступ к информации о параметрах ФРКИ пакетам управления данными и загрузкой системы в динамическом режиме.

При исследовании ФРКИ и создании архитектуры необходимо следовать следующим принципиальным подходам:

— единому методу и уровню абстракции управления ресурсами;

— общей системе управления загрузкой и данными в гетерогенной компьютерной среде;

— интегрируемым и развиваемым средствам для управления программным обеспечением ФРКИ.

Все вместе это составляет проблему фундаментального характера. Отсутствие ее адекватного решения в настоящее время приводит к экономическим и функциональным потерям. Логика развития компьютерной модели экспериментов на LHC, в частности для эксперимента ATLAS, привела к выводу о необходимости создания системы для обработки данных в гетерогенной среде, федеративного устройства вычислительных ресурсов и системы управления данными и загрузкой в распределенной вычислительной среде. И хотя начальная мотивация была связана с экспериментами на LHC, а также с будущими мегапроектами, такими как NICA, FAIR и XFEL, но как количественные, так и качественные требования не являются специфическими для физики, они типичны для научных приложений в таких областях, которые требуют хранения, анализа, обработки и управления данными в мультипетабайтном и эксабайтном диапазонах.

2.2. Вопросы конвергенции высокопропускных и высокопроизводительных вычислений. Роль приложений в области физики высоких энергий и ядерной физики для суперкомпьютеров. Рассмотрим, какое место занимает высокопропускной компьютинг (HTC — High Throughput Computing) и его использование на основе исследований, проведенных сотрудниками BNL, ORNL, ОИЯИ, НИЦ КИ, Ратгерского университета и UTA [56]. Научные задачи ФВЭ и ЯФ являются прекрасным примером приложений HTC.

Высокопропускные вычисления имеют следующие характеристики:

— рабочее (выполняемое) задание состоит из нескольких задач;

— каждое из заданий является частью одной кампании (запроса, цепочки заданий, как будет показано далее в разд. 3); типичным примером запроса является моделирование физических процессов методом Монте-Карло (см. рис. 13), состоящее из цепочки последовательных заданий: генерации событий, оцифровки, моделирования, реконструкции, создания объектов, используемых для физического анализа;

— количество выполненных заданий (это основная характеристика).

Одновременно HTC может характеризоваться временны́м параметром (временем выполнения заданий) и иметь дополнительный параметр (назовем его параллелизм — это количество одновременно выполняемых задач в предоставленной вычислительной среде, например количество задач виртуальной организации, выполняемых ежедневно в среде WLCG. В HTC практически не используются параллельно выполняемые (и связанные между собой) задачи.

Работы по использованию суперкомпьютеров для приложений HTC, например для вычислительной химии с попыткой ввести параллелизм на уровне задач, представленных в [57], для задач $O(10^3)$, были не очень успешны, что привело к решению выполнять задачи последовательно на суперкомпьютере, но важным посланием является понимание важности модели выполнения заданий и роли их планировщика, определяющего, какие задания где и когда выполняются.

Реализация такого подхода потребовала изменений и введения абстракции на нескольких уровнях модели обработки данных. Основными решениями явились введение понятия пилотных заданий (фундаментальное понятие для грид-НТС, поздняя «привязка» ресурсов к выполняемым заданиям) и создание нового поколения системы управления загрузкой (WMS — Workload Management System). Таким образом, существуют четыре базовых уровня (рис. 31):

- L4 — научные приложения;
- L3 — система управления загрузкой;
- L2 — система управления выполнением задач;
- L1 — вычислительный ресурс.

Для интеграции всех возможных вычислительных ресурсов и их оптимального использования необходимо разработать систему управления выполнением задач (этот вопрос подробно рассмотрен в разд. 3).

Рассмотрим типичное научное приложение в области ФВЭ на примере заданий обработки и анализа данных эксперимента ATLAS. С точки зрения потребностей в обработке и анализе данных ATLAS является ярким примером того, почему необходимы федеративная организация вычислительных мощностей и использование всего доступного вычислительного ресурса.

Рис. 31. Схема интеграции системы управления загрузкой и вычислительных мощностей

Типичные вычислительные потребности эксперимента ATLAS характеризуются такими величинами, как 2 млн выполненных задач, 3 млн используемых ЦПУ-часов в сутки:

— наиболее интенсивное ATLAS-«задание» — это приложение для обработки данных, использующее до 2 ГБ памяти на одном ядре в течение примерно 12 ч, обрабатывающее максимально несколько гигабайт входных данных и производящее выходные данные такого же объема;

— непрерывная глобальная передача данных — на уровне до 30 ГБ/с суммарно;

— необходимые 6 млн ядро-часов (core-hours) для первого шага обработки 1 ПБ данных, полученных от 1 млрд актов столкновений на LHC;

— совокупный поток заданий, использующий более 250 000 ЦПУ-ядер с пиковой производительностью приблизительно 0,32 Пфлопс (такую производительность обеспечивает суперкомпьютер, находящийся на 50-й позиции из списка топ-500 [58, 77]).

Есть, по крайней мере, два дополнительных параметра, заслуживающих упоминания:

1) загрузка мощностей не является статичной во времени и зависит от многих причин: при хорошо определенном среднем значении в 1,5 млн задач в день существуют сильные временные флуктуации, когда потребность в ресурсах возрастает в 10 и более раз в течение короткого промежутка времени (день), что соответствует классическим примерам систем с ограничением в передаче данных и доступа к распределенному вычислительному ресурсу;

2) стабильная во времени потребность в компьютерных мощностях (так называемых steady state demand).

Даже при отсутствии пиковых нагрузок ресурс, предоставляемый консорциумом WLCG, недостаточен. Пример ATLAS не единичен, другие эксперименты на LHC — ALICE, CMS и LHCb — сталкиваются с подобной проблемой.

Необходимо отметить, что следующее поколение ПО экспериментов в области ФВЭ и ЯФ будет гораздо более комплексным [59], сложным и неоднородным, поэтому эра постчастицы Хиггса (исследование свойств новой частицы и, возможно, поиск второй и третьей частицы *a la* Хиггса) потребует в будущем гораздо большего вычислительного ресурса в дополнение к тому, что количество данных, набираемых ежегодно, будет расти (как это было рассмотрено ранее).

Более того, «загрузка» суперкомпьютеров приложениями в области ФВЭ и ЯФ может быть не только потребностью экспериментов, но и иметь сильный экономический аргумент для «владельцев» подобных машин. Хотя точное число загрузки суперкомпьютерных центров широко не афишируется, можно предположить, что оно не превышает 90 % (согласно данным последних суперкомпьютерных конференций (SC17–SC19 2017–2019 гг.) и техническим

данным некоторых центров России, США и Европы. Таким образом, около 10 % ресурса могут быть использованы в фоновом режиме (backfill) без изменения существующего портфеля задач, что повысит эффективность и процент использования суперкомпьютеров, т. е. более гибкое и эффективное использование суперкомпьютерного ресурса возможно за счет приложений в области ФВЭ и ЯФ в случае, когда такой ресурс доступен в суперкомпьютерном центре и не используется для специальных приложений.

Важно понимать, что подобный подход представляет интерес для многих научных приложений (за пределами экспериментов на LHC) и для многих научных дисциплин.

2.3. Роль суперкомпьютеров для приложений в области физики высоких энергий и ядерной физики. Научные приоритеты в области ФВЭ и ЯФ представляют проблемы управления и работы с большими данными, требующие современных вычислительных подходов и, следовательно, служат проводниками новых идей по созданию интегрированной компьютерной и информационной инфраструктуры. Для ФВЭ в приоритеты входят исследования свойств бозона Хиггса с целью лучше понять происхождение массы элементарных частиц и поиска новых законов физики, для ЯФ — исследования свойств кварк-глюонной плазмы. Во избежание потенциальных проблем, связанных с недостатком существующих и будущих ресурсов, предоставляемых консорциумом WLCG, эксперименты на LHC (как и будущие эксперименты на FAIR, NICA, LSST) начали рассматривать суперкомпьютеры как возможный дополнительный вычислительный ресурс, который позволил бы не уменьшать объемы данных, набираемых с установок, а наоборот, вести моделирование физических процессов в необходимых объемах. Как отмечалось ранее, время выполнения задач моделирования методом Монте-Карло (см. рис. 14) занимает до 42 % всех вычислительных ресурсов. Первые два этапа работы LHC (2009–2013, 2015–2018 гг.) убедили физиков, что их ПО нуждается в фундаментальном переосмыслении. Возможность использования особенностей суперкомпьютеров, таких как параллельные вычисления и графические процессоры, должна привести к созданию нового поколения ПО для физических экспериментов. Разработка ПО для выполнения на СК должна стать задачей ФВЭ и ЯФ, и эта задача должна быть выполнена до начала четвертого этапа работы коллайдера.

После успеха открытия новой частицы ATLAS и CMS проводят более точные измерения и исследования ее свойств, необходимые для дальнейших открытий, которые станут возможными при гораздо более высоких энергиях работы LHC. Одновременно потребность в моделировании и анализе будет превосходить ожидаемую мощность вычислительных мощностей WLCG, ценой отказа от использования СК будет сокращение диапазона и точности физических исследований. Даже сегодня важные задачи анализа физических данных, которые требуют больших наборов моделируемых событий, откла-

дываются на месяцы, так как существующие ресурсы WLCG полностью заняты. Кроме того, некоторые физические процессы, представляющие интерес для LHC, практически невозможно смоделировать на традиционных грид-ресурсах из-за чрезвычайно высоких вычислительных требований. Суперкомпьютеры предлагают уникальную возможность смоделировать и создать такие наборы данных посредством массивного распараллеливания вычислений. Кроме того, сгенерированные события могут представлять интерес для физиков-теоретиков по всему миру и не ориентироваться только на научные интересы, специфические для экспериментов на LHC. По мере роста энергии и светимости коллайдера отсутствие адекватного вычислительного ресурса приведет к отставанию выполнения физической программы экспериментов.

Поиск новых открытий в фундаментальной физике требует сравнения результатов экспериментов и предсказаний новых теорий, наборов реальных событий и соответствующих наборов моделированных событий методом Монте-Карло для различных теоретических моделей. Кроме того, для получения статистических выводов требуется массовое моделирование всех известных физических процессов. Из-за необходимости сравнения имеющихся данных с моделированными физическая программа экспериментов на LHC часто ограничена не возможностями к набору данных и проведению измерений, а способностью проанализировать набранные данные из-за отсутствия соответствующих моделированных данных. Кроме того, существуют ограничения на общее количество событий, их сложность, а в некоторых случаях — и на полное моделирование теоретических моделей. Задачи моделирования в области ФВЭ и ЯФ хорошо подходят для работы на СК в режиме фоновой загрузки. Моделирование эксперимента состоит из ряда последовательных шагов (см. рис. 13), из которых шаг генерации событий (evgen) является наиболее ЦПУ-затратным, с минимальными требованиями к вводу/выводу, дальнейшее «разбиение» этого шага на десятки тысяч «шажков» по генерации отдельных событий позволит выполнить код программ на многих возможных аппаратных архитектурах, и СК являются наиболее вероятными кандидатами в силу своей вычислительной мощности и минимальной коммуникационной нагрузки между задачами, выполняемыми на рабочих узлах.

Полностью оптимизированное использование существующих и новых суперкомпьютерных мощностей для приложений в области ФВЭ и ЯФ является долгосрочной задачей, требующей работы в течение нескольких лет. Таким образом, эти приложения требуют не только больших объемов вычислений, но и возможностей, которые могут предоставить только суперкомпьютеры. Вклад суперкомпьютеров порядка 100 млн или более ЦПУ-часов в год становится важным и ценным дополнением к имеющимся ресурсам WLCG. Следует отметить, что выполнение приложений в области ФВЭ и ЯФ может быть близким к идеальному «заполнению пустот», когда они используют сво-

бодный ресурс в дополнение к классическим приложениям, выполняемым на суперкомпьютерах, таким как моделирование климата или теоретические расчеты в квантовой хромодинамике.

Интеграция суперкомпьютеров и ресурсов облачных вычислений потребовали разработки новой архитектуры системы управления потоками заданий, которая могла бы работать с динамически изменяющимися вычислительными ресурсами и использовать мощности, доступные в течение относительно коротких периодов времени. Одновременно необходимо было расширить компьютерную модель и ввести понятие ВЦ без «дискового элемента», потому что ни СК-центры, ни центры облачных вычислений не предоставляют дискового ресурса для постоянного хранения данных.

Итак, в разд. 2 обоснована важность разработки новой архитектуры системы управления потоками заданий и ПО для ее реализации. Распределенная система обработки данных должна работать с динамически изменяющимися вычислительными ресурсами и использовать мощности, доступные в течение относительно коротких временных интервалов. Существует необходимость расширения компьютерной модели и введения понятия ВЦ без «дискового элемента», поскольку ни суперкомпьютерные центры, ни центры облачных вычислений не предоставляют дисковый ресурс для постоянного хранения данных (речь идет о дисковом ресурсе в масштабе сотни петабайт, необходимом для экспериментов на LHC, в 2020 г. идея использования ВЦ без «дискового элемента» получила дальнейшее развитие при создании прототипов «озер научных данных»). Также в разд. 2 определена роль суперкомпьютеров для научных приложений в области ФВЭ и ЯФ.

3. РАЗРАБОТКА КОНЦЕПЦИИ, МЕТОДОВ И АРХИТЕКТУРЫ СИСТЕМЫ УПРАВЛЕНИЯ ПОТОКАМИ ЗАДАНИЙ В РАСПРЕДЕЛЕННОЙ ГЕТЕРОГЕННОЙ КОМПЬЮТЕРНОЙ СРЕДЕ

Этот раздел посвящен разработке методов управления загрузкой и архитектуре системы управления заданиями в гетерогенной компьютерной среде. Сформулированы требования к системе, проведен анализ классов научных приложений в области ФВЭ и ЯФ, предложена логическая модель данных системы распределенной обработки. Рассмотрены вопросы разделения вычислительного ресурса между различными потоками заданий для обработки, моделирования и анализа данных, принципы построения системы для глобальной распределенной обработки данных, ее уровни, функции и взаимодействие компонентов системы между собой, а также взаимодействие системы для обработки данных с внешними системами, такими как система управления данными и информационная система.

Как обсуждалось в разд. 2, при создании федеративной распределенной киберинфраструктуры были функциональные трудности, связанные с управлением загрузкой и описанием ресурсов (проблема описания вычислительных ресурсов была решена в созданной информационной системе AGIS/CRIC), другая часть проблем была связана с идентификацией пользователей, протоколами обмена информацией и определением политики использования и разделения ресурса. В последнее время есть успешные попытки синхронизировать и гармонизировать вторую группу проблем, но в этом есть смысл, если существуют все узловые функциональные блоки для создания ФРКИ.

Для решения проблемы управления загрузкой в гетерогенной среде необходимы система управления потоком заданий нового поколения (WMS) и модели выполнения заданий для динамически определяемых федеративных гетерогенных ресурсов. Такая система должна работать независимо от типа инфраструктуры, ее неоднородности и с учетом параметров, определяющих динамику возможного изменения ресурса. В целом предлагаемая модель выполнения заданий и управления загрузкой имеет следующие основные особенности:

— интеграцию информации о выполняемом задании и ресурсах;

— стратегию выполнения, основанную на последовательности решений, используемой для выполнения данного задания, которая может измениться при условии трансформирования инфраструктуры и/или типа задания;

— подход, позволяющий интегрировать рабочую нагрузку и ресурсы: а) оценить потребности, необходимые для выполнения задания (рабочей нагрузки); б) оценить возможности ресурса, выработать стратегию выполнения и начать выполнение задания. Схематично базовые уровни системы и их взаимодействие показаны на рис. 32.

Таким образом, новое поколение системы управления загрузкой должно: а) абстрагировать задание от управления ресурсами; б) иметь достаточно высокую гранулярность для выбора ресурса, наилучшим образом соответствующего заданию; в) проводить жесткий контроль выполнения заданий; г) обеспечивать создание инфраструктуры управления заданиями без явного управления ресурсами.

3.1. Классификация типов заданий современного эксперимента в области физики высоких энергий и ядерной физики. ***3.1.1. Моделирование методом Монте-Карло.*** Моделирование физических процессов и детекторов методом Монте-Карло является первым этапом при написании технического документа для любого эксперимента. Стандартная последовательность преобразований в задачах Монте-Карло физического эксперимента представляет собой цепочку логически связанных заданий для следующих этапов их выполнения (шагов обработки данных) [61]:

— генерация событий (шаги evgen, evnt);

— моделирование (шаг simul);

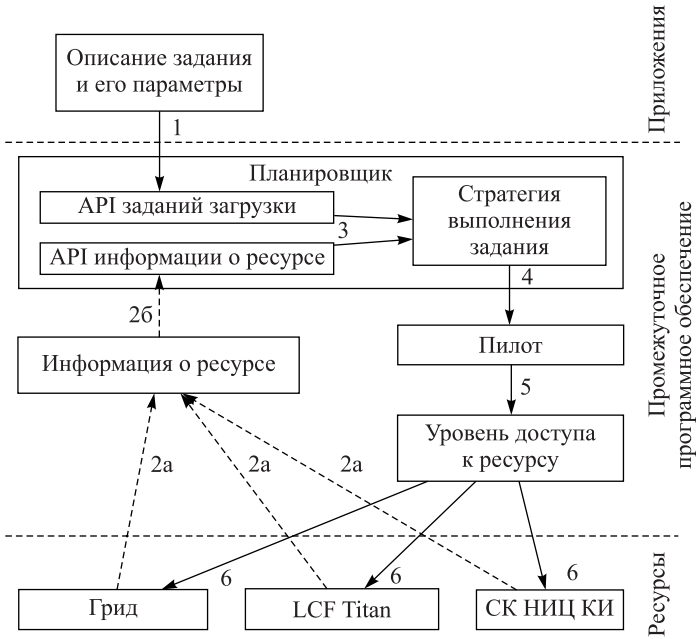

Рис. 32. Этапы (1–6) выполнения задания и взаимодействия с гетерогенными вычислительными ресурсами

— реконструкция (шаг recon);

— создание объектов для последующего анализа данных (шаг AOD).

Этапы генерации событий и моделирования требуют значительного вычислительного ресурса и отличаются от последующих этапов сравнительно невысокими требованиями по обмену информацией между памятью и диском на рабочем узле. На рис. 14 показано время выполнения для различных типов заданий. Из этого рисунка следует, что время, затраченное на моделирование и генерацию событий, составляет 42 % от общих вычислительных потребностей современного эксперимента. В запросах на выполнение заданий последовательность выполнения может начинаться с любого шага, например с шага реконструкции, либо выборочно могут быть использованы события, сгенерированные ранее. В рамках одного физического эксперимента или «виртуальной организации» (в терминах грид) все запросы на моделирование поступают и выполняются централизованно через одно «входное окно», приоритеты определяются физической программой эксперимента. Сводный запрос составляется на основании запросов групп, занимающихся исследованиями по различным направлениям (Стандартная модель, свойства бозона Хиггса, суперсимметрия и т. д.). Преимуществом данного класса заданий является предсказуемость требуемого вычислительного ресурса и, как правило,

времени выполнения задания. Данный класс потоков заданий выполняется централизованно для всего эксперимента.

3.1.2. Обработка и переобработка данных эксперимента. Обработка данных для системы отбора событий триггера «высшего» уровня (триггера HLT). Централизованная обработка данных эксперимента выполняется в два шага:

1) реконструкция (шаг recon);

2) создание объектов для последующего анализа данных (шаг AOD).

Как правило, физические эксперименты стараются провести первую обработку данных в течение 24–48 ч после окончания набора (при наличии калибровок и вычислительных мощностей), т. е. это непрерывный поток заданий во время работы ускорителя. Переобработка данных проводится для уточненных калибровок (и/или с уточненной информацией о параметрах работы детектора), а также при изменении версии ПО (в среднем переобработка составляет 1,5 раза для каждого года работы ускорителя). Переобработка данных проводится централизовано в течение 1–2 мес. Обработка данных для системы отбора событий «высшего уровня» HLT (High Level Trigger) отличается только версией ПО и требованием получения конечного результата в течение нескольких часов, что накладывает дополнительные требования на систему управления загрузкой. Во всех перечисленных случаях последовательность шагов обработки не отличается от «рутинной» реконструкции событий при шаге recon. Программы реконструкции, используемые для реальных и моделируемых событий, одинаковы, особенностью для данного класса заданий является разветвленная иерархия, когда создаются объекты промежуточных форматов (например, ESD — Event Summary Data), имеющие время «жизни» несколько месяцев, при этом результаты обработки одной задачи могут быть полностью или частично входными данными для нескольких новых задач. Класс заданий выполняется централизованно для всего эксперимента [60].

3.1.3. Обработка, фильтрация и анализ данных, проводимые физическими группами. Задания обработки для отдельных физических групп имеют ту же последовательность, что и при обработке данных эксперимента, но могут отличаться версиями ПО. Задания фильтрации отличаются высокими требованиями к вводу/выводу и в результате работы создают выборку событий, используемую для физического анализа, проводимого группами и отдельными учеными. Последовательность задач анализа физических данных чаще всего имеет простую структуру и состоит из одной задачи для каждого набора данных. Входные параметры задачи хранятся в файле текстового формата. Особенностью анализа, проводимого физическими группами, является большое количество пользователей, что требует продуманной системы аутентификации и авторизации при управлении заданиями и доступа к данным. Класс заданий выполняется централизованно для каждой группы. Как правило, количество групп соответствует количеству научных тем, по которым ведутся

исследования, и варьируется от 10 до 30, в зависимости от сложности эксперимента и его программы. (Обработка «поездом», которая будет описана ниже, позволила сократить количество вариаций и выполнять централизованно данный класс заданий.) В результате этого типа обработки создаются данные в формате DAOD (Derived Analysis Object Data) и данные в табличном формате NTUP (от ntuples).

3.1.4. Физический анализ данных. Это наиболее важный и конечный шаг всего процесса получения физического результата. Подразумевается, что на данном этапе ученые используют наборы данных, созданные либо на шаге реконструкции, либо в результате выполнения заданий физических групп. В реальности эта группа заданий наиболее разнородна и непредсказуема как по набору шагов, так и по количеству используемых версий ПО. Задания данной группы децентрализованы, запросы на анализ физических данных поступают в среднем от 1000 ученых ежемесячно, а количество заданий составляет до 50 % от всех выполняемых системой заданий.

3.2. Модель данных. После проведения анализа классов заданий необходимо определить базовые компоненты системы и построить логическую модель данных. Были определены следующие сущности.

Запрос (Request) — верхний уровень абстракции, объединяющий задания одного класса. Типичным примером может быть запрос коллаборации на (пере)обработку всех данных для определенного периода работы установки, или кампания по моделированию детектора и физических процессов для этапа работы ускорителя с новыми параметрами (энергии, светимости, множественности событий). Таким запросом может также быть запрос физической группы на специфический анализ данных, например поиск редкого распада в одном из каналов. Каждый запрос имеет статус. Статус запроса отражает его текущее состояние: подготовлен, в процессе выполнения, выполнен. Изменения статуса запроса выделены в отдельную сущность для возможности хранения истории изменений состояния, а также для удобства мониторирования и учета работы (аккаунтинга).

Список входных параметров (input list) включает в себя параметры для запуска задач генерации событий, входные наборы данных и атрибуты, относящиеся к ним, например приоритет и комментарий.

Шаблон шагов обработки данных (step template) — заранее определенный набор параметров (включая информацию о форматах выходных данных, версии ПО, . . .), который содержит всю необходимую информацию для запуска задач при данном шаге обработки (recon, AOD), моделирования (evnt, simul) или анализа.

Исполняемый шаг обработки (step execution) — иерархически зависимые «инициализированные» шаблоны шага. Исполняемый шаг транслируется в выполняемые задания и хранит их текущее состояние, как и состояние всего шага в целом.

Вертикальный срез (slice) — комбинация из исполняемых шагов обработки, каждый запрос состоит из одного или нескольких срезов.

Задание (task) — сущность для передачи параметров в систему запуска задач. Имя задания формируется автоматически исходя из информации в запросе, шаге... Каждое задание имеет уникальный идентификатор. Задание может иметь входные данные, результатом выполнения *задания* является набор файлов, организованный как датасет. Имя датасета наследуется из имени задания с учетом выходных форматов, определяемых в шаге выполнения задания, и версии ПО, используемого для данного шага. Задание имеет состояния, описывающие ход его подготовки и выполнения. Каждое задание состоит из одной или многих задач (до 10 000).

Задача (job) — единица измерения работы системы управления загрузкой. Задача выполняется на единичном вычислительном элементе грид (см. разд. 1), для суперкомпьютеров — на рабочем узле СК. Она может иметь входные данные (файлы) и записывает результат работы в выходные файлы. Задачи имеют состояния, описывающие ход их подготовки и выполнения. Каждая отдельная задача имеет уникальный идентификатор.

Система хранит информацию о всех созданных и выполненных запросах, заданиях и задачах, а также метаинформацию о ходе выполнения каждой из сущностей.

Пилотная задача (pilot job) — некоторый шаблон задачи, выполняемый на СЕ, запрашивающий реальную задачу при наличии определенных условий (например, свободного ресурса, наличия версии ПО...). Пилот следит за выполнением задачи на СЕ, отвечает за передачу результатов выполнения задачи на элемент хранения и удаляет информацию о выполнении задачи на рабочем узле после ее выполнения. Пилотные задачи имеют состояния, описывающие ход их выполнения. Задача не может быть выполнена (а точнее, быть направлена для выполнения на СЕ), если нет информации о пилотной задаче, успешно работающей на данном СЕ.

Центральной сущностью рабочего процесса является запрос. Пользователь создает запрос, определяет его параметры и набор входных данных. К каждому набору входных данных пользователь задает последовательный набор шаблонов шагов выполнения, который транслируется системой в последовательность выполняемых заданий.

Основной особенностью созданной модели данных является логическое разделение этапов обработки («запроса») на «срезы», «шаги», «задания» и «задачи». Шаги являются шаблонами, из которых после инициализации всех параметров создаются задания, которые, в свою очередь, транслируются в задачи и непосредственно отвечают за обработку данных на вычислительных ресурсах.

Набор шагов также является шаблоном, что позволило реализовать сложные процессы обработки, моделирования и анализа данных физического экс-

перимента (в ATLAS большой вклад в разработку логической модели данных внес сотрудник НИЯУ МИФИ М. С. Бородин). В данной модели также использовались понятия «датасет» и «контейнер» для организации данных. Файлы одного формата, произведенные одним заданием, организованы как единый набор (датасет), который является единицей управления данными. Наборы (датасеты), произведенные разными заданиями, но имеющие одинаковые метаданные (версию ПО, используемого для обработки, энергию ускорителя, калибровочные константы), помещаются в один контейнер. Контейнер может использоваться как «входной» набор данных для задания.

3.3. Новые методы организации поточной обработки данных. Обработка данных «поездом» и «постоянная» обработка данных. Созданная модель данных, введение понятий «датасет» и «контейнер» позволили реализовать новые подходы по созданию потоков обработки данных физическими группами. Опыт обработки данных физическими группами во время первого этапа работы LHC требовал упорядоченного подхода, это было продиктовано как ограничениями, связанными с имеющимися вычислительными ресурсами (ресурс грид был практически полностью использован), так и тем, что алгоритмы обработки данных отдельными группами были схожи между собой и различались только на конечном этапе фильтрации событий. На рис. 33 приведена статистика обработки для 1400 наборов данных (шкала абсцисс) различными физическими группами (шкала ординат). Цветом показано количество заданий для (пере)обработки данных. Из графика следует, что группы во многих случаях работают с одинаковыми наборами данных и количество попыток обработать один набор группами JetMet и SUSY достигло 14 раз.

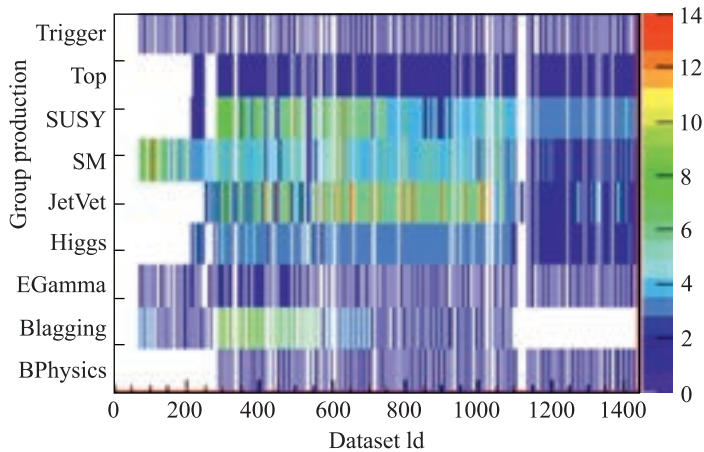

Рис. 33 (цветной в электронной версии). Статистика обработки наборов данных физическими группами эксперимента ATLAS

Созданная модель и методы обработки, рассмотренные ниже, позволили существенно снизить общее (в целом по эксперименту или по виртуальной организации) время обработки, уменьшить количество ошибок при создании запросов на выполнение заданий, уменьшить общее количество выполняемых заданий и общее время, необходимое для получения научных результатов. Двумя такими методами стали: создание запросов на «постоянную» обработку, создание запросов на обработку данных «поездом».

3.3.1. «Постоянная» обработка. Как было рассмотрено в п. 3.1, входными данными для класса заданий, выполняемых физическими группами, являются данные, произведенные централизованно после шага реконструкции.

Классическая методика состоит в том, что только после завершения этого шага начинается выполнение заданий физических групп. Это требовало «отслеживания» выполнения шага реконструкции со стороны многих групп. Новая модель позволила существенно упростить весь процесс. Пользователь (в данном случае физическая группа) заранее определяет шаблон шагов выполнения запроса, где «прописан» набор шагов обработки, и поток входных данных, который содержится в контейнере. После этого запрос поступает на выполнение в систему распределенной обработки данных. Ко времени поступления запроса «контейнер» может не содержать данных или содержать только часть из них. По мере заполнения контейнера данными автоматически определяются задания обработки и/или анализа согласно предопределенным шагам шаблона. Реализация такого метода показала свою эффективность для запуска заданий отдельными физическими группами, где входным потоком являются данные, полученные после первичной обработки. Это позволило быстро начать этап анализа новых данных, с задержкой менее 36 ч после их получения.

3.3.2. Обработка данных «поездом». Обработка «поездом» позволяет в одном запросе запустить задания с ПО для нескольких физических групп, для одного и того же набора данных. Каждая группа определяет ПО, таким образом, «поезд» имеет столько «вагонов», сколько различных версий ПО было определено.

Но при формировании заданий одинаковые версии ПО составляют единое задание («физические группы» попадают в один «вагон»). Сложностью работы с такими запросами является большое количество возможных типов обработки при большом количестве версий ПО. Основанная на выбранной модели данных реализация данного метода позволила существенно автоматизировать процесс создания подобных задач.

Следует отметить, что также был реализован подход «постоянный поезд», когда в качестве входных данных использовался контейнер, описанный в п. 3.3.1. В таком случае общее время обработки оптимизируется, поскольку задачи задания n с входными данными, произведенными заданием m, могут

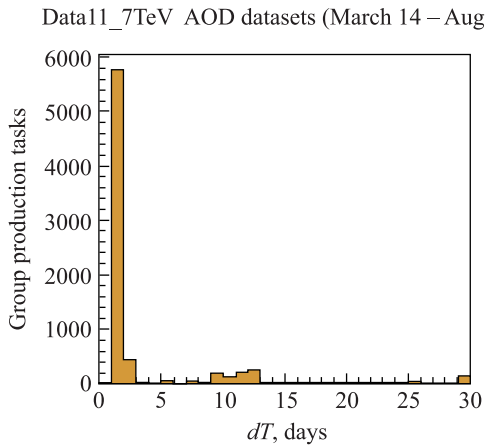

Рис. 34. Время задержки с начала обработки данных физическими группами

начаться, как только будут произведены первые файлы, т. е. еще до полного завершения задания m.

Результат реализации новых методов обработки показан на рис. 34, более чем в 75 % случаев физические группы начали обработку в течение 24 ч после того, как данные были доступны на грид-сайтах.

3.4. Архитектура системы управления загрузкой и глобальной обработки данных физического эксперимента. Принятие концепции грид и модели распределенной обработки данных потребовали создания нового поколения систем управления загрузкой (систем управления потоками заданий) и пересмотра парадигмы использования вычислительного ресурса для различных классов физических задач. Рассмотрим, каким требованиям должна отвечать система управления загрузкой современного физического эксперимента:

— сотни вычислительных центров, распределенных по всему миру, для конечного пользователя должны выглядеть как единый ВЦ;

— система должна обеспечивать доступ и выполнение заданий на $O(10^2)$ ВЦ, для $O(10^3)$ пользователей и $O(10^6)$ задач в день;

— вычислительные центры могут быть представлены не только центрами грид, но и суперкомпьютерными центрами и центрами облачных вычислений, при этом все центры рассматриваются как равноправные участники. Весь набор центров составляет единый пул вычислительных ресурсов. Отметим, что этот подход стал возможен после перехода к описанной в разд. 1 «смешанной компьютерной модели» и обеспечил создание гетерогенной киберинфраструктуры при интеграции дополнительных вычислительных ресурсов с грид-ресурсами;

— очередь на выполнение пользовательских заданий в распределенной среде должна быть единой, сравнимой по функциям с очередью пакетной

обработки на локальном ВЦ (все участники эксперимента («виртуальной организации», ВО) должны иметь доступ к ресурсам ВО через единую систему запуска заданий или, на более высоком уровне, через систему «запросов»);

— ошибки в работе вычислительных центров и задержки, связанные с распределенным характером обработки, должны быть минимизированы (для этого необходимо использовать «позднюю привязку» реально выполняемой задачи к вычислительному ресурсу, используя концепцию «пилотных задач»);

— сложность и разнообразие промежуточного программного обеспечения (ППО) грид должны быть «скрыты» от пользователя (для этого необходимо выполнение следующих условий: а) система управления загрузкой «знает» о ППО и взаимодействует с ним (в обоих направлениях), конечный пользователь взаимодействует только с системой управления загрузкой; б) механизмы автоматизации управления загрузкой «скрыты» от пользователя);

— изменения и эволюция ППО не должны менять пользовательский интерфейс управления заданиями;

— система должна быть адаптируема к изменениям в аппаратном и программном обеспечении вычислительных центров, и эти изменения не должны быть видимыми для пользователя;

— единая система управления загрузкой должна использоваться для всех классов задач физического эксперимента, таких как моделирование, реконструкция, физический анализ, а также для потоков заданий, генерируемых экспериментом, физическими группами, отдельными учеными;

— система должна обладать высокой степенью автоматизации в части обнаружения и «исправления» ошибок, связанных со сбоями в работе распределенной инфраструктуры;

— мониторирование и контроль должны быть частью системы управления загрузкой;

— задания должны использовать ресурс, выделенный для работы виртуальной организации, согласно единой системе приоритетов, пользовательских квот, квот для классов задач.

Архитектура системы должна быть разработана таким образом, чтобы обеспечить непрерывный и оптимальный доступ научного сообщества к вычислительным ресурсам. Это должно быть достигнуто за счет использования расширяемой многоуровневой архитектуры. На рис. 35 схематично представлены уровни системы управления потоком заданий.

Ее архитектура имеет три основных уровня абстракции:

• DEFT (Database Engine For Tasks) — подсистема верхнего уровня. DEFT принимает запросы на выполнение заданий (через специальный пользовательский интерфейс и/или из подготовленных списков в формате, удобном для пользователя: текстовом файле, документе google или excel), обрабатывает их и отвечает за формирование шагов обработки, заданий, входных данных и параметров.

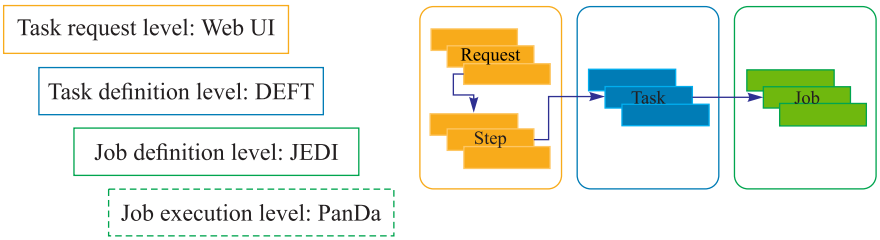

Рис. 35. Логические уровни системы управления потоками заданий

• JEDI (Job Execution and Definition Interface) — подсистема среднего уровня, использующая описания заданий, подготовленных DEFT. JEDI динамически определяет количество задач для каждого задания и отвечает за запуск и выполнение отдельных заданий.

• PanDA — основной «мотор» системы, подсистема выполнения задач. PanDA определяет, какой ресурс и в какой момент будет использован каждой из задач, получает информацию от пилотных заданий и информационной системы, управляет ходом выполнения задач.

Информация о выполняемых заданиях находится под контролем JEDI и хранится в рабочей базе данных (БД). Программы мониторирования и учета (аккаунтинга) используют копию БД JEDI, что гарантирует отсутствие нежелательного доступа к рабочей БД, включая запросы мониторирования, требующие агрегации информации из различных таблиц БД, что может привести к снижению производительности системы в целом. Определены основные архитектурные компоненты (подсистемы) системы управления загрузкой (PanDA WMS) и их функции. Каждая подсистема должна иметь общие компоненты и настраиваемые уровни. Специализированные уровни должны быть конфигурируемы. Основные компоненты системы управления загрузкой, их взаимодействие между собой, с внешними системами (хранение и доступ к метаинформации, управление данными DDM, информационная система), а также с вычислительными ресурсами показаны на рис. 36. Система должна обеспечивать управление загрузкой с учетом особенностей трех реализаций грид в рамках WLCG: проект EGEE/EGI, проект NorduGrid, проект OSG, а также использовать дополнительные ресурсы, в том числе университетские кластеры, ресурсы облачных вычислений и суперкомпьютеры.

Рассмотрим основные компоненты уровней системы управления загрузкой и их функции.

Уровень DEFT (Database Engine For Tasks): принимаются пользовательские запросы, проверяется их корректность и формализуются запросы.

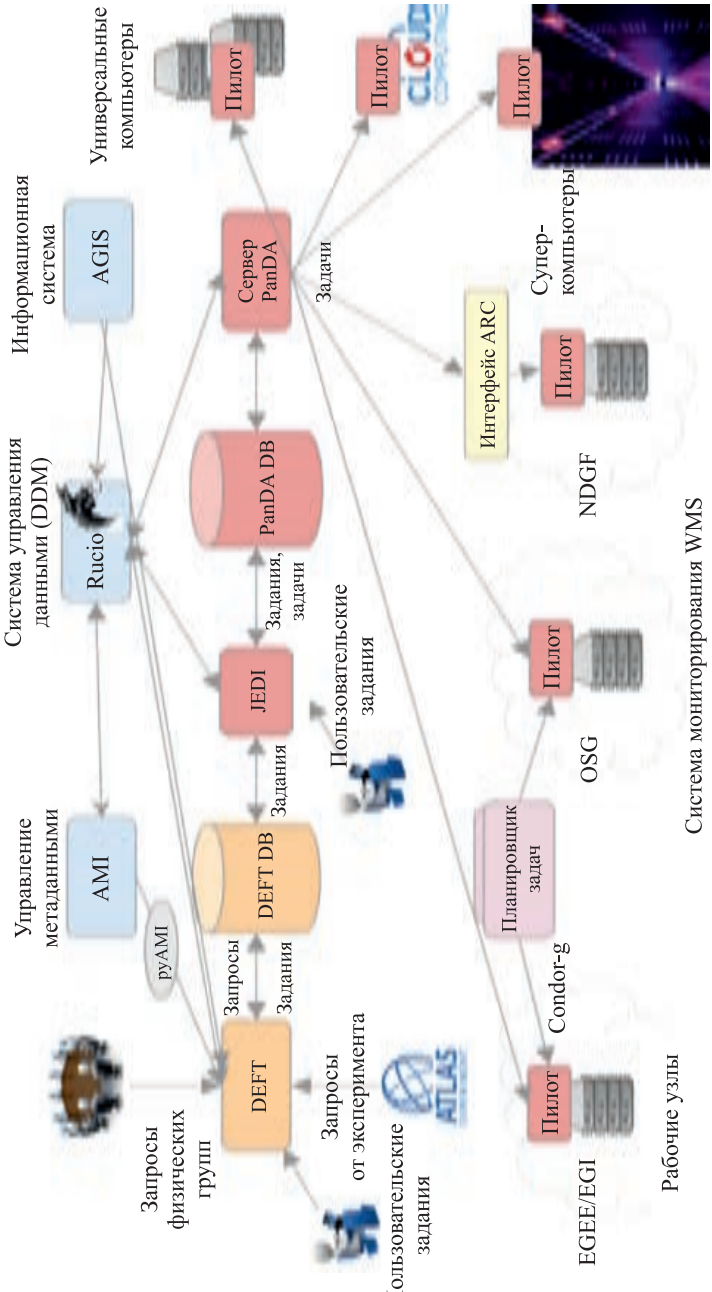

Рис. 36. Схема управления потоком заданий и основные компоненты системы для распределенной обработки данных

Формируется последовательность выполнения заданий для каждого запроса. DEFT получает необходимую информацию от систем управления данными и системы хранения метаинформации и имеет следующие компоненты.

Интерфейсы контроля и управления DEFT. Интерфейс программирования приложений (API) должен быть реализован в виде отдельного веб-сервиса, поддерживающего протокол RESTful. В API должны быть реализованы основные сервисы для мониторинга и управления объектами системы. Соответствующий веб-сервис должен обеспечивать аутентификацию с помощью специальных цифровых ключей для разграничения доступа к системе и журналирования действий пользователей. Система должна иметь механизм отложенного выполнения запросов и автоматического восстановления после сбоя. Необходимо иметь возможность управления запросами, срезами, отдельными заданиями в системе. Важной характеристикой является стабильность и скорость работы API. Интерфейс должен быть интегрирован с системой обработки ошибок для оперативного отслеживания возможных проблем при выполнении запросов к системе.

Интерфейс управления запросами, шагами выполнения, заданиями, задачами. Данная группа интерфейсов позволяет контролировать выполнение запросов, заданий, задач, например (при)останавливать их выполнение, изменять параметры задания, перенаправлять задания между ВЦ.

Интерфейсы управления потоками заданий. Позволяет управлять специальными потоками, такими как «постоянная» обработка, обработка «поездом».

База данных. База данных запросов, шагов выполнения и заданий в масштабе всей системы хранит всестороннюю статическую и динамическую информацию и метаинформацию обо всех запросах, «шагах» и заданиях, определенных, выполняемых и/или выполненных в системе, в том числе историю их выполнения, текущее состояние и статус, ошибки, возникшие при этом. Информация о заданиях во время их выполнения синхронизуется с БД следующих уровней (JEDI, PanDA).

Уровень JEDI (Job Execution and Definition Interface): принимаются формализованные описания заданий от DEFT, определяются ресурс для выполнения задания и количество задач, «разбивается» задание на задачи, а также проверяется информация о данных (через систему управления данными, DDM). JEDI работает с «рабочими очередями», описанными в ИС. JEDI и PanDA используют общую базу данных для хранения информации о состоянии заданий и задач.

Уровень PanDA (Production and Distributed Analysis). PanDA — «мотор» всей системы и наиболее сложная ее часть, которая должна включать в себя следующие компоненты.

Сервер — основа системы управления загрузкой, должен быть создан как общий сервис WMS.

База данных — база данных заданий и задач в масштабе всей системы, которая хранит всестороннюю статическую и динамическую информацию и метаинформацию обо всех заданиях и задачах, определенных, выполняемых и/или выполненных в системе, в том числе историю их выполнения и ошибки, возникшие при этом.

Пилот — пилотные задачи для сбора информации о состоянии вычислительных ресурсов. Рабочие задачи передаются успешно активированным и проверенным пилотам сервером WMS на основе критериев выбора ресурса. «Поздняя привязка» рабочих задач к месту выполнения предотвращает задержки и отказы и максимизирует гибкость выделения ресурсов для задачи на основе динамического состояния обрабатывающих ресурсов и приоритетов задач. Также пилот — основной «изолирующий слой» для WMS, инкапсулирующий сложные неоднородные среды и интерфейсы грид и средств, с которыми взаимодействует WMS. Пилотные задания для анализа способны переключить свои идентификационные данные на рабочем узле на данные пользователя, запустившего задание, используя инструмент grid, если правила компьютерной безопасности сайта (ВЦ) того требуют.

Система распределения заданий (брокер). Брокер WMS — это интеллектуальный модуль, с помощью которого выбор ресурса происходит на основе типа и приоритета задания, наличия программного обеспечения, входных данных и их местоположения, статистики задания в реальном и доступном времени ЦПУ, ресурсов хранения, связи ВЦ с «внешним миром» (пропускная способность WAN). Это ключевой компонент автоматизации потока операций WMS.

Диспетчер WMS — компонент в сервере WMS, который получает запросы на задачи от пилотов и диспетчеризирует их, используя информацию о заданиях в очереди(ях) к данному ВЦ, их приоритетах, квотах, политике распределения ресурсов и стратегии повторного запуска задачи (например, n-кратного запуска задачи на рабочем узле в случае отсутствия фатальной ошибки).

Автоматическая фабрика пилотных задач (АФП) — независимая от WMS подсистема, которая управляет поставкой пилотных задач к рабочим узлам (CE). Пилот, запущенный на рабочем узле, связывается с диспетчером и получает доступную задачу для задания, выполняемого на данном ВЦ.

Важным свойством этой схемы псевдоинтерактивного анализа, где важна минимальная задержка от запуска задачи до начала ее выполнения, является то, что диспетчеризация пилотных заданий обеспечивает устранение любых задержек в системе планирования при запуске самого пилотного задания. Механизм пилотных заданий изолирует рабочие задания от сбоев в работе грид и от систем пакетной обработки (рабочие задачи предоставляются на сайт только после успешного запуска пилотной задачи).

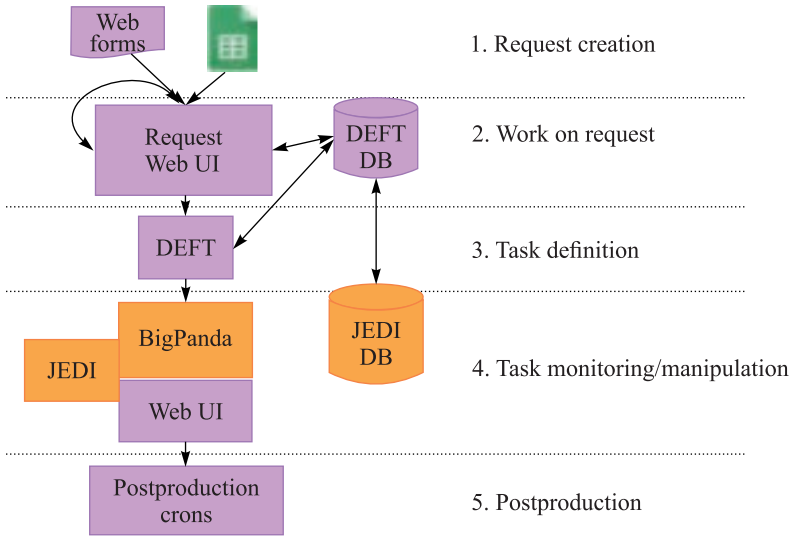

Рис. 37. Схема взаимодействия различных уровней WMS

На рис. 37 схематически показано взаимодействие между уровнями WMS. Эта система получила название megaPanDA.

Информационная система, система мониторирования, интерфейсы контроля и управления WMS, система компьютерной безопасности и аутентификации работают со всеми уровнями. Перечислим их основные функции.

Информационная система (ИС) — база данных, хранящая информацию о вычислительных центрах и очередях в масштабе всей распределенной киберинфраструктуры. ИС хранит статическую и динамическую информацию, используемую WMS (особенностью ИС AGIS/CRIC, описанной ранее, стало ее использование как системой управления загрузкой (WMS), так и системой управления данными (DDM). ИС должна быть конфигурируема и иметь возможность изменения информации в «ручном» режиме. Через ИС происходит контроль вычислительных ресурсов и параметров отдельных очередей. Содержимое базы данных ИС — это информационный кэш, агрегирующий и интегрирующий данные из отдельных информационных систем грид, систем управления данными и других источников. Пилотные задачи запрашивают информацию от ИС, чтобы сконфигурировать задачу в соответствии с параметрами очереди, в которую задачу направил брокер WMS.

Интерфейсы контроля и управления системы управления потоком заданий

Интерфейс запуска заданий и задач — пользовательский интерфейс, который обеспечивает интеграцию с разнообразными средствами для запуска заданий и задач в систему управления загрузкой. Интерфейс используется при

отладке и настройке WMS, а также предоставляет возможность для пользователя запустить задачу (или задание) и определить ее параметры (например, пользователь хочет осуществить брокеровку задачи вручную, указав определенный ВЦ в качестве места выполнения задачи).

Интерфейс управления системой. Осуществляется управление потоком операций, использованием ресурсов для пользователя, групп пользователей и регулирование квот использования ресурсов. Управление может быть как глобальным (на уровне всей системы), так и локальным (на уровне одного из ВЦ или определенной очереди).

Интерфейс управления запросами, шагами выполнения, заданиями, задачами. Данная группа интерфейсов позволяет контролировать выполнение запросов, заданий, задач, например (при)останавливать их выполнение, изменять параметры задания, перенаправлять задания между ВЦ.

Интерфейсы управления потоками заданий. Осуществляется управление специальными потоками, такими как «постоянная» обработка, обработка «поездом».

Система мониторирования. Осуществляется всесторонний мониторинг заданий (и задач) для всех классов заданий: заданий всего эксперимента, заданий отдельных групп и отдельных ученых. Система мониторирования: а) предоставляет подробную информацию о запросах, заданиях, задачах и сайтах для диагностики их состояния и возможных проблем; б) отображает информацию об использовании процессорного времени, квотах, правильности работы и производительности подсистем megaPanDA и используемых вычислительных средств.

Система аутентификации и компьютерной безопасности. WMS должна быть интегрирована с соответствующими системами безопасности грид.

Управление данными после завершения работы задания и/или завершения запроса (PostProduction). Этот уровень необходим для управления состоянием заданий и запросов после завершения работы заданий, например при повторной обработке экспериментальных данных с уточненными значениями калибровок детектора или при повторении реконструкции моделируемых данных с новой версией ПО. В обоих случаях данные, произведенные ранее, могут быть признаны «устаревшими» (как это обсуждалось в п. 1.4.2). В таком случае статус заданий также должен быть изменен, а данные удалены со всех элементов хранения ВО. Этот уровень может быть реализован как набор программ-агентов и через пользовательский интерфейс. Например, запрос может быть направлен не только на «производство» данных, но и на их удаление.

Более детально архитектура системы для распределенной обработки данных и взаимодействие между различными ее подсистемами показаны на рис. 38.

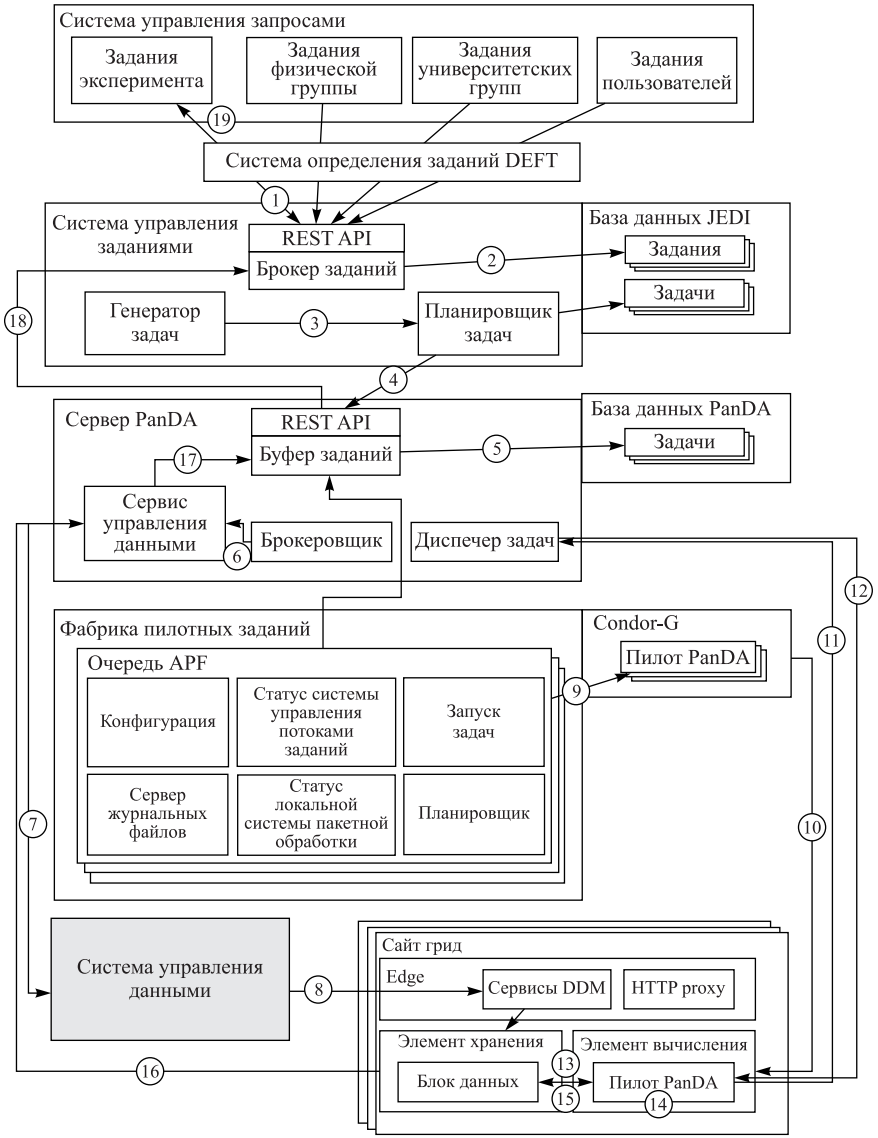

Рис. 38. Архитектура системы для распределенной обработки данных megaPanDa

3.5. Система обработки, моделирования и анализа данных эксперимента ATLAS. ***3.5.1. Эксперимент и международное сотрудничество ATLAS.*** ATLAS (A Toroidal LHC ApparatuS) является одной из двух установок общего назначения (вторым таким детектором является установка CMS).

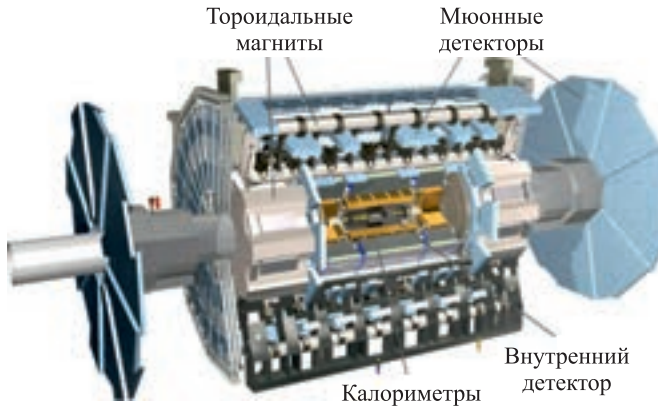

Рис. 39. Общий вид детектора ATLAS

В 2013 г. ученые ATLAS и CMS открыли предсказанную в 1964 г. частицу — бозон Хиггса. Научная программа ATLAS имеет следующие основные направления исследований:

— использование бозона Хиггса как инструмента для новых открытий;

— поиск темной материи;

— поиск новой физики частиц, взаимодействий, физических законов.

Научное сообщество ATLAS включает более 3000 ученых из 40 стран. ATLAS — самая крупная в мире научная установка в области ФВЭ и ЯФ. Она размещена в шахте на глубине 100 м и имеет длину около 45 м и высоту более 25 м (высота семиэтажного дома), ее вес составляет около 7000 т (общий вид установки и ее основные компоненты приведены на рис. 39). Основными детекторами установки являются: внутренний детектор, электромагнитный и адронный калориметры, мюонный спектрометр. Установка имеет более 150 млн каналов считывания. Поток информации с установки составляет 1 ПБ/с, отбор интересных событий для последующей обработки и анализа производится трехуровневой системой триггера. На вход триггера первого уровня события поступают с частотой 40 МГц, на последней стадии отбора (триггер высокого уровня) для дальнейшей обработки и анализа отбирается примерно 1000 событий в секунду.

Размер «сырого» события составляет 1,5 МБ, на втором этапе работы коллайдера в год было набрано $O(10^7)$ событий, и ежегодно моделируется 4 млрд событий. Физические явления, исследуемые в эксперименте, представляют собой очень редкие физические процессы. На рис. 40 представлен график сечений (по оси ординат в логарифмическом масштабе). Из графика видно, что 10^{11} столкновений необходимы для того, чтобы найти среди них 10 бозонов Хиггса.

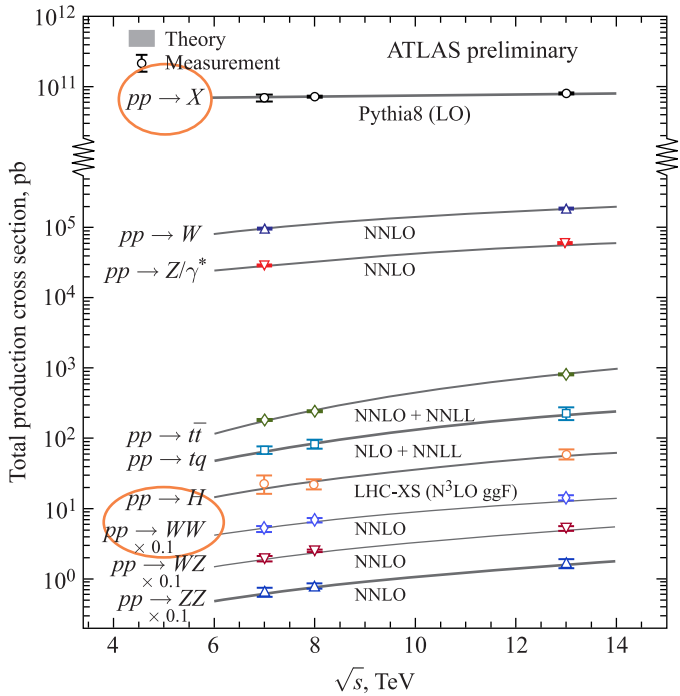

Рис. 40. График сечений (протон-протонные столкновения на LHC)

Вероятность растет логарифмически с увеличением энергии (ось ординат). Теоретические предсказания (линии) хорошо согласуются с измерениями (маркеры). Общий (все форматы для моделируемых и реальных данных) объем управляемых данных ATLAS на начало 2020 г. составил 460 ПБ и представлен на рис. 41. Из графика видна корреляция роста объема данных с периодами работы коллайдера: первая фаза работы (2010–2013 гг.), вторая

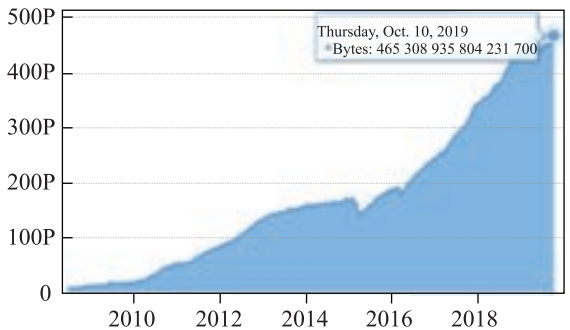

Рис. 41. Объем управляемых данных эксперимента ATLAS

фаза работы (2014–2018 гг.). Для сравнения отметим, что все письменное наследие человечества на всех языках мира составляет 50 ПБ.

В табл. 3 приведен объем информации для различных групп в эпоху больших данных [63, 64]. Если компьютерные мощности WLCG значительно уступают имеющимся мощностям гигантов ИТ-индустрии, таким как Google и Amazon, то по объемам информации ФВЭ и ЯФ являются заметным игроком на «поле BigData». Поэтому задача создания систем обработки и анализа данных для физических установок класса мегасайенс вызывают интерес со стороны многих ИТ-компаний. Рассмотрим, как была реализована глобальная система распределенной обработки данных эксперимента ATLAS.

3.5.2. Система обработки, моделирования и анализа данных эксперимента ATLAS. Для моделирования, обработки и анализа данных экспериментов класса мегасайенс требуется слаженная работа гетерогенных вычислительных ресурсов. В частности, эксперимент ATLAS использует ресурсы 250 ВЦ по всему миру, а также мощности суперкомпьютерных центров, национальные, академические и коммерческие ресурсы облачных вычислений. Разработанные методы и описанные выше подходы позволили создать систему обработки и анализа данных эксперимента ATLAS на LHC. Система была успешно реализована для управления вычислительными ресурсами ATLAS в конце 2013 г., а после тонкой настройки в начале 2014 г. она была принята как основная система управления заданиями эксперимента. Система получила название ProdSys2–PanDA (ProdSys2 — Production System 2-го поколения,

Таблица 3. **Объем информации для различных групп в эпоху больших данных**

Источник информации	Размер в единицу измерения	Комментарий	Объем информации
Библиотека конгресса США	—	—	~ 200 ТБ
Одно электронное сообщение (e-mail)	~ 1 КБ	30 трлн писем в год, без учета спам-рассылки	30 ПБ $\times N$ копий
Электронная фотография	~ 2 МБ	500 млрд фотографий в год, 25 млрд фотографий в Facebook	1 ЭБ
LHC	~ 2 МБ/ событие	4 эксперимента, «сырые» и приведенные события	700 ПБ
WWW	—	25 млрд страниц, 1 трлн документов	~ 1 ЭБ
YouTube	—	Ежегодно	15 ПБ
Диски blue ray	$\sim$ 25 ГБ	Ежегодно 100 млн шт.	2,5 ЭБ

PanDA — Production and Distributed Analysis). Все модули системы управления потоками заданий были отлажены и протестированы на реальных задачах эксперимента, дополнительная проверка и отладка были проведены для экспериментов ALICE и COMPASS. Эксперимент ALICE использовал данную систему в тестовом режиме для управления потоками заданий на суперкомпьютере Titan в 2017 г., эксперимент COMPASS использует данную систему для обработки данных в ЦЕРН и ОИЯИ (адаптация системы для эксперимента COMPASS была выполнена сотрудником ОИЯИ А. Ш. Петросяном).

ProdSys2 является необходимым уровнем абстракции, скрывающим от пользователя сложности работы, связанные с запуском и выполнением задач на рабочем узле. В целом работу системы обработки и анализа данных можно описать как процесс преобразования входных, обычно неструктурированных, данных эксперимента (или пользователя) в набор параметров для выполнения задач на вычислительных системах. Система имеет три уровня (DEFT, JEDI, PanDA) и отвечает всем требованиям, предъявляемым к системам управления загрузкой, подробно рассмотренным в разд. 3.

Система управляет всеми потоками заданий физического эксперимента ATLAS.

• Потоки заданий, выполняемые WMS для всего эксперимента («виртуальной организации»):

— обработка и (пере)обработка данных;

— моделирование методом Монте-Карло;

— создание приведенных данных для физического анализа;

— обработка данных для триггера высокого уровня;

— специальные случаи, например обработка «поездом» для всех или нескольких физических групп;

— проверка и валидация версий программного обеспечения.

• Потоки заданий, выполняемых физическими группами эксперимента:

— создание данных для физического анализа, проводимого физической группой;

— анализ данных.

• Потоки заданий, выполняемых отдельными пользователями (физический анализ данных).

В системе реализованы обработка «поездом» и «постоянная» обработка (рассмотренные ранее), а также специальные потоки заданий для проверки кода и валидации физических результатов. Специальным случаем является поток заданий «event index», когда для каждого обработанного события в базу данных записывается краткая информация о нем. В системе управления потоками заданий реализован механизм разделения ресурса и «всемирного облака».

Ниже приведены рисунки, на которых показана работа и производительность системы. На рис. 42 приведено количество задач для различных пото-

ков заданий, выполняющихся одновременно. Из графика видно, что система выполняет до 300 000 задач в день одновременно.

На рис. 43 показано количество задач, выполненных ежедневно с января 2016 г. по февраль 2017 г. Из графика следует, что системой выполняется до 2 млн задач в день и среднее значение выполняемых задач значительно превосходит 1 млн/день (это очень важный показатель масштабируемости си-

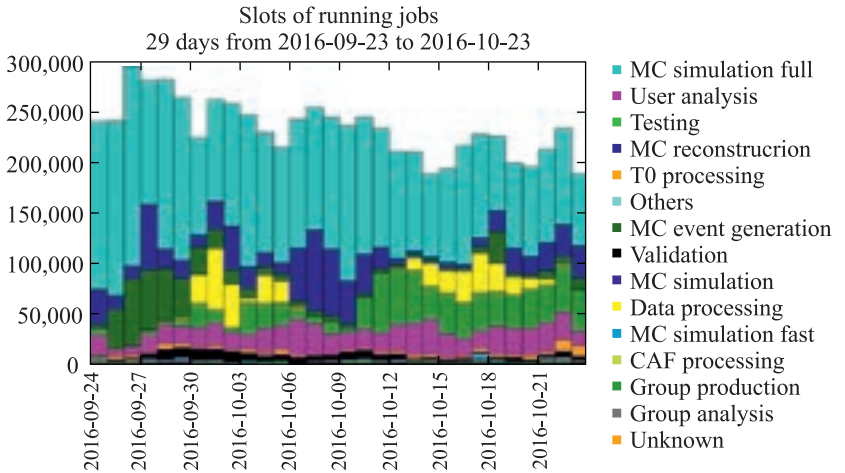

Рис. 42. Количество задач для различных потоков заданий, выполняющихся одновременно

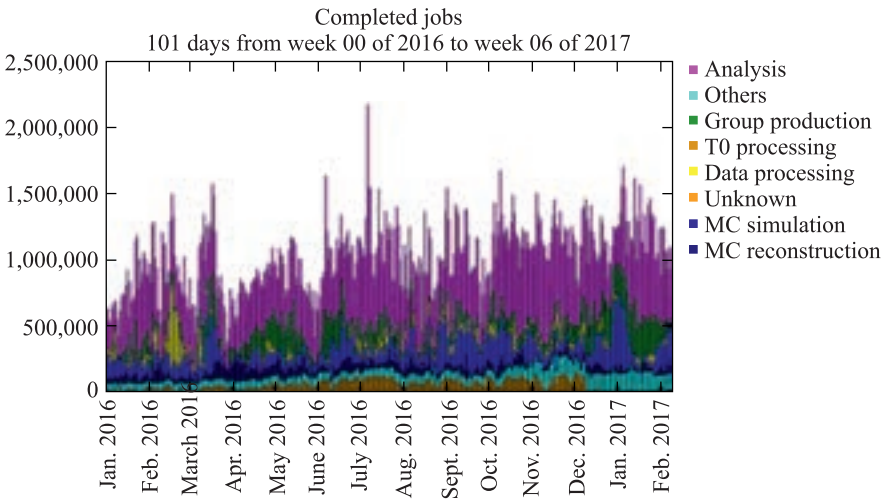

Рис. 43. Количество одновременно используемых ядер ЦПУ для обработки данных

стемы). График дает представление о разделении ресурса между различными потоками заданий.

На рис. 44 приведено количество данных, обработанных системой в течение 1 мес., а рис. 45 дает представление о том, какие вычислительные ресурсы

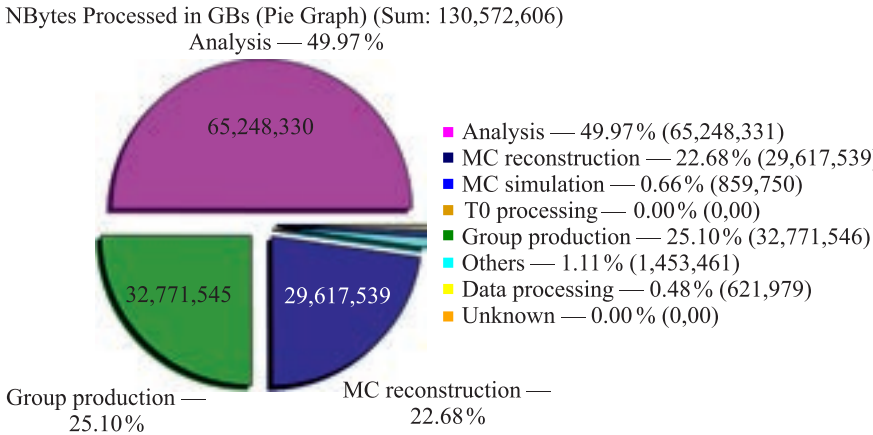

Рис. 44. Количество данных, обработанных в системе в течение 1 мес.

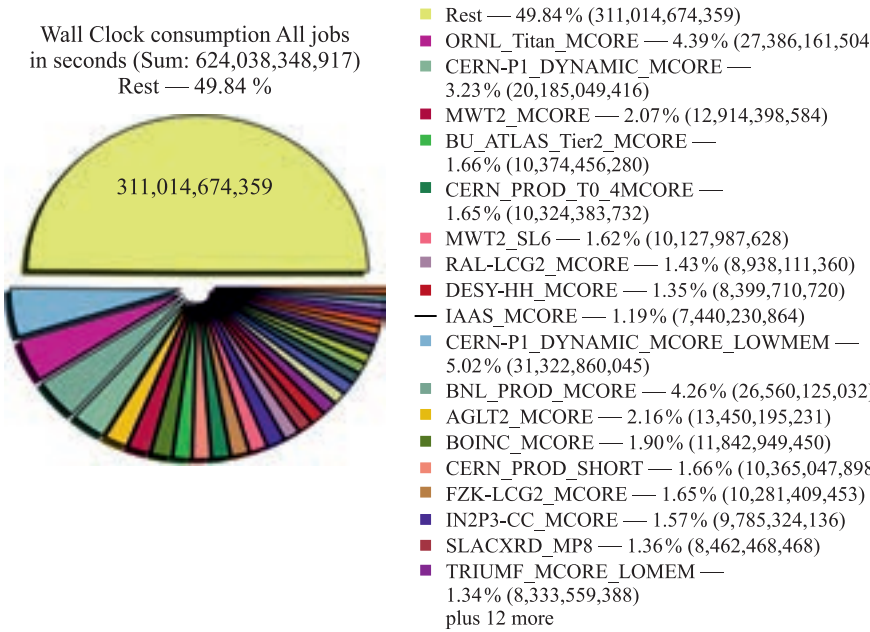

Рис. 45. Используемые вычислительные ресурсы

были использованы (из графа видно, что очереди выполнения были определены для центров грид (T1, T2), суперкомпьютеров (Titan) и ресурса облачных вычислений ЦЕРН (CERN_P1_DYNAMIC_MCORE), а также то, что управление потоками осуществлялось для ресурсов разных типов: LOMEM (менее 2 ГБ памяти на ядро), MCORE (multi-core, задача использует несколько ядер одновременно), SHORT (время выполнения задачи ограничено 24 ч) и т. д.

Созданная система является уникальной по своим параметрам и не имеет мировых аналогов. В эксперименте ATLAS система управляет до 2 млн вычислительных задач в день в гетерогенной компьютерной среде, состоящей из более чем 250 ВЦ, включая ресурсы облачных вычислений и суперкомпьютеры.

3.6. Системы мониторирования для системы распределенной обработки данных эксперимента ATLAS. ***3.6.1. Архитектурные принципы, методы и технологии при ее реализации.*** Система управления загрузкой в распределенной гетерогенной компьютерной среде представляет собой сложный и неоднородный комплекс аппаратных и программных средств. Система взаимодействует с другими системами, сервисами и базами данных распределенной инфраструктуры: а) системой управления данными, б) информационной системой, в) фабрикой пилотных заданий и многими другими. Одним из основных назначений подсистемы мониторирования научного эксперимента является отслеживание ошибок при выполнении задач обработки и анализа данных. В эксперименте ATLAS аппаратные ошибки встречаются примерно в 11 % от общего числа выполненных задач (см. рис. 46). Из-за больших объемов метаинформации, связанной с каждой задачей, поиск ошибок превращается в достаточно ресурсоемкую процедуру. Ежедневное выполнение миллиона и более задач требует высокого уровня автоматизации, постоянного мониторирования, сбора и обработки больших объемов информации

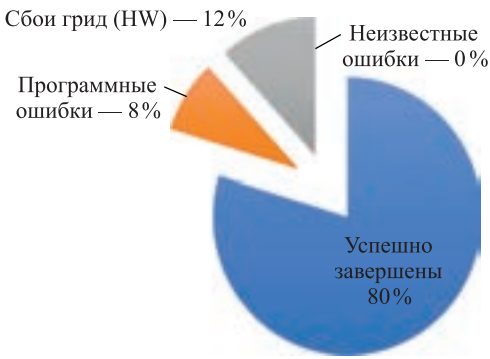

Рис. 46. Ошибки при выполнении задач в грид-инфраструктуре (в процентах)

о параметрах работы системы, возникающих ошибках, обнаружения и по возможности предсказания аномалий в работе системы управления загрузкой.

Разработка системы мониторирования являлась сложной и комплексной задачей. Необходимо было ответить на следующие вопросы:

— какие задачи должна решать система мониторирования;

— какие ресурсы она должна контролировать;

— кто является пользователем системы;

— какие время «жизни» информации (и метаинформации) в системе и объемы хранимой информации;

— какой метод мониторинга (централизованный или децентрализованный).

В первом случае сбор и интеграция информации происходят в одном узле (сервере системы мониторирования), в случае децентрализованного подхода не происходит агрегации и интеграции информации в одном месте и допускается наличие автономных агентов, самостоятельно собирающих информацию и взаимодействующих с сервером мониторинга с использованием push- или pull-модели доставки информации.

Система мониторирования должна не только давать представление о текущем состоянии и ходе выполнения потоков заданий, но и включать в себя функции аккаунтинга (accounting) для учета используемого ресурса. Метриками использования ресурса являются процессорное и астрономическое время, дисковое пространство, количество и классы выполненных задач, информация о состоянии очередей и ВЦ. Система должна как давать общее представление о состоянии обработки данных физического эксперимента, так и позволять проверять состояние и/или ход выполнения отдельной задачи.

Основные требования к системе мониторирования. Перечислим основные требования, предъявляемые к системе мониторирования:

— универсальность (одна и та же система должна предоставлять информацию обо всех потоках заданий для обработки данных, выполняемых в рамках физического эксперимента);

— производительность (производительность системы должна обеспечивать мониторинг всех компонентов системы управления загрузкой в реальном времени, допустимое время отклика должно составлять секунды);

— модульность (уровни доступа к данным и визуализации данных должны быть разделены, при введении новых типов потоков обработки данных, новых вычислительных ресурсов, например добровольных ресурсов суперкомпьютерных центров, система должна иметь возможность быть расширенной без потери производительности и универсальности за счет добавления новых функциональных возможностей и сбора данных от новых источников информации);

— масштабируемость (время отклика системы мониторирования не должно деградировать при росте информации);

— доступность и стандартизация доступа к информации (система должна иметь развитые веб-сервисы, пользовательский интерфейс и интерфейс программирования приложений (API), в которых должны быть реализованы основные сервисы для мониторинга и управления объектами системы, соответствующие веб-сервисы должны обеспечивать аутентификацию с помощью специальных цифровых ключей для разграничения доступа к системе и журналирования действий пользователей, система должна иметь механизм отложенного выполнения запросов);

— целостность (система должна предоставлять общую картину управления потоком заданий, а также детальную информацию о выполнении каждого запроса, задания, задачи с возможностью исследования причины сбоя);

— анализ информации и автоматизация (система должна иметь высокий уровень автоматизации и возможность анализировать имеющуюся информацию, в частности ошибки в ее работе, что позволило бы сформулировать правила и применить, например, алгоритмы «машинного обучения» для обнаружения аномалий и/или сбоев в работе системы управления загрузкой);

— мобильность (выбор используемых технологий должен быть таким, чтобы система могла быть использована различными «виртуальными организациями»).

Уровни и функции системы мониторирования. Система для распределенной обработки данных является сложным объектом, пользователями которой являются несколько различных групп.

• «Виртуальная организация»:

— ВО в целом (физический эксперимент), физические группы, работающие над отдельными темами научной программы эксперимента, физические группы в университетах и лабораториях, входящих в эксперимент, отдельные физики;

— сотрудники ВО, отвечающие за каждодневную работу систем распределенного компьютинга (включая системы управления данными и потоками заданий);

— сменные «операторы» и «службы поддержки» пользователей (в зависимости от размера и сложности физического эксперимента операторская служба может функционировать в режиме 24/7 в периоды работы ускорителя и основных кампаний по (пере)обработке или моделированию данных и в режиме 8/5 в остальное время).

• Компьютерные специалисты, системные администраторы и службы эксплуатации ВЦ, предоставляющих компьютерные ресурсы физическому эксперименту.

• Финансирующие организации (как правило, для данной группы наиболее интересной является информация об использовании вычислительного ресурса, т. е. аккаунтинг).

Можно выделить следующие уровни системы мониторирования.

• Мониторинг работы вычислительных ресурсов:

— стабильность работы, ошибки выполнения задач, типы ошибок (одним из типичных примеров могут служить сбои в системе хранения данных);

— производительность вычислительных центров (количество выполняемых заданий и задач и их типов, количество обработанных событий, количество полученных и/или переданных для/после обработки данных).

• Мониторинг вычислительной инфраструктуры системы управления потоками заданий, включая коммуникацию с внешними системами: системой управления данными, ИС и др. (сбой в работе системы управления данными приводит к остановке определения новых потоков заданий и задержке выполнения текущих заданий).

• Мониторинг потоков заданий:

— по группам пользователей (всего эксперимента в целом, отдельных физических групп, отдельных ученых);

— по классам потоков заданий (обработка данных, анализ данных, моделирование методом Монте-Карло . . .);

— по выделенным квотам, долям, приоритетам для различных классов заданий.

• Мониторинг хода выполнения цепочки заданий и/или запросов (например, запрос может быть одобрен/отвергнут или его исполнение может быть отложено лицом, отвечающим за физическую программу эксперимента).

• Мониторинг работы групп сайтов, выбранных системой управления потоками заданий для выполнения запроса/задания/цепочки заданий.

3.6.2. Реализация системы мониторирования для системы ProdSys2–PanDA эксперимента ATLAS на Большом адронном коллайдере и за его пределами. Основной задачей при реализации системы мониторирования стало быстрое обнаружение ошибок и мониторирование хода выполнения задач для различных классов потоков данных под управлением системы mega-PanDA (ProdSys2–PanDA). Одна из первых задач, которую необходимо было решить, — это популярность данных в зависимости от времени. За время жизни эксперимента ATLAS общий архив информации о выполненных задачах, заданиях и запросах содержал информацию о ходе выполнения более чем 10 млн заданий и около 3000 млн задач, включая информацию о месте выполнения, ПО, ошибках, количестве повторов задач и т. д. Необходимо было решить задачу о способе хранения данных и метаданных, обеспечить доступ к ним для приложений аналитического анализа информации и/или экспертных систем и одновременно бесперебойную работу системы мониторирования. Вся информация была разделена на 3 группы.

• *Текущая информация.* В эту группу входит информация о задачах и заданиях, выполняемых в системе, ждущих выполнения и/или выполненных в течение последних 3 мес. Для информации этой группы характерны запросы

на модификацию содержимого как изменения приоритета задания/задачи и превентивная остановка и/или перезапуск задачи.

• *Среднесрочная информация.* Это информация о работе в период от 3 до 6 мес. Задачи этой группы часто составляют часть выполняемой цепочки заданий, и доступ к ним в режиме RO (read only) происходит до 100 раз в сутки.

• *Архив системы.* Это информация о потоках заданий (включая всю информацию об отдельных задачах), выполненных 6 мес. назад или раньше.

Выбор шага в 3 мес. был сделан на основе количества запросов к информации, хранящейся в БД. Так, все, что касалось выполнения заданий старше этого периода, использовалось только для проведения аналитических исследований по производительности работы системы управления загрузкой, классификации типов сбоев или для составления отчетов об использовании вычислительных ресурсов. Текущая информация, наоборот, пользовалась наибольшей популярностью у всех пользователей. Все пользователи системы по своим «поведенческим» характеристикам были разделены на 4 группы:

— «системные администраторы» (компьютерные специалисты и системные администраторы ВЦ, предоставляющих вычислительные мощности);

— «операторы» (сменные операторы и служба поддержки первого уровня, следящие за выполнением потоков заданий);

— «физики» (участники международного сотрудничества ATLAS, проводящие анализ данных и запускающие задания для их анализа);

— «менеджеры» (участники международного сотрудничества ATLAS, формирующие потоки для различных классов заданий: (пере)обработки данных, моделирования методом Монте-Карло, потоков заданий для физических групп, обработки данных для триггера высокого уровня, проверки и валидации базового ПО эксперимента);

— «координаторы» (участники международного сотрудничества ATLAS, координирующие физическую программу эксперимента или проекты в SW&C, также эта группа отвечает за предоставление регулярных отчетов для проверяющих и финансирующих организаций);

— «эксперты» (разработчики ПО системы управления загрузкой, БД, исследующие производительность ПО и ошибки в его исполнении и/или в случае возникновения аномалий в работе всей системы).

Такое распределение дало ответ на один из фундаментальных вопросов при создании системы мониторирования: кто является пользователем системы? Потребовалось создание такого набора функций, который позволит иметь как страницы, дающие представление о работе системы в целом и глобальном использовании вычислительного ресурса, так и детальную информацию о выполнении задачи «физика». Анализ журнальной информации о доступе к системе показывает, что наиболее быстро нужную информацию получают представители групп «эксперты» и «операторы». Скорость получения

информации представителями других групп существенно зависит от опыта работы с системой, поэтому был выбран подход интуитивного поиска, а также «моя любимая страница», когда система «запоминала» предпочтения пользователя и на первом этапе работы предлагала выбор из наиболее посещаемых страниц.

Основными принципами в реализации были производительность, надежность, простота. Была выбрана централизованная модель. Рассмотрим некоторые из выбранных технических решений.

Выбор технологии для хранения и предоставления информации. Разделение информации по временной шкале давало возможность реализовать хранение данных с выбором различных технологий. Хранилище могло быть как гомогенным (СУБД), так и гетерогенным (комбинация СУБД_NoSQL). Сравнение возможных решений для технологий NoSQL подробно обсуждалось экспертами ЦЕРН, НИЦ КИ, DESY и ОИЯИ [65]. Были рассмотрены вопросы масштабируемости и производительности и было показано, что при контроле за размером «текущего компонента» информации 400 ГБ и общем размере информации менее 100 ТБ (в настоящее время размер базы данных составляет 30 ТБ) использование СУБД ORACLE дает лучшие характеристики при доступе к информации, при этом классы индексов и организация таблиц могут быть различными для трех групп, например «архив» хранится в «сжатом» виде и имеет внутреннее разделение по годам. Важным фактором выбора технологии SQL или NoSQL служил фактор «мобильности» в предположении, что вся система управления загрузкой и ее мониторинг будут в будущем использоваться за пределами ATLAS, а также ФВЭ и ЯФ, и использование СУБД ORACLE позволит другим экспериментам использовать аналогичную технологию, например MySQL (следует признать, что это решение оказалось правильным и эксперименты AMS и BlueBrain используют именно его). Выбор технологии ORACLE (а не MySQL) также диктовался решением группы разработчиков системы управления данными эксперимента и отделения ИТ ЦЕРН, получена лицензия на ее использование безвозмездно для всех университетов и лабораторий, входящих в ATLAS (подробно исследование вопроса о сравнении технологий хранения данных отражено в работе, написанной в соавторстве с сотрудниками Лаборатории технологии больших данных НИЦ КИ [76]). Выбор остальных технологий был менее драматичен, и были использованы технологии, широко применяемые для предоставления и визуализации информации в ИТ-компаниях с объемами информации и типом доступа, сопоставимыми с требованиями к системе мониторирования (Instagram, Mozilla, YouTube, Google), например технология фреймворка django для веб-приложений и шаблон проектирования MVC (Model View Controller) [62, 66, 67]. Алгоритмы обработки информации были разделены между уровнями frontend и базой данных (с использованием большого набора функций и возможностей ORACLE), генерация графиков выполняется на сто-

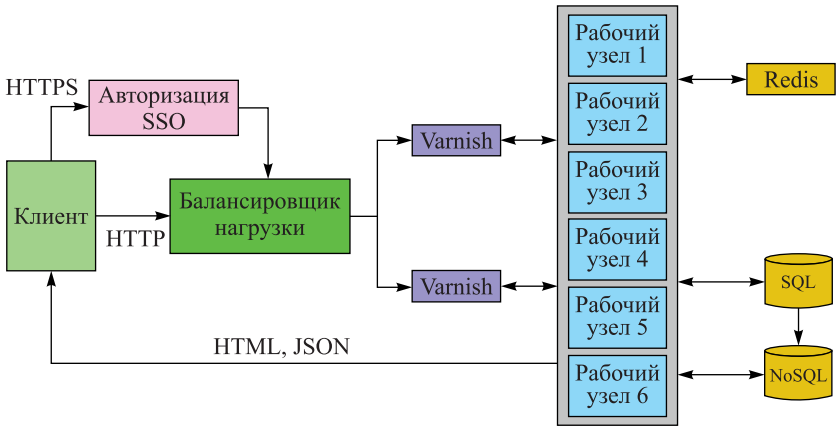

Рис. 47. Архитектура и основные компоненты инфраструктуры системы мониторирования

роне клиента с использованием библиотек ds.j3 (data driven documents) [78], технология ajax используется для обмена данными между браузером и веб-сервером. Для аутентификации пользователей был использован пакет SSO (Single Sign-On Management), предоставляемый ЦЕРН. Схематично архитектура и основные компоненты инфраструктуры системы мониторирования представлены на рис. 47.

Созданная система [68, 69] отвечает всем требованиям и имеет высокую производительность, например, генерация информации о выполнении задач системой управления загрузкой (с информацией об ошибках) за последние

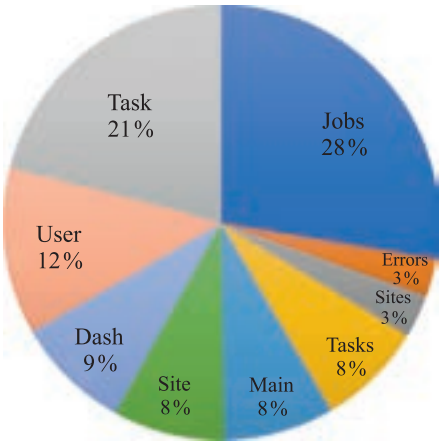

Рис. 48. Статистика обращений к системе мониторирования в течение календарного года

12 ч и для $O(1\text{M})$ задач занимает 10 с [71, 72]. На рис. 48 представлена статистика обращений к системе мониторирования в течение календарного года работы.

Количество ежедневных обращений составляет в среднем 13 000 от 1500 индивидуальных пользователей. Из приведенной статистики видно, что наиболее популярной является информация о заданиях (группах заданий task, tasks) и задачах (jobs). Многие иллюстрации выполнены с использованием графических возможностей системы мониторирования (в разработку системы большой вклад внесли сотрудники Брукхейвенской национальной лаборатории (BNL) Т. Венаус и С. Подольский и UNAB Т. Корчуганова и А. Алексеев).

Итак, в разд. 3 рассмотрена архитектура системы распределенной обработки данных и методика распределения вычислительного ресурса между различными классами заданий. Система управления потоками заданий едина для всего физического эксперимента, отдельных групп, работающих по определенной тематике, и индивидуальных пользователей. Система имеет квоты и приоритеты, а также позволяет разделять с высокой гранулярностью вычислительный ресурс между различными потоками заданий. Реализация системы и демонстрация ее работы для эксперимента ATLAS показали высокую степень ее масштабируемости, надежности и управляемости. Система не имеет мировых аналогов по этим показателям. Она имеет развитую подсистему мониторирования и аккаунтинга, позволяющую собирать и предоставлять информацию не только о ходе обработки и анализа данных, но и о работе сайтов в рамках гетерогенной инфраструктуры.

Система позволяет одновременно выполнять более 500 000 задач на всех имеющихся типах ресурсов (высокопропускные центры (грид), ресурсы облачных вычислений, суперкомпьютерные центры — более 250 центров по всему миру), более 2 млн задач в день и более 30 млн задач ежемесячно.

4. ДАЛЬНЕЙШЕЕ РАЗВИТИЕ КОМПЬЮТЕРНОЙ МОДЕЛИ. ИНТЕГРАЦИЯ СУПЕРКОМПЬЮТЕРОВ И РЕСУРСОВ ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ С РАСПРЕДЕЛЕННЫМИ ВЫЧИСЛИТЕЛЬНЫМИ РЕСУРСАМИ ГРИД

В предыдущих разделах были рассмотрены роль суперкомпьютеров для приложений в области физики частиц и создание динамической системы управления потоками заданий, обоснована идея интегрированной связки HPC–HTC, а также причины перехода от иерархической модели MONARC к «смешанной компьютерной модели», а затем к динамическому управлению ресурсами и созданию «всемирного облака» для выполнения потоков заданий. Эти работы сделали возможным дальнейшее развитие компьютерной модели для приложений в области физики частиц и переход к использованию гетерогенных вычислительных ресурсов. В рамках вопросов интеграции суперкомпью-

теров и ресурсов облачных вычислений с распределенными вычислительными ресурсами грид рассмотрим дополнительные требования к распределенной системе обработки данных при использовании гетерогенных вычислительных ресурсов. Проведем анализ технологий, позволяющих перейти к созданию федерации географически распределенных дисковых ресурсов и дальнейшему развитию компьютерной модели для экспериментов в области физики частиц.

Фундаментальным вопросом для развития компьютерной модели в области физики частиц является такой вопрос: как данные будут обрабатываться, анализироваться и моделироваться через 7–10 лет? При ответе на этот вопрос необходимо учитывать ограничения бюджета практически во всех странах, увеличение вычислительных мощностей для экспериментов на LHC и существующие бюджеты для новых комплексов (NICA, FAIR) и проектов (LSST, DUNE). До последнего времени компьютерная модель строилась в предположении, что эксперименты в области ФВЭ и ЯФ являются «собственниками» вычислительного ресурса. Работы многих групп в разных странах в последние годы были направлены на то, чтобы показать, как вычислительная инфраструктура, не принадлежащая экспериментам и/или ассоциированным с ними ВЦ, может быть эффективно использована и интегрирована с системой распределенных вычислений грид. Вариантами ответа на поставленный вопрос могут быть следующие.

• Для экспериментов в области ФВЭ и ЯФ будут продолжать покупать необходимое аппаратное обеспечение и расширять компьютерную инфраструктуру:

— очевидное преимущество — это преимущество «собственника» ресурса, который может быть использован и доступен в любой момент;

— данное преимущество надо учитывать только в случае, если есть достаточный ресурс во время максимальной загрузки (кампании анализа и переобработки данных), в остальное время вычислительный ресурс не будет использован в полном объеме.

• Для экспериментов в области ФВЭ и ЯФ будут покупать мощности у тех, кто их предоставляет на коммерческой основе:

— преимущество такого подхода состоит в том, что капитальные затраты несет третья сторона;

— недостатком является отсутствие гарантий, что ресурс будет доступен в требуемом объеме или будет доступен для использования, когда это потребуется, а также необходимость «доверия» третьей стороне и предоставления ей доступа к данным международной коллаборации.

Компромиссным является вариант, когда базовые ресурсы принадлежат экспериментам, а в период максимальной нагрузки «используются» поставщики вычислительных услуг и сервисов.

Ландшафт современных вычислительных ресурсов и потребности в них драматически отличаются от ситуации 10-летней давности, когда приложе-

ния в области ФВЭ и ЯФ были одним из основных «потребителей» вычислительных мощностей в глобальном мире (за исключением приложений, связанных с военной тематикой и исследованием климата). Долгие годы закупки компьютеров и расчет необходимой мощности ВЦ осуществлялись с ориентиром на среднюю загрузку и одновременное выполнение всех потоков заданий (рис. 49). В настоящее время существует больший пул вычислительных ресурсов за пределами ФВЭ и ЯФ. В первую очередь, это коммерческие ресурсы и суперкомпьютерные центры. Так, вычислительный ресурс гигантов ИТ-индустрии Yandex, Google, Amazon, Microsoft в сотни раз превышает мощности консорциума WLCG (уже в 2018 г. ресурс третьего по мощности суперкомпьютера Titan превышал весь ресурс WLCG для эксперимента ATLAS), что требует пересмотра «усредненного» подхода к использованию вычислительных мощностей и смены модели с ориентацией на максимальное использование вычислительного ресурса в периоды пиковой нагрузки и соответствующего планирования потоков заданий (рис. 50). При таком сценарии классы потоков заданий могут быть переориентированы соответственно, так как переобработка данных и моделирование методом Монте-Карло, как правило, имеют название «кампания» и должны быть закончены как можно быстрее, в то же время обработка и физический анализ данных имеют хорошо выраженную постоянную составляющую.

Ориентация на пиковые нагрузки предъявляет повышенные требования к системам распределенной обработки данных. Они должны быть готовы увеличить количество задач в сотни раз, быстро «захватывать» доступные ресурсы и уметь быстро их освобождать. Одновременно требования предъявляются и к базовому ПО (программам реконструкции событий, восстановления треков частиц и т. д.) физических экспериментов (помимо обозначенных в разд. 2). При использовании коммерческого ресурса любые «бесконечные» петли в коде, огромные журнальные файлы (и, соответственно, дисковое про-

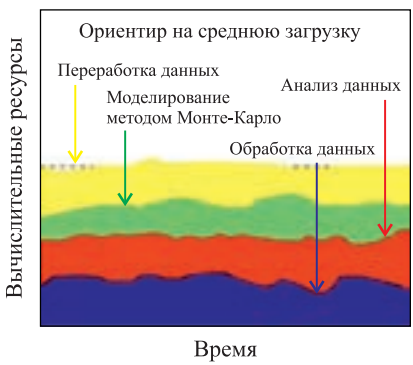

Рис. 49. Использование вычислительного ресурса при сценарии «средняя загрузка»

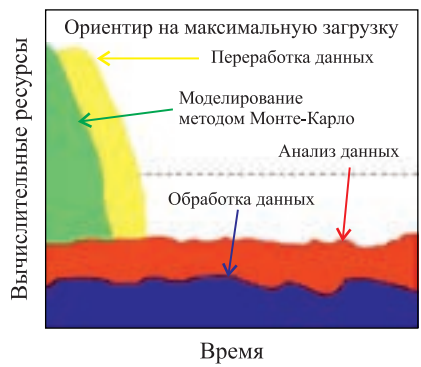

Рис. 50. Использование вычислительного ресурса при сценарии «пиковая загрузка»

странство и время передачи, необходимые для их хранения и передачи), неэффективное использование процессорного времени становятся настоящим расточительством. Возможно, необходимо было обратить на это внимание и раньше, но при новом сценарии такая неэффективность приведет к заметным финансовым затратам.

4.1. Интеграция ресурсов облачных вычислений и грид. Идея облачных вычислительных ресурсов не нова, в 1961 г. пионер ИТ Джон Маккарти (более известный как человек, введший в обиход выражение «искусственный интеллект» — artificial intelligence) предсказал, что «вычисления могут быть организованы как общедоступный сервис» (программа-утилита), и он продолжал размышлять и описывать, как это может быть реализовано. Идея о том, что вычисления проводятся не на локальных ресурсах, а на централизованных вычислительных мощностях, предоставляемых и поддерживаемых некоторой третьей стороной, получила новое развитие и реализацию в последние десятилетия.

Развитие коммерческих и академических ресурсов облачных вычислений и применение их для приложений в области физики частиц началось сравнительно недавно в экспериментах в области ФВЭ, ЯФ и астрофизики и совпало по времени с необходимостью обработки и анализа десятков петабайт данных, с пониманием ограничений грид и стоимости поддержания существующих центров и создания новых. В эти годы Amazon, Google, Yandex, Microsoft создали действующие центры, состоящие из сотен тысяч компьютеров. Существуют разные определения ресурсов облачных вычислений, остановимся на определении, данном одним из «отцов» грид Яном Фостером: «Широкомасштабная распределенная компьютерная парадигма, обусловленная экономией затрат, в которой пул абстрактных, виртуальных, динамически масштабируемых управляемых вычислительных мощностей, хранилищ, платформ и сервисов предоставляется по требованию внешним клиентам через интернет» [70]. В определении, данном Фостером, и его статье не противопоставляются понятия грид и «облачные вычисления» (cloud computing), а наоборот, подчеркивается общность между различными подходами в архитектуре, технологии и взглядах на организацию распределенных вычислений. Фостер предсказал, что необходимо найти пути и решения, чтобы определить, как будет эволюционировать общая инфраструктура, но он думал, что поиск таких путей и решений займет гораздо больше времени и следующие пять лет приведут к созданию реальных прототипов и прообразов будущих инфраструктур.

Для научного сообщества стал откровением доклад проф. Мартина Севиора из Мельбурнского университета на симпозиуме ISGC (International Symposium for Grid and Clouds) в 2009 г. об использовании ресурсов компании Amazon для проведения моделирования методом Монте-Карло в эксперименте Belle [71]. В докладе была приведена статистика о «стоимости» гене-

рации 0,85 млн событий с использованием коммерческих ресурсов, «цена» одного события составила 0,53 доллара США, что дешевле понесенных центром уровня T2 в Мельбурне (с учетом накладных) расходов. Интерес к теме доклада привел к дальнейшему развитию работ в данном направлении, потому что кампания по моделированию событий в эксперименте Belle была кратковременной (неделю) и использовала сравнительно небольшие вычислительные мощности (20 машин архитектуры HighCPU-XL: 8 ядер, 17 ГБ RAM). Для проведения полного сравнения эффективности использования коммерческого ресурса облачных вычислений и грид необходимо было получить ответы на следующие вопросы:

— насколько стабильно будет работать коммерческий ресурс облачных вычислений в течение продолжительного времени (месяцы), включая передачу и хранение данных;

— насколько стабильно будет работать коммерческий ресурс облачных вычислений, если размер ресурса сравним с размером центра уровня T2;

— насколько легко и прозрачно ресурсы могут быть интегрированы с системой управления заданиями;

— сколько это будет стоить.

С самого начала данное исследование рассматривалось с практической точки зрения — для применения в будущем в больших экспериментах (класса мегасайенс). Решение о проведении исследования совпало с желанием компании Google дать доступ к GCE (Google Compute Engine) [72, 92]. Группой разработчиков эксперимента ATLAS и представителями компании Google была достигнута договоренность о предоставлении 5 млн ЦПУ-часов (это соответствовало примерно 4000 ядер в течение 2 мес.). Вычислительный ресурс был интегрирован с системой управления потоками заданий и описан в ИС (тип рабочей очереди, доступность ресурса, формализованное описание сайта).

Система состояла из нескольких интегрированных вместе компонентов. В состав системы входили: HTCondor [73] (использовался как для отправки и развертывания виртуальной машины (VM) на облачной платформе, так и для локальной (внутри VM) системы пакетной обработки, промежуточное программное обеспечение рабочего узла (для авторизации и передачи файлов), сетевая файловая система CVMFS [74] (для хранения и доступа к программному обеспечению физического эксперимента) и фабрика пилотных заданий APF (AutoPyFactory). Также был использован пакет виртуализации, разработанный в ЦЕРН (CERNVM [75]). Схематично процесс запуска задания показан на рис. 51 (значение и функции фабрики пилотных заданий и очередей подробно рассмотрены в разд. 3). Результаты исследования могут быть суммированы следующим образом:

— планирование, создание и настройка системы, включая описание очереди в ИС, заняли 2 недели;

Рис. 51. Схема запуска задания в облачном вычислительном ресурсе

— вычислительный ресурс использовался в течение 8 недель (10 недель, включая настройку);

— около 458 000 задач было выполнено, произведено/обработано 214 млн событий (максимально достигнутое использование составило 15 000 задач/день; число сбоев составило 6 %, что меньше, чем в среднем по грид);

— работа предоставленного коммерческого ресурса была стабильна и надежна (по окончании работы все результаты были переданы в один из центров уровня T1 эксперимента ATLAS).

На графике (рис. 52) показан ход выполнения задач ATLAS на ресурсах GCE по дням. Зеленым отмечены успешные задачи, розовым — задачи, имевшие ошибки. Данная работа была пионерской. Впервые была показана возможность масштабного использования коммерческого ресурса облачных вычислений для приложений в области ФВЭ и ЯФ. Такая интеграция стала возможна только при наличии системы управления загрузкой, которая была разработана и реализована согласно авторским методам и подходам, а также новой компьютерной модели.

Дальнейшее развитие облачных вычислений привело не только к их широкому использованию, но и к расширению классов заданий для них. За последние годы (2015–2019) были интегрированы академические ресурсы облачных вычислений в Канаде и Австралии, а также ресурсы компании Amazon.

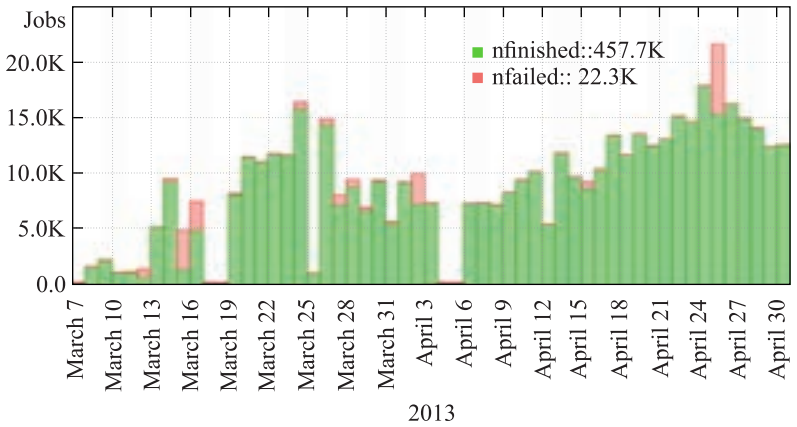

Рис. 52 (цветной в электронной версии). Количество задач ATLAS, выполненных на ресурсах Google Compute Engine

На рис. 53 показано еженедельное количество задач, выполненных только на ресурсах облачных вычислений за 2,5 года (2014–2016 гг.). Из графика видно, что до 400 000 задач выполняется еженедельно. Среди потоков заданий доминируют задачи моделирования методом Монте-Карло (MC simulation), в то же время существуют периоды, когда задачи анализа (analysis) активно используют облачный ресурс.

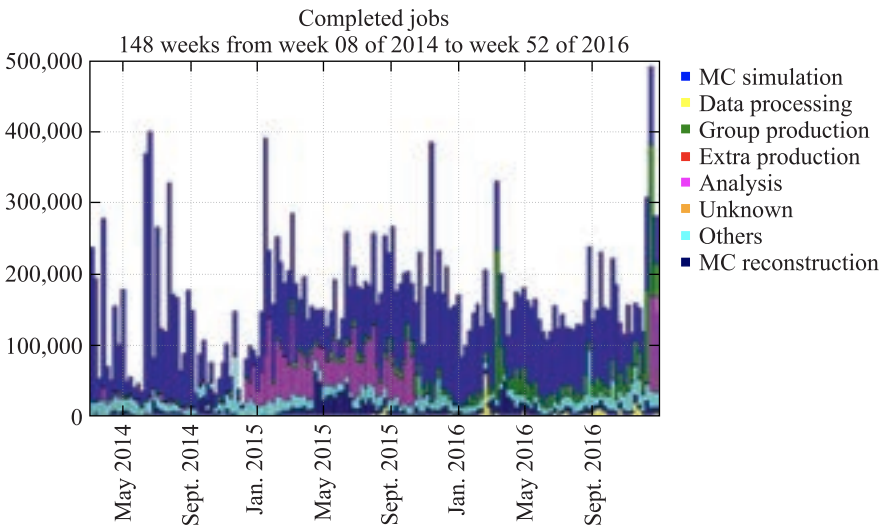

Рис. 53. Еженедельное количество задач, выполненных только на ресурсах облачных вычислений в 2014–2016 гг.

Следует отметить, что коммерческие ресурсы все еще дороже на 20–30 %, чем ресурсы WLCG [78]. Если рассматривать существующий прайс-лист, то передача 1 ПБ данных из хранилища Google (GCS — Google Cloud Storage) в центр WLCG может стоить до 50 тыс. долларов США. В случае интеграции коммерческих ресурсов важно было начать исследования по созданию гетерогенной компьютерной среды и выхода за пределы WLCG, показав возможности системы управления загрузкой для нового класса ресурсов и их интеграции с распределенной системой вычислений грид.

4.2. Интеграция суперкомпьютеров и грид. В разд. 2 была подробно рассмотрена роль приложений в области ФВЭ и ЯФ для суперкомпьютерных центров и возможная роль суперкомпьютеров для научной программы физики частиц. Мы также обсудили, чем был мотивирован выбор распределенной компьютерной модели для экспериментов в области физики частиц. В настоящее время практически все эксперименты на новых и строящихся машинах используют и планируют применять распределенную модель компьютинга. Потребности в дополнительных вычислительных мощностях и создание системы управления заданиями ProdSys2–megaPanDA позволили разработать методы интеграции суперкомпьютеров с грид и использовать мощности СК для научных приложений в области физики частиц. Одним из основных аргументов является факт сравнения мощности WLCG и LCF Titan (LCF — Leadership Class Facilities):

— WLCG ATLAS: $220\,000 \times 86$ вычислительных ядер;

— Titan: $300\,000 \times 86$ вычислительных ядер и 18 000 графических процессоров.

Есть взаимный интерес сторон к совместной работе, потому что научная программа международных сотрудничеств ATLAS и ALICE, а также возможные результаты исследований и потенциальных открытий представляют интерес для суперкомпьютерных центров, а разработанная система управления загрузкой открывает новые возможности для управления потоками заданий в суперкомпьютерных центрах (более подробно этот вопрос рассмотрен в разд. 2). Эволюция аппаратной базы суперкомпьютеров по материалам суперкомпьютерных конференций [79, 80] позволяет сделать два важных вывода:

— ко второму десятилетию XXI в. количество аппаратных решений значительно сократилось и фактически свелось к трем архитектурам. Многие СК из первого десятка топ-100 аппаратно совместимы с вычислительными мощностями WLCG;

— 12 первых машин из списка топ-500 обладают половиной мощности от всех компьютеров списка.

Таким образом, наиболее прагматичным подходом было попробовать разработать не только универсальное решение, но и использовать СК из первого десятка топ-100 для его демонстрации. Необходимо было ответить на следующие фундаментальные вопросы:

— Как получить «время» на суперкомпьютерах?

— Как интегрировать суперкомпьютеры с инфраструктурой WLCG и системой распределенных вычислений, принятой экспериментами в области ФВЭ и ЯФ?

— Как выполнять программный код экспериментов в области ФВЭ и ЯФ на СК и как делать это эффективно?

4.2.1. Выделение ресурса на суперкомпьютерах. Выделение ресурса на СК — это та область, в которой существует очень высокая конкуренция между научными и техническими предложениями высокого класса для различных областей знаний: биоинформатики, квантовой хромодинамики, исследований климата, моделирования в астрофизике и астрономии и т. д. Распределение ресурса происходит после рассмотрения проектов, предложенных сравнительно небольшими (по меркам экспериментов в области ФВЭ и ЯФ) группами ученых. Многие группы имеют длительную историю по использованию и экспертизе СК. Этот подход принципиально отличается от политики WLCG, в которой распределение ресурса происходит между виртуальными организациями (экспериментами). Так, коллаборация ATLAS не может участвовать в конкурсе на вычислительные мощности СК Titan или СК НИЦ КИ. Проект должен быть представлен группой ученых и поддержан соответствующим экспертным советом. Как и в случае с GCE, интерес состоял не в разовом действии, а в интеграции ресурса СК на длительной основе. И оказалось, что интерес может быть обоюдным (вернемся к обсуждению классической системы выделения ресурса СК позже).

Классическая карта использования LCF показана на рис. 54. Возможно, это был не лучший день работы конкретной машины, так как она была заполнена задачами только на 85 % (заполненные узлы отмечены цветом, свободные узлы отмечены белым). Из карты видно, что наибольший свободный раздел может предоставить «только» 1024 узла. Видимо, задачи, ожидающие своего выполнения, требовали в этот момент большего количества узлов, поэтому на короткий срок (а нам будет необходимо определить, что означает «короткий срок») возможно использование этих узлов. Среднегодичная занятость LCF Titan составляет 90 %, т. е. «свободный» ресурс составляет около 300 млн ЦПУ-часов в год. В отличие от НТС «занятость» СК не всегда предполагает 100%-ю загрузку (безусловно, это зависит от типов задач, выполняемых на СК). В то же время типичные приложения в области физики частиц, такие как Geant4 (G4 [81]) или ROOT [57]), являются прекрасными кандидатами. Эти приложения могут использовать временно свободные узлы СК. Предложение о выполнении задач ФВЭ и ЯФ на СК в фоновом режиме и тем самым повышение общей эффективности использования СК заинтересовало сразу несколько центров в России, США и Чехии. Для отладки данной методики и интеграции СК с WLCG были выделены квоты в 1 млн ЦПУ-часов в год в предположении, что основное время будет получено за счет работы

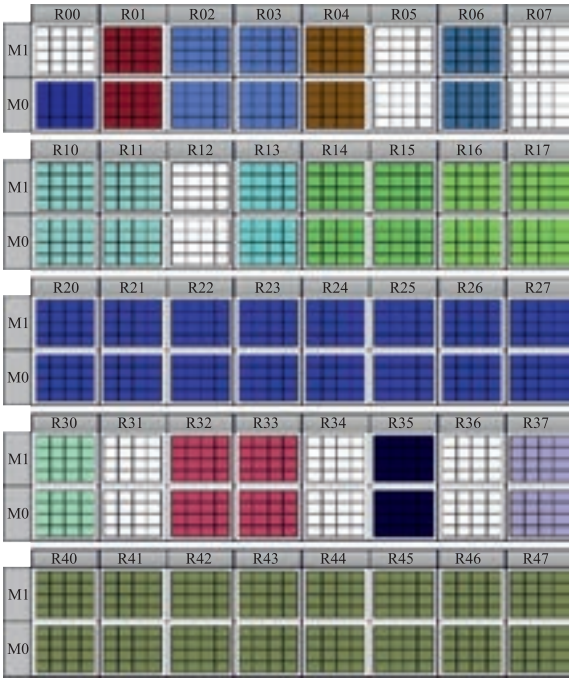

Рис. 54 (цветной в электронной версии). Карта использования процессоров суперкомпьютера класса LCF

в фоновом режиме, а квота будет гарантировать время, необходимое для отладки и проверки предложенной методики. Так был найден ответ на первый фундаментальный вопрос, и группе эксперимента ATLAS удалось вступить в «клуб» пользователей СК.

4.2.2. Интеграция суперкомпьютеров с инфраструктурой WLCG и системой распределенных вычислений. Рассмотрим некоторые особенности суперкомпьютеров и то, как они могут влиять на процесс интеграции ресурсов HPC и HTC.

• Каждый суперкомпьютер уникален:

— уникальная архитектура и оборудование (специализированная операционная система, рабочие узлы с ограниченной оперативной памятью на рабочий узел; архитектура может отличаться от принятой в WLCG Intel $\times$ 86, в таком случае требуется кросскомпиляция кода);

— есть еще более сложный случай с использованием графических процессоров, который здесь не рассматривается.

• Уникальная система запуска задач в СК (с ограничением числа задач в очереди, запускаемых одним пользователем).

• Уникальная система безопасности:

— однократная интерактивная аутентификация с помощью пароля;

— отсутствие связи рабочих узлов с интернет (программа пилотных заданий не может работать на рабочем узле, так как не будет иметь связи с сервером). Единственный из узлов, имеющий связь с «внешним» миром, это входной узел (login node).

• Высокая конкуренция в распределении времени между проектами с ориентацией на проекты демонстрации «превосходства» в различных областях науки.

Следует отметить, что для СК, которые могут быть рассмотрены как HPC-кластер и могут характеризоваться наличием $N \times 86$ ядер, рабочие узлы имеют связь TCP/IP с внешним миром. Вопрос интеграции с ресурсами HTC не так сложен, СК такого типа должны быть правильно описаны в ИС, но с точки зрения системы управления загрузкой могут рассматриваться как сайт инфраструктуры грид без дискового ресурса, а результаты вычислений должны быть переданы в место постоянного хранения. Использование таких машин и их интеграция имеют, скорее, значение для операторских служб и не предъявляют дополнительных требований к системе управления потоками заданий.

Теоретическое обоснование и возможные подходы эффективной интеграции ресурсов суперкомпьютеров (HPC) и распределенных высокопропускных ресурсов (HTC) грид были рассмотрены в разд. 2. Проблема интеграции имеет более общий характер, чем просто применение суперкомпьютеров для обработки данных LHC (или экспериментов в области физики частиц и результатов наблюдений в астрономии).

С точки зрения крупных суперкомпьютерных центров основным является вопрос, как наилучшим образом интегрировать рабочие нагрузки с большими требованиями к вычислительному ресурсу, например традиционные рабочие нагрузки для СК-центров (климат, биоинформатика, расчет на решетках КХД), с большими объемами рабочей нагрузки, возникающими, например, при работе с экспериментальными данными и данными астрономических наблюдений. Для такой интеграции, в первую очередь, необходима система управления загрузкой (потоками заданий). Модульная структура системы управления потоками заданий оказалась применима для интеграции ресурсов HPC и HTC. На первом этапе необходимо было исследовать, насколько предположение о «кратковременном» наличии свободных узлов справедливо, и понять, могут ли узлы эффективно использоваться для приложений в области ФВЭ и ЯФ. На рис. 55 представлен двумерный график, показывающий корреляцию между количеством свободных узлов и длительностью временного интервала, в течение которого они были свободны (измерения проводились для LCF Titan в течение 1 мес., было сделано более 62 500 измерений). Из графика видно, что в среднем свободны: 691 рабочий узел в течение 126 мин (красная и оранжевая линии на графике соответственно) и до 15 000

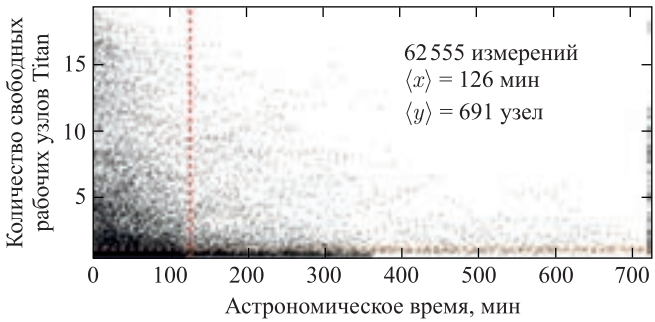

Рис. 55 (цветной в электронной версии). Корреляция количества свободных узлов LCF Titan и времени, когда узлы были свободны

узлов в течение 30–100 мин. Это дает брокеру задач дополнительные возможности в их распределении по вычислительным ресурсам, а варьированием числа генерируемых или моделируемых событий «создавать» задачи различной длительности. Среднее время выполнения задач приведено на рис. 56 (задание № 9235668).

Из приведенного распределения времени выполнения 99 400 задач видно, что среднее время выполнения составляет около 64 мин. Таким образом, варьирование числа обрабатываемых/генерируемых событий, а значит, и времени выполнения задания в зависимости от количества и длительности свободных узлов СК позволит более эффективно использовать «свободные узлы» СК и для выполнения задач.

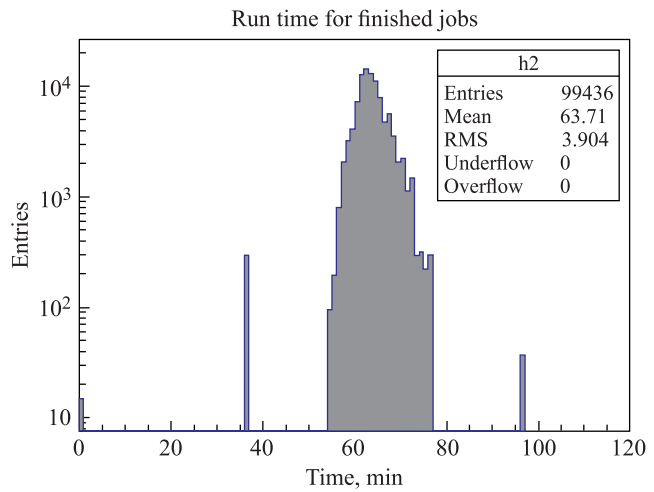

Рис. 56. Среднее время выполнения задач моделирования эксперимента ATLAS (ноябрь 2016 г.) на СК Titan (задание № 9235668)

Интеграция ресурсов HPC и их будущее эффективное использование требуют изменения в методике управления заданиями. «Размер» задачи (время ее выполнения) должен устанавливаться с учетом «размера» (мощности) свободного ресурса. Такого требования не существовало в гомогенной системе грид, когда размер всех задач в рамках одного задания был одинаков. При архитектуре WMS, описанной ранее, и разделении уровней подготовки (DEFT) и выполнения (JEDI) задания требование о динамическом распределении количества задач становится реализуемым на уровне JEDI. Более того, разработанная методика позволяет использовать различные ресурсы (HTC, HPC, облачные ресурсы) для одного задания, формируя «всемирное облако».

4.2.3. Интеграция суперкомпьютера Titan с грид-инфраструктурой с использованием подхода ProdSys2–PanDA. Основная идея подхода — использование уже существующих компонентов системы управления загрузкой и следование логике ее работы. Основные изменения были связаны с работой и логикой работы пилотных задач. Классическое использование таких задач предполагает их выполнение на каждом рабочем узле и передачу информации центральному серверу PanDA (см. рис. 38). Единственный узел СК, имеющий связь с внешним миром, — это DTN (Data Transfer Node). Реализованная архитектура показана на рис. 57. Пилотная задача (PanDA Broker) включает в себя:

— обмен информацией с центральным сервером;

— получение информации от сервиса СК о наличии свободных узлов и периоде их доступности;

— через систему пакетной обработки СК запуск задачи на выполнение;

— по окончании задачи инициирование передачи результатов на центр грид.

Данная архитектура не является специфичной для ФВЭ и ЯФ или для конкретного типа СК. Для взаимодействия с системой пакетной обработки СК используется интерфейс на базе пакета SAGA-Python (Simple API for Grid Applications). Выполняемый программный код должен находиться на распределенной файловой системе СК, центральный сервер может находиться в любом месте (так, для различного применения сервер был установлен в ЦЕРН (эксперимент ATLAS), EC2 (проект LSST), ОИЯИ (эксперимент COMPASS), НИЦ КИ (приложения в области биоинформатики)).

Сама методика интеграции и использования ресурсов HPC и HTC была успешно применена для многих СК, в том числе Titan, Anselm, НИЦ КИ и СК Иллинойсского университета в Урбане-Шампейне. Количество свободного ЦПУ-ресурса и его использование для 15 мес. (2016–2017 гг.) показано на рис. 58. Из графика видно, что использование свободного ресурса наращивалось постепенно и к февралю–марту 2017 г. превысило 31 %.

Общий вклад суперкомпьютерных ресурсов в реализацию программы экспериментов в области ФВЭ и ЯФ рассмотрен в конце данного раздела (эти ра-

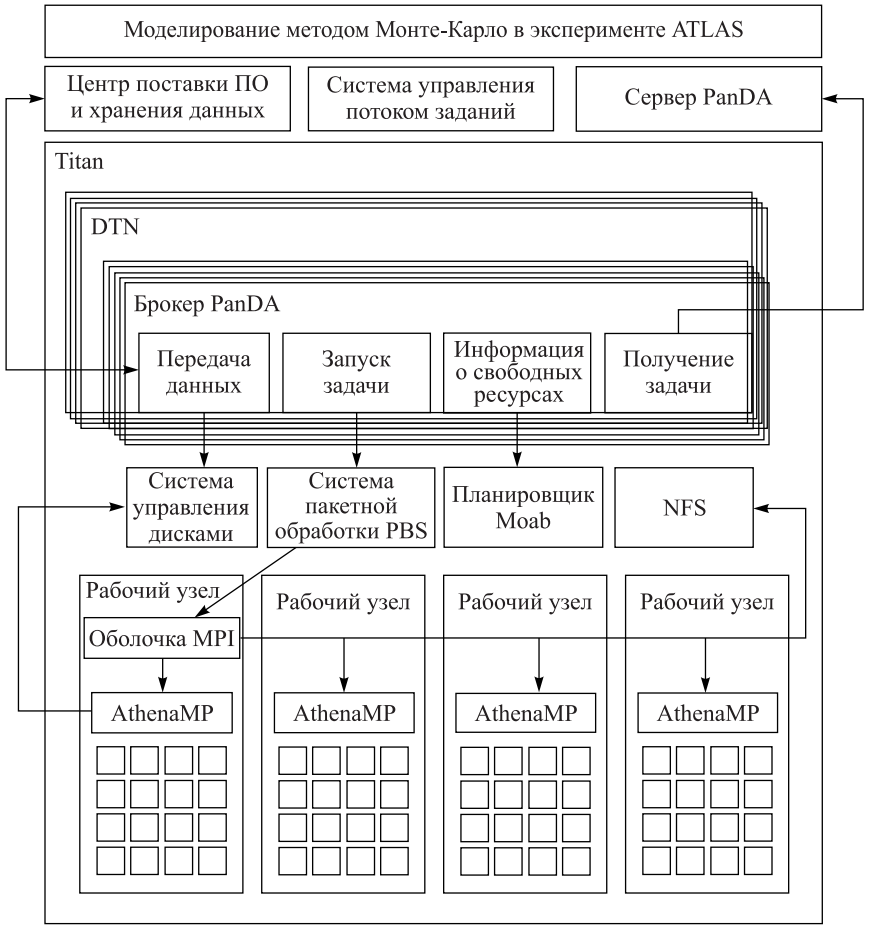

Рис. 57. Архитектура системы управления заданиями для суперкомпьютера

боты, во многом пионерские, велись совместно со специалистом ЛИТ ОИЯИ Д. Олейником и сотрудниками Брукхейвенской и Ок-Риджской национальных лабораторий).

Третий фундаментальный вопрос интеграции суперкомпьютеров — *как выполнять программный код экспериментов в области ФВЭ и ЯФ на СК и как делать это эффективно*. Разработка архитектуры базового программного обеспечения и оптимизация работы кода физических программ выходят за пределы данного исследования и представляют собой отдельный крупный проект в ФВЭ и ЯФ. Остановимся кратко на том, что будет способствовать дальнейшему использованию СК для приложений в области ФВЭ и ЯФ. Ответ

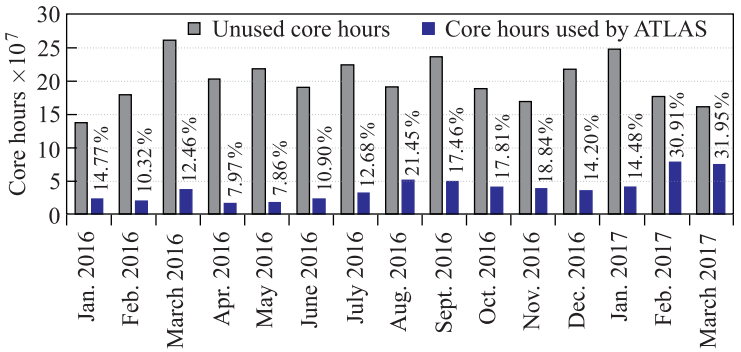

Рис. 58. Количество свободного ЦПУ-ресурса LCF Titan и его использование в фоновом режиме

на этот вопрос требует значительного изменения в структуре кода, его большей модульности, например, четкого разделения между кодом моделирования и реконструкцией событий, что позволит уменьшить зависимость программ моделирования от внешних библиотек. Код многих экспериментов органично рос с годами работы и сейчас достигает 4 млн инструкций (в большей части на языках python и C++). Используемые фреймворки (Gaudi, Athena, AliROOT [82–84]) были разработаны 10–15 лет назад. Вышеизложенное требует совместной работы физических экспериментов и специалистов в области системотехники и других компьютерных наук. И проблема НИР последних лет, хорошо понимаемая в физическом сообществе, направлена на разработку новых фреймворков: AthenaMP, AlFa.

Использование графических процессоров (ГПУ) может дать гораздо больший эффект, если они будут использованы на СК для приложений в области физики частиц. Одними из первых кандидатов могут стать задачи «машинного обучения» для исследования работы сложных систем обработки и анализа данных, а также программы восстановления треков частиц.

4.2.4. Развитие компьютерной модели. Интеграция суперкомпьютера НИЦ «Курчатовский институт» с системой вычислений грид. Рассмотрим интеграцию суперкомпьютера НИЦ «Курчатовский институт» с грид. Эта работа привела к использованию СК не только для приложений в области ФВЭ и ЯФ, но и к применению системы управления потоками заданий для приложений в области биоинформатики.

Детектор переходного излучения (TRT — Transition Radiation Tracker) ATLAS является составной частью внутренней системы эксперимента, в которой измеряются импульсы заряженных частиц. Роль TRT состоит в улучшении пространственного разрешения при восстановлении высокоэнергетических треков, он позволяет провести распознавание электронов и π-мезонов.

Восстановление сигнала от каждого пропорционального счетчика в TRT в условиях большой загрузки детектора увеличивает затрачиваемое процессорное время и позволяет провести исследования для одних из самых сложных задач, а именно исследовать работу детектора и смоделировать его работу на этапе супер-LHC (работу ускорителя с увеличенной светимостью и при большей энергии пучков).

Для выполнения задач реконструкции использовалась компьютерная инфраструктура WLCG, но ко времени проведения (пере)обработки данных все выделенные для эксперимента компьютерные ресурсы полностью были загружены. В связи с этим важным этапом является интеграция новых компьютерных мощностей, таких как суперкомпьютеры, в единую компьютерную инфраструктуру.

Задача реконструкции pp-событий при высокой множественности — одна из наиболее сложных проблем, возникающих в ходе физических исследований в эксперименте ATLAS. Решение этой проблемы требует значительного вычислительного ресурса. Необходимо было решить две задачи: интегрировать суперкомпьютерный центр (СКЦ) и центр уровня T1 грид, используя созданную систему управления загрузкой (megaPanDA), проверить полученные физические результаты (в силу сложности программного кода ATLAS и аппаратных различий между T1 и СКЦ этот шаг был необходим). Проверка была осуществлена в несколько этапов. На первом этапе происходило подтверждение наличия всех необходимых версий программных пакетов на ресурсах T1 и СКЦ. Для этой цели были запущены базовые задания по реконструкции pp-событий в трех центрах: ЦЕРН, T1 НИЦ КИ и суперкомпьютере НИЦ КИ.

После успешного завершения пилотных задач на ресурсах НИЦ КИ в необходимой версии программной среды Athena (эксперимента ATLAS для физического кода) требования к реконструкции событий были изменены. Дополнительно к базовой реконструкции было добавлено требование о восстановлении полной информации о треках частиц в детекторе переходного излучения. Данное введение было применено для полного соответствия задач реальным физическим задачам в группе ATLAS TRT.

Временные тесты были проведены с использованием моделированных данных. В качестве входных файлов были использованы данные, содержащие информацию о траекториях частиц в виде электронных сигналов, снимаемых с детекторов. Временные тесты, использующие 500 реконструированных событий в детекторе ATLAS, показали идентичные результаты для всех трех центров.

Проверка физических параметров продемонстрировала 100%-е согласие выходных данных, полученных для двух разных вычислительных архитектур в ЦЕРН (T0), НИЦ КИ (T1) и НИЦ КИ (СК). Для данной проверки использовались комплексные переменные TRT, согласие в распределении которых

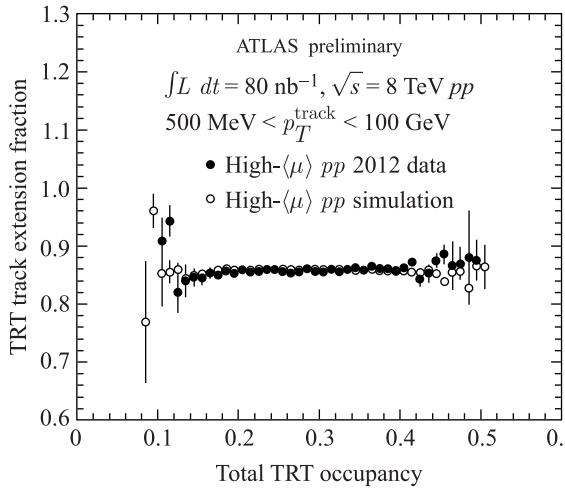

Рис. 59. Доля треков частиц, имеющих более 19 хитов (сработавших пропорциональных счетчиков) в TRT в зависимости от загрузки детектора

может быть достигнуто только при условии совпадения распределений многих кинематических величин, таких как импульс частицы, псевдобыстрота и т. п. На рис. 59 представлено одно из таких распределений: доля треков частиц, имеющих более 19 хитов (сработавших пропорциональных счетчиков) в TRT в зависимости от загрузки детектора.

На втором этапе после подтверждения идентичности физического результата было необходимо проинтегрировать СКЦ и провести массовое моделирование и обработку данных. Для этого была создана специализированная *очередь выполнения заданий* (RRC-KI-HPC) с параметрами 32 узла (256 вычислительных ядер, 2 ГБ оперативной и 1 ГБ swap-памяти на ядро) для выполнения задач моделирования и реконструкции. На рис. 60 показана статистика выполнения заданий ATLAS с использованием вычислительных ресурсов НИЦ КИ (суперкомпьютерные очереди имеют префикс HPC). Эта работа позволила начать масштабную интеграцию суперкомпьютеров с центрами обработки данных грид сначала для эксперимента ATLAS, а потом и для эксперимента в области ЯФ ALICE, и для эксперимента на ускорителе SPS — COMPASS.

4.2.5. Реализация и использование системы управления загрузкой megaPanDA для приложений в области биоинформатики на суперкомпьютере НИЦ КИ. Демонстрация успешного решения при управлении потоками заданий для приложений в области физики частиц, интеграция суперкомпьютеров и их эффективное использование, а также повышение эффективности использования СК за счет выполнения заданий в «фоновом режиме»

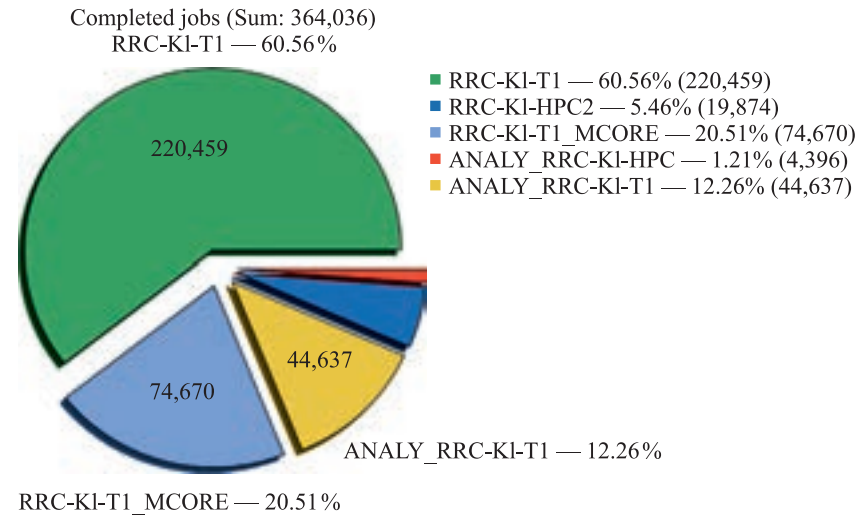

Рис. 60. Статистика выполнения заданий ATLAS с использованием вычислительных ресурсов НИЦ КИ

привлекли интерес научных сообществ за пределами LHC, ФВЭ и ЯФ, что требует значительных вычислительных ресурсов и высокоинтенсивных вычислений. Одним из таких примеров стало использование системы управления загрузкой megaPanDA для решения задачи анализа данных геномного секвенирования шерстистого мамонта (анализ древней ДНК *).

К настоящему времени исследования в области древней ДНК все чаще используются для решения многих фундаментальных и прикладных вопросов. Использование ДНК из археологического и палеонтологического материала позволяет решать многочисленные задачи, связанные с эволюцией экосистем в различных климатических условиях, с происхождением и эволюцией многих патогенных микроорганизмов.

В данном случае использовались геномные чтения шерстистого мамонта (Mammuthus primigenius) [85]. В настоящее время для анализа древней ДНК используются специальные программные пакеты-конвейеры, включающие ряд программных компонентов, с помощью которых осуществляется быстрая обработка данных NGS. Одним из наиболее популярных программных конвейеров является пакет PALEOMIX [86]. Ранее опубликованные результаты

*Древняя ДНК — генетический материал, извлеченный из древних биологических образцов. Первые исследования с выделением и анализом древних фрагментов ДНК начались более 30 лет назад, первоначально работы проводились с небольшими участками митохондриального или ядерного генома.

были получены на 80-ядерном сервере, имеющем 512 ГБ оперативной памяти. На получение результата затрачено около 2 мес. Анализ данных требовал 900 млн парных чтений. Срок в 2 мес. был обусловлен несколькими причинами:

— большим количеством неавтоматизированных операций на этапе подготовки и выполнения программы секвенирования (ручной запуск и перезапуск задачи (в случае сбоя), загрузка входных данных, мониторирование времени исполнения задачи);

— особенностями программного пакета PALEOMIX и метода его использования (отсутствие параллельного выполнения между несколькими узлами, отсутствие разбиения процесса выполнения на этапы);

— требованием выделенного ресурса с аппаратными характеристиками (оперативная память, количество ядер).

Было предложено использовать созданную систему управления загрузкой, выделить задания, которые могут быть выполнены параллельно, и автоматизировать процесс запуска и выполнения, т. е. применить разбиение входных данных (около 350 ГБ) на группы и провести «сборку» результатов в конце работы программного пакета PALEOMIX. Таким образом, было введено параллельное выполнение заданий на уровне групп данных. И цепочка заданий стала аналогичной цепочке моделирования методом Монте-Карло для приложений в области ФВЭ и ЯФ. На первом этапе производится разбиение всего пространства данных на отдельные файлы согласно логике, определенной специалистами. Этот процесс оформлен в виде отдельного задания, выполняемого на одном узле. На втором этапе для каждого полученного файла запускается конвейер PALEOMIX как набор задач в рамках одного задания (аналоги task и job рассмотрены ранее). Эти задачи могут выполняться в параллельном режиме на распределенной инфраструктуре. На третьем этапе с помощью конвейера PALEOMIX объединяются все результаты предыдущего этапа, что реализуется в одном задании.

Для демонстрации были использованы СК НИЦ КИ и система управления заданиями megaPanDA (в данной работе большую роль сыграл сотрудник НИЦ КИ Р. Ю. Машинистов). В результате весь процесс анализа данных занял 4 дня [87]. Тем самым продемонстрировано, что программные средства и методы обработки больших объемов экспериментальных данных, которые были разработаны в области ФВЭ для экспериментов на ускорителе LHC, могут быть успешно применены и в других областях науки.

4.3. Роль суперкомпьютеров для научной программы экспериментов в области физики частиц. В разд. 2 и выше мы рассмотрели методику и подходы при интеграции суперкомпьютеров с вычислительными мощностями грид. Такая интеграция позволила провести исследования, которые были отложены на неопределенный срок из-за отсутствия вычислительного ресурса WLCG, причем исследования эффективности работы детектора пе-

реходного излучения не являются единичным примером. Группа под руководством проф. Р. В. Коноплича работала над исследованием редкого процесса: $pp \to x0 \to ZZ \to 4l$. Необходимо было выполнить полное моделирование процессов физики элементарных частиц с участием бозона Хиггса, адаптированное в соответствии с требованиями коллаборации ATLAS. При моделировании нужно было учитывать специфику протекающих событий на генераторном уровне, адронизацию конечного состояния и особенности установки ATLAS. По причинам, выходящим за интерес обсуждения в данной работе, это исследование не было признано приоритетным координатором физической программы эксперимента, поэтому группе не была выделена квота для использования грид-ресурсов. Тогда было предложено использовать суперкомпьютерный ресурс для демонстрации интеграции СК и грид на реальном физическом приложении и показать прозрачность этого использования для системы управления загрузкой и одновременно возможность применения его как дополнительного ресурса для решения реальной задачи физики частиц. Было смоделировано более 15 млн событий (рис. 61), результат исследований был признан настолько значительным, что был опубликован в научном журнале «Physics Letters B» и был представлен на международных конференциях [88].

Разработанные методика, методы и архитектура позволили создать глобальную распределенную систему для обработки данных. Реализация такой системы стала ключевым этапом для дальнейшего развития компьютерной модели и открыла возможность для создания гетерогенной киберинфраструктуры, что позволило использовать ресурсы суперкомпьютеров и облачных

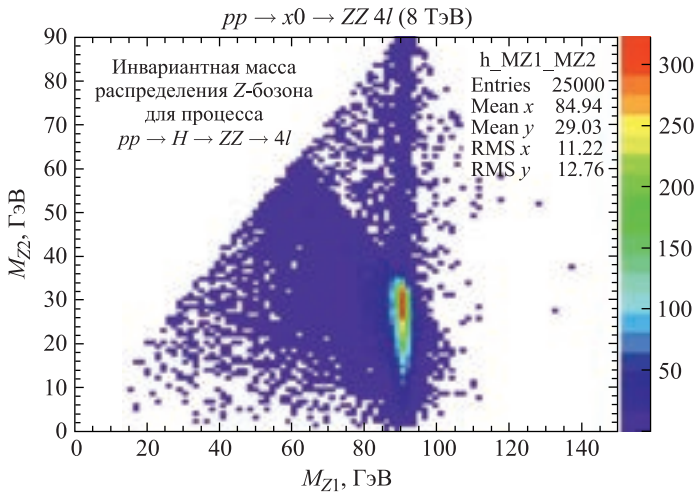

Рис. 61. Результат моделирования 15 млн событий для распада $pp \to x0 \to ZZ \to 4l$

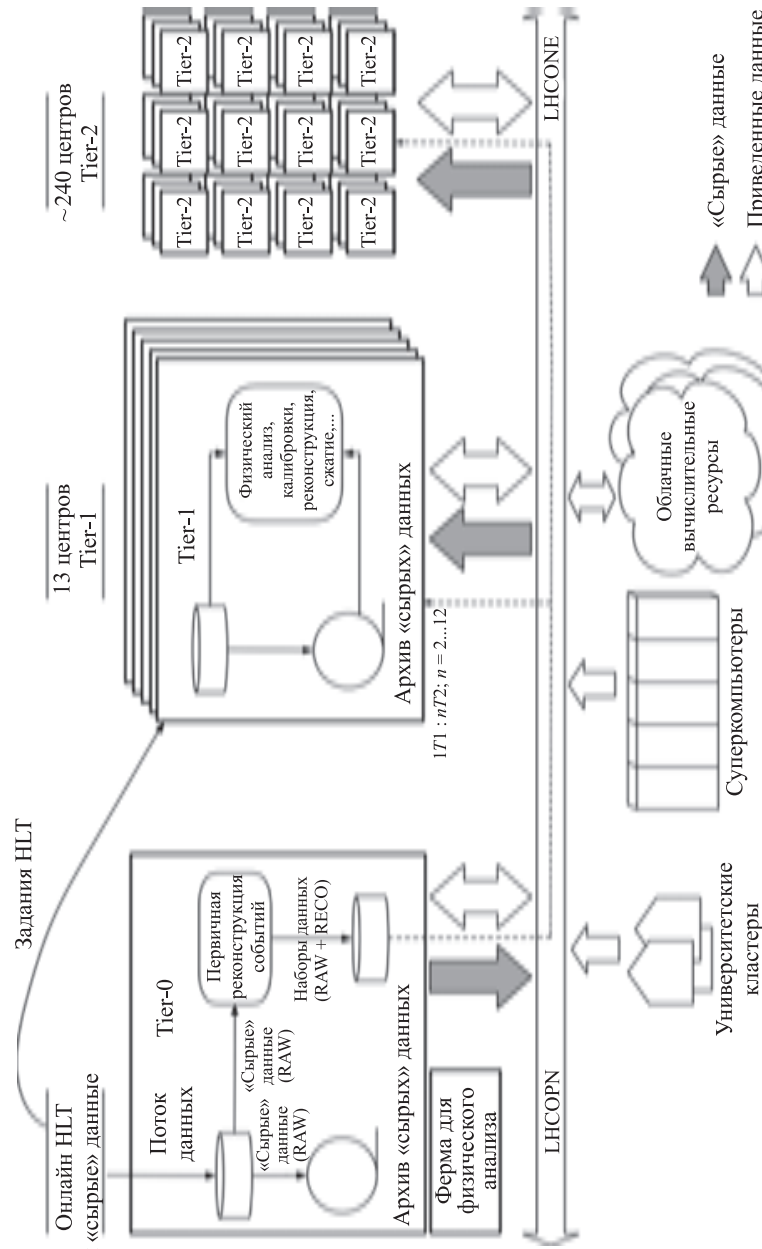

Рис. 62. Новая компьютерная модель, реализованная для второго и последующих этапов работы

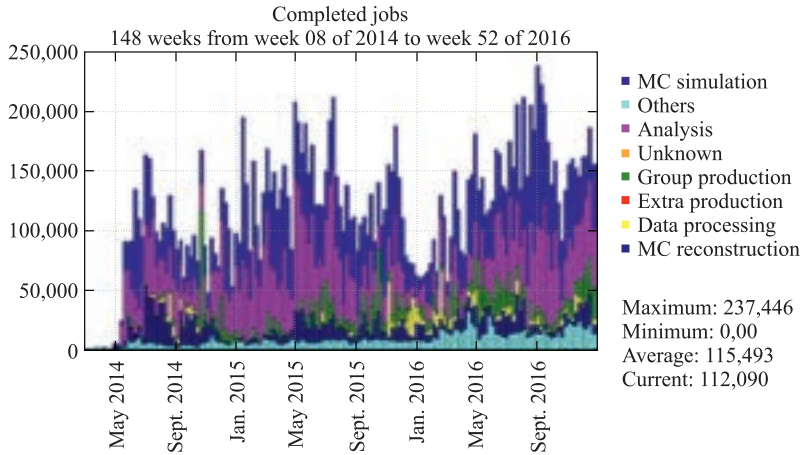

Рис. 63. Количество задач, выполненных на ресурсах HPC в 2014–2016 гг.

вычислений наряду с существующей инфраструктурой грид и нивелировать архитектурные различия вычислительных мощностей. На рис. 62 схематично представлена компьютерная модель, принятая после интеграции ресурсов грид с ресурсами суперкомпьютеров, облачными ресурсами и ресурсами университетских кластеров. Созданная система распределенной обработки данных имеет беспрецедентную эффективность (более 2 млн задач, выполняемых ежедневно). На рис. 63 представлен график, показывающий количество задач, выполненных на суперкомпьютерных ресурсах в 2014–2016 гг. В среднем выполняется более 115 тыс. задач в день, а из потоков заданий наиболее популярными являются задачи моделирования и анализа. Вклад HPC в компьютерный бюджет эксперимента ATLAS исчисляется миллионами ЦПУ-часов в месяц.

4.4. Архитектурные принципы, методы и технологии при создании географически распределенного федеративного дискового пространства в рамках гетерогенной киберинфраструктуры. На пути к созданию «озера научных данных». Рассмотрим базовые принципы и тенденции при управлении и хранении данных физического эксперимента и то, как это влияет или может повлиять на развитие компьютерной модели. Перечислим основные положения и требования со стороны физических экспериментов к системам хранения данных.

• Цена данных может быть рассчитана исходя из стоимости строительства и ввода в эксплуатацию научной установки и ускорителя, стоимости эксплуатации установки и ускорителя, нормированными на время работы ускорителя:

— цена данных может составить до тысячи долларов США за секунду работы ускорителя;

— в случае астрофизических экспериментов цена может быть гораздо выше (из-за затрат по запуску эксперимента в космос).

• Эксперименты не могут доверить свои данные и ответственность за них третьей стороне (на это есть как научные, так и социологические причины, не говоря уже о цене такого решения).

• Места хранения данных и их обработки до последнего времени были жестко связаны между собой:

— расширение компьютерных ресурсов за счет суперкомпьютеров и ресурсов облачных вычислений сделали эту зависимость менее явной, но для гарантированного ресурса данное утверждение справедливо практически для всех центров уровней T0, T1 и T2;

— такое решение ограничивает выбор третьей стороны, которая могла бы предоставлять вычислительный ресурс на коммерческой основе.

• Данные распределены между многими центрами ($O(100)$) по всему миру.

• Сценарий работы, желательный для физических экспериментов:

— данные хранятся все время, но эксперименты оплачивают их обработку, только когда они используются;

— данные должны динамично доставляться в точку, где они будут обрабатываться.

• У нас есть данные различных классов, возможно применение различных технологических и архитектурных решений для данных, используемых в физическом анализе и при архивировании (презервации) данных.

• Информация о локализации данных требует существенного упрощения и «укрупнения» (поиск данных по 100 центрам и управление всеми копиями не оптимальны уже сегодня и ведут к задержкам при обработке и анализе данных).

Но основной проблемой остается сценарий, когда петабайты данных передаются между сотнями ВЦ, поэтому предложенная идея «всемирного облака» при управлении потоками заданий и динамическом создании «облака ресурсов» заставляет искать аналогичное решение для организации хранения и управления данными.

По различным оценкам в ближайшие 5–10 лет объемы данных научных исследований достигнут эксабайтного диапазона [89]. К таким исследованиям, в первую очередь, относятся эксперименты в области ФВЭ и ЯФ на Большом адронном коллайдере (LHC, ЦЕРН), эксперименты на коллайдере NICA (ОИЯИ, Дубна) и комплексе FAIR (GSI, Германия), исследования в области радиоастрономии (квадратный километровый массив SKA — Square Kilometre Array), молекулярной биологии и биоинформатики (геномное секвенирование), вычислительной нейробиологии (например, проект по созда-

нию цифровой модели мозга BlueBrain в Политехническом институте в Лозанне). Это может стать новой эпохой в науке — эпохой эксаскейл.

Для разработки модели управления, обработки и хранения данных в эпоху эксаскейл необходимо учитывать развитие информационных технологий последних лет, в частности наличие высокоскоростных сетей передачи информации (WAN — Wide Area Networks) и увеличение их пропускной способности. Возможности вычислительных сетей должны учитываться наряду с такими классическими ИТ-ресурсами, как процессоры и системы хранения информации. Фундаментальной проблемой является создание единой вычислительной инфраструктуры на основе ресурсных центров высокоскоростных вычислений (суперкомпьютеров), ресурсных центров высокопроизводительных вычислений (грид) и центров облачных вычислений, связанных WAN с пропускной способностью 100 Гбит/с. Проблема целого и его частей привлекает внимание с давних времен: Аристотель писал, что «целое больше суммы его частей». Мы предполагаем, что именно использование ресурса высокоскоростных сетей передачи информации и позволит решить проблему «создания целого в ИТ», когда вычислительный ресурс и обрабатываемые (анализируемые) данные находятся в различных географических точках, но для конечного пользователя вычислительная инфраструктура выглядит как единый вычислительный кластер. Для реализации такой инфраструктуры необходимо решить следующие задачи.

1. Создание единого дискового пространства между географически удаленными центрами, когда для конечного пользователя инфраструктура выглядит как единый массив хранения данных, с единым пространством имен и единой точкой входа. Конечным результатом будет создание распределенной вычислительной инфраструктуры (российского «озера научных данных»), включающей вычислительные центры с различными архитектурами (суперкомпьютеры, ресурсы облачных вычислений, ресурсы высокоскоростных вычислений).

2. Создание новых алгоритмов и приложений для обработки и управления данными на основе методов машинного обучения (с использованием графических процессоров общего назначения — GPGPU и, возможно, квантовых компьютеров) и систем запросов о состоянии данных (с использованием технологий нереляционных баз данных для хранения метаинформации).

3. Создание систем управления потоками заданий для обработки и анализа данных в созданной вычислительной инфраструктуре для нивелирования различий в архитектурных решениях различных вычислительных центров для пользователя.

4. Применение методов машинного обучения при определении популярности данных и автоматического распределения данных между носителями информации (высокоскоростные твердотельные диски — SSD, менее

дорогие шпиндельные дисковые носители, системы ленточного архивирования).

5. Создание новых систем мониторирования и контроля (на основе систем принятия решений), а также визуального анализа функционирования распределенной вычислительной инфраструктуры с использованием классических методов визуальной аналитики и машинного обучения:

а) совместное применение методов статистического анализа, машинного обучения и интерактивной визуальной аналитики для глубокого анализа процессов функционирования распределенных вычислительных инфраструктур с целью поиска возможных причин нестабильной работы различных вычислительных задач и принятия оптимизационных решений;

б) моделирование процессов анализа и обработки для предсказания состояния системы (предиктивная аналитика) и отслеживание возможных отказов системы для принятия превентивных мер по перераспределению ресурсов.

Для научных приложений вопрос создания «озера данных» является одним из наиболее актуальных. В рамках международных проектов (WLCG — Worldwide LHC Computing Grid, ЦЕРН), национальных проектов (IRIS-HEP, США; INDIGO, Италия–Германия) коммерческими ИТ-компаниями (Google, Amazon, Yandex) начаты и ведутся НИР по данной тематике. Концепция «озеро данных» для физических экспериментов в эпоху супер-LHC обсуждалась при написании единого документа по развитию программного обеспечения для экспериментов в области ФВЭ [90]. Одно из возможных решений создания архитектуры вычислительной инфраструктуры и использование концепции «озера данных» представлено на рис. 64.

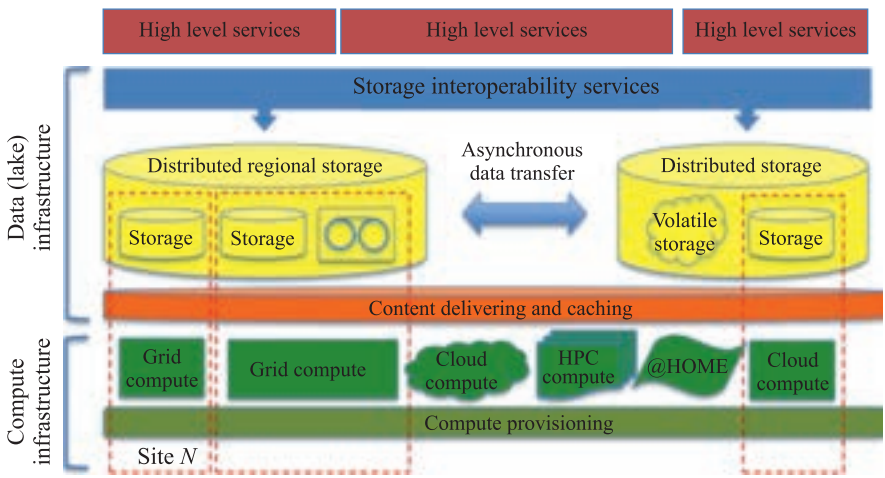

Рис. 64. Вычислительная архитектура «озера научных данных»

Основные концепции, заложенные в модель такой архитектуры:

— логическое разделение вычислительной инфраструктуры и хранилищ данных (информации);

— наличие сервисов управления потоками заданий, загрузкой и данными высокого уровня (такие сервисы взаимодействуют со всеми уровнями инфраструктуры и управляют использованием ресурсов для задач обработки и анализа данных);

— наличие иерархической, географически привязанной структуры «озер данных» различного объема с четко определенной и мониторируемой сетевой топологией (такая структура имеет активные внутренние механизмы балансировки, поддержания целостности и необходимой избыточности данных);

— наличие специальных «умных» сервисов для передачи данных между всеми компонентами инфраструктуры;

— наличие сервисов, необходимых для определения и предсказания объемов вычислительных ресурсов при выполнении потоков заданий.

Рассмотрим, каким образом может быть реализовано «озеро научных данных». В основе этой разработки будут использованы результаты, ранее полученные группой ученых и ИТ-профессионалов из НИЦ КИ, ОИЯИ и СПбГУ при создании прототипа распределенного федеративного хранилища данных [91, 93] (рис. 65).

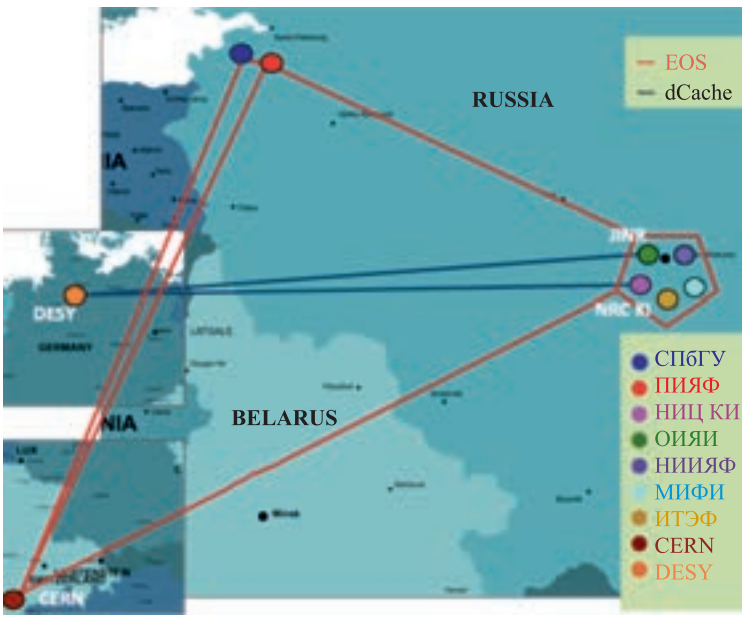

Рис. 65. Прототип распределенного федеративного хранилища данных

Ландшафт современных компьютерных ресурсов, используемых в экспериментах в области ФВЭ и ЯФ, заметно изменился. В последние годы возросла роль суперкомпьютеров для нетрадиционных приложений (в области ФВЭ и ЯФ), характеризующихся большим количеством однородных задач, в отличие от канонических приложений, таких как моделирование климата, квантовой хромодинамики, материаловедения. Для приложений ФВЭ во многих компьютерных центрах используются только вычислительные ресурсы (МИФИ, ИТЭФ, НИИЯФ МГУ), наряду с этим существуют специализированные центры (ВЦ установки ПИК, Гатчина), суперкомпьютерные центры (ВЦ+МГУ, ВЦ РАН, СКИФ и т. д.), а также крупные центры обработки и хранения данных (НИЦ КИ, ОИЯИ). Крупнейшие ВЦ России связаны с европейскими исследовательскими центрами (ЦЕРН, DESY, GSI) через академические высокоскоростные вычислительные сети (LHCOPN, LHCONE, Geant, NorduNet). Нижний уровень передачи данных обеспечивается сервисом FTS (File Transfer Service — *de facto* стандарт для передачи данных экспериментов на LHC).

Существует несколько проблем при создании распределенной гетерогенной вычислительной инфраструктуры:

1) иерархия хранения информации;
2) гетерогенность компьютерных ресурсов;
3) перемещение и управление данными;
4) единое пространство имен.

Мы предлагаем создать на базе российских научных центров и ОИЯИ российское «озеро научных данных» с центрами управления в двух ведущих научных организациях: НИЦ КИ и ОИЯИ. Такая инфраструктура, в первую очередь, необходима для экспериментов на LHC и NICA, но применима для эффективного хранения данных как в других проектах класса мегасайенс (ПИК, с-τ-фабрика), так и в коммерческих приложениях (в частности, для данных крупных компаний, имеющих географически распределенные центры обработки информации (банки), и медицинских учреждений).

Для приложений в области ФВЭ и ЯФ узлы распределенного федеративного хранилища могут быть ассоциированы с российскими исследовательскими центрами и университетами, а само хранилище может выступать как часть глобальной федерации с такими международными научными центрами, как ЦЕРН, DESY, GSI. При этом ключевым требованием к узлам федеративного хранилища является наличие высокоскоростного сетевого соединения с центрами управления. Для научных приложений и ученых, участвующих в исследовательских проектах, данная инфраструктура будет выглядеть как единая высокопроизводительная отказоустойчивая система хранения научных данных с единой точкой входа. Предлагаемая федеративная инфраструктура хранения данных при надлежащей организации будет обладать гибкостью, необходимой для удовлетворения требований проектов класса мегасайенс.

Создание такого хранилища позволит научным центрам меньшего размера эффективно и полноценно участвовать в общей распределенной обработке данных, что избавит их от необходимости разворачивать и поддерживать собственное полнофункциональное хранилище данных, требующее как регистрации в информационных системах научных коллабораций, так и наличия высококвалифицированного штата системных администраторов, разбирающихся в тонкостях организации программного стека различных научных коллабораций. Такие специалисты потребуются только для обслуживания управляющих узлов, являющихся «точками входа» в единую систему хранения данных.

Наличие отказоустойчивого централизованного сервиса управления позволит динамически перераспределять копии данных между центрами — участниками федеративной системы хранения, это минимизирует время доступа к данным без потери надежности. Необходимость и перспективность такого типа организации современных хранилищ научных данных не раз высказывалась в крупных научных коллаборациях (WLCG), однако только сейчас для этого появилась необходимая технологическая база.

Российский проект «озеро научных данных», в первую очередь, состоит в использовании реальных высокопроизводительных ресурсов хранения данных крупнейших российских научных центров, ориентированности на мультидисциплинарное применение, а также в учете особенностей организации современных научных исследований в области физики элементарных частиц.

Основными целями проекта являются:

• Обеспечение прозрачного и эффективного доступа к научным данным для участников научных экспериментов, включая автоматическую репликацию и перебалансировку данных внутри федеративного хранилища, а также оптимизацию эксплуатационных затрат на обслуживание инфраструктуры, что позволит ученым из российских научных центров получить прямой доступ к данным даже в условиях значительной географической удаленности заинтересованных научных коллективов как от экспериментальных установок, так и от высокопроизводительных вычислительных ресурсов.

• Организация единого федеративного дискового пространства для эффективного доступа из крупных российских центров обработки данных, территориально распределенных более чем в 10 географических удаленных местах, и развертывание систем мониторинга и аутентификации, позволяющих гарантировать высокую доступность и безопасность ценных научных данных большого объема.

• Ориентированность на перспективные научные эксперименты в ближайшие 5–10 лет, а также на совместные эксперименты на установках класса мегасайенс со значительной долей российских и зарубежных ученых, таких как LHC, NICA, ПИК, с-τ-фабрика.

В данной концепции «озеро данных» является крупным хранилищем данных, обслуживающим большой географический регион (например, на начальном этапе европейскую часть России). Это потребует постепенной консолидации значительной части хранилища в нескольких ВЦ. Ресурсы хранения, не объединенные в «озеро данных», можно использовать для временного хранения («кэширования» данных). Такой подход позволит существенно уменьшить издержки на эксплуатацию большого количества центров и упростит возможность доступа к данным (а в более широком смысле — к научным данным для информационноемких областей науки).

Создание такой структуры потребует изменения концепции и новых подходов и методов. Рассмотрим возможные компоненты структуры.

• «Озеро данных» (крупные хранилища (сотни петабайт) и мульти-ВЦ):

— (само)балансировка данных внутри «озера»;

— сервисы проверки целостности данных внутри «озера» (при наличии более чем одной копии данных);

— возможны различные типы «озера»:

а) «серверное озеро» — мастер-копия данных (данные для обработки из такого озера копируются в «кэши», задачи обработки не работают напрямую с содержимым «серверного озера»);

б) «озеро для анализа данных» (задачи анализа данных характеризуются высокими требованиями к вводу/выводу, доступ к данным происходит внутри вычислительной инфраструктуры, в которую входят «озеро для анализа данных» и ВЦ, прототип такого решения был создан и его работа была продемонстрирована для задач ATLAS и ALICE [91]);

в) «озеро-архив» — хранилище данных для презервации информации (такой тип хранилища не предполагает интенсивного доступа на чтение информации).

• Внешние ВЦ — центры, не входящие в состав распределенной вычислительной инфраструктуры (могут иметь доступ ко всему набору «озер данных», все компьютерные ресурсы имеют доступ к «озеру данных» через единую точку входа).

• (Супер)компьютерные ресурсы:

— фабрики обработки данных (крупные ВЦ, имеющие прямой доступ к «озеру данных» и достаточную (не менее 10 Гбит/с) пропускную способность WAN);

— суперкомпьютерные центры, используемые для обработки и/или анализа данных (могут иметь доступ к «озеру данных» (СК ОИЯИ и НИЦ КИ и хранилище данных связаны в едином ВЦ) или использовать локальное кэширование данных и обмен между кэшем и «озером данных», что было продемонстрировано при обработке данных ATLAS и исследовании ДНК мамонта на СК НИЦ КИ);

— ресурсы облачных вычислений (могут иметь прямой доступ к данным или копировать данные из «озера» для обработки внутри «облака»).

Создание инфраструктуры «озера данных» потребует разработки системы управления потоками заданий и системы управления данными. Система должна будет знать регион, где есть данные, и направить поток заданий на свободные ВЦ данного региона или принять решение о временном кэшировании данных и использовании СК или ресурсов облачных вычислений (в том числе на коммерческой основе). Помимо разработки нового поколения системы управления загрузкой необходимо будет исследовать следующие вопросы.

• Хранение данных:

— какие данные хранятся на твердотельных накопителях, шпиндельных дисках, лентах;

— как связать метаданные с данными (как (автоматически) мигрировать данные между различными носителями (типами «озер»)).

• Вычислительная инфраструктура:

— вопросы отказоустойчивости;

— оптимальные размеры хранилищ;

— оптимальные размеры кэша данных;

— требования к пропускной способности WAN/LAN;

— мониторинг основных компонентов инфраструктуры.

• Как оптимизировать и сделать эффективными передачу данных в «озеро» и распределение данных внутри «озера» (и/или типов «озер»):

— на основе предопределенного сценария (данные равномерно распределены внутри «озера»);

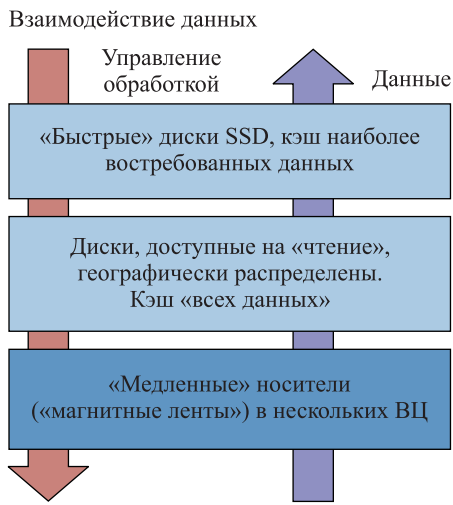

Рис. 66. Взаимодействие данных в пределах «озера»

— на основе свободных (имеющихся) ресурсов (данные поступают в части хранилища, где есть достаточные вычислительные ресурсы);

— с использованием алгоритмов машинного обучения и информации о времени обработки для данного типа данных на конкретном типе вычислительного ресурса.

На рис. 66 представлено взаимодействие данных в пределах «озера».

Существенной мотивацией к реализации данного проекта в России является необходимость создания собственных управляющих центров, контролирующих политику распределения данных внутри «озера данных». При подключении к федерации с центрами управления в иностранных научных центрах участникам из России, вероятнее всего, будет отведена роль разрозненных файловых хранилищ без возможности контроля над получаемыми в рамках федерации данными.

ЗАКЛЮЧЕНИЕ

В данной работе обобщены многолетние исследования по разработке систем для распределенной обработки данных экспериментов и по созданию и развитию компьютерной модели экспериментов в области физики частиц.

Исследования в области ФВЭ и ЯФ невозможны без использования вычислительных систем, а также программного обеспечения для управления, обработки, моделирования и анализа данных. Это определяется рядом факторов: а) большими объемами информации, получаемыми с установок на современных ускорителях; б) сложностью алгоритмов обработки данных; в) статистической природой анализа данных; г) необходимостью (пере)обрабатывать данные после уточнения условий работы детекторов и ускорителя и/или проведения калибровки каналов считывания; д) необходимостью моделирования условий работы современных установок и физических процессов одновременно с набором и обработкой «реальных» данных.

Введение в эксплуатацию Большого адронного коллайдера, создание и запуск установок такого масштаба, как ATLAS, CMS, ALICE, новые и будущие проекты класса мегасайенс (NICA, FAIR, XFEL, LSST), характеризующиеся сверхбольшими объемами информации, потребовали новых подходов, методов и решений в области информационных технологий.

Создание распределенной компьютерной модели для экспериментов в области ФВЭ и ЯФ стало одним из наиболее важных этапов развития компьютинга и изменило подход к работе с данными. Компьютерная модель обработки данных физического эксперимента прошла в своем развитии много этапов.

1. Создание иерархической модели MONARC. Иерархическая модель распределенных вычислений, характеризующаяся иерархией центров (T0, T1, T2, T3), предопределением функций центров, статической организацией групп

центров и «связок» T1:nT2, имеет статический характер распределения данных между центрами с заранее определенным количеством копий.

2. Эволюция модели MONARC: введение понятия популярности (востребованности) классов и наборов данных, переход к «смешанной компьютерной модели». Исследование популярности данных, разработка методов для определения стабильности работы центров уровней T1 и T2, оценка роли и возможностей глобальной вычислительной сети при обработке данных, а также создание методики и методов динамического распределения данных на основе их популярности позволили отказаться от ограничений, введенных иерархической моделью, и перейти к «смешанной компьютерной модели».

3. Создание «смешанной компьютерной модели» и оптимизация использования ресурсов всех центров независимо от их классификации в рамках WLCG. В частности, переход к динамической организации групп центров для выполнения заданий, разработка методики динамического распределения ресурса и принципиально нового поколения системы управления потоками заданий привели к изменению парадигмы обработки и анализа данных, был предоставлен ученым по всему миру (в больших и малых центрах) гарантированный доступ к данным и вычислительным ресурсам, а также возможность активно участвовать в анализе физических результатов.

4. Использование гетерогенных компьютерных ресурсов. Осуществляется интеграция суперкомпьютеров, ресурсов облачных вычислений, университетских кластеров и высокопропускных вычислений (грид) в единую инфраструктуру с прозрачным доступом ко всем ресурсам через систему управления загрузкой. Используется гетерогенная компьютерная инфраструктура с динамическим разделением ресурса между различными потоками заданий и группами пользователей. Создается прототип дисковой федерации, и проводится работа с данными в эксабайтном диапазоне.

За последние два десятилетия физиками, компьютерными учеными и ИТ-инженерами были созданы глобальная распределенная инфраструктура и программное обеспечение для обработки данных на основе динамического управления потоками заданий и динамического распределения данных с учетом пропускной способности WAN. Реализация такой системы стала ключевым этапом для дальнейшего развития компьютерной модели и открыла возможности для создания гетерогенной киберинфраструктуры, что позволило использовать ресурсы суперкомпьютеров и ресурсы облачных вычислений наряду с существующей инфраструктурой грид и нивелировать архитектурные различия вычислительных мощностей. Таким образом, разнородные вычислительные ресурсы стали доступны физикам в виде единой киберинфраструктуры.

Созданная система для глобальной распределенной обработки данных позволяет динамически организовывать группы ресурсов для выполнения научных заданий и динамически разделять вычислительный ресурс между различ-

ными классами заданий, такими как моделирование методом Монте-Карло, обработка данных, анализ данных, потоки заданий отдельных научных групп. Система имеет уникальные характеристики: выполняет более 2 млн задач в день в 250 ВЦ. Использование системы для приложений в других научных областях (астрофизики и биоинформатики) подтверждает ее универсальность.

Дальнейшее развитие компьютерной модели будет связано с использованием дополнительного вычислительного ресурса (включая суперкомпьютеры и коммерческие вычислительные ресурсы) в период пиковых нагрузок и переходом от отдельных центров к федеративной организации ресурсов.

Создание прототипа географически распределенной федерации, демонстрация ее возможностей для реальных приложений в области ФВЭ и ЯФ и работы по созданию «озера научных данных» стали важным шагом в развитии распределенной киберинфраструктуры.

Благодарности. Автор выражает глубокую благодарность своим учителям и наставникам профессорам А. В. Арефьеву, Ю. А. Камышкову, С. Ч. Ч. Тингу и Ю. В. Галактионову, своим коллегам по экспериментам ATLAS, AMS, L3 за плодотворную совместную работу.

Многие результаты, представленные в данной работе, получены совместно с А. К. Кирьяновым, А. К. Зароченцевым, М. А. Григорьевой, Д. В. Краснопевцевым, М. А. Бородиным, Д. В. Голубковым, Т. А. Корчугановой, А. А. Алексеевым, сотрудниками ATLAS (Т. Венаусом, Т. Маено, К. Де, Ф. Баррейро, С. Паниткиным, С. Подольским), коллегами из ORNL (Дж. Велсом) и ЦЕРН (С. Кампана, А. Ди Джироламо). Особенно следует отметить вклад сотрудников ЛИТ ОИЯИ в ряд пионерских исследований на суперкомпьютерах (Д. А. Олейник), а также по созданию и адаптации систем обработки данных (А. Ш. Петросян). Выражаю благодарность за обсуждение результатов работы профессорам В. П. Гердту, В. П. Зрелову, В. А. Ильину, В. В. Иванову, В. В. Коренькову, Г. А. Ососкову.

Работы по созданию российского «озера научных данных» с 2019 г. ведутся в Российском экономическом университете им. Г. В. Плеханова и поддержаны грантом Российского научного фонда № 19-71-30008.

ПЕРЕЧЕНЬ ПРИНЯТЫХ СОКРАЩЕНИЙ И НАИМЕНОВАНИЙ

БД — база данных

ВЦ — вычислительный центр

ДНК — дезоксирибонуклеиновая кислота, одна из трех основных макромолекул

ЕС — Европейский союз

ИС — информационная система

ИТ — информационные технологии

ЛИТ — Лаборатория информационных технологий в ОИЯИ

МКС — Международная космическая станция

НИР — научно-исследовательская работа

НИЦ КИ — Национальный исследовательский центр «Курчатовский институт»

ОИЯИ — Объединенный институт ядерных исследований (Дубна)

ПО — программное обеспечение

СК — суперкомпьютер

СКЦ — суперкомпьютерный центр

СУБД — система управления базами данных

Тэватрон — ускоритель заряженных частиц в Национальной ускорительной лаборатории им. Э. Ферми (Чикаго, США)

У-70 — кольцевой ускоритель в Институте физики высоких энергий (Протвино, Россия)

ФВЭ — физика высоких энергий

Фермилаб — Национальная ускорительная лаборатория им. Э. Ферми (Чикаго, США)

ФРКИ — федеративная распределенная киберинфраструктура

ЦЕРН — Европейская организация по ядерным исследованиям

ЭБ, эксабайт — 10^{18} байт

ЯФ — ядерная физика

AGIS — ATLAS Grid Information System, информационная система грид-эксперимента

AGS — Alternative Gradient Synchrotron, ускоритель в Брукхейвенской национальной лаборатории (США)

AJAX — подход к построению интерактивных пользовательских интерфейсов веб-приложений, заключающийся в «фоновом» обмене данными браузера с веб-сервером

AlFa — новое поколение фреймворка для базового физического кода, совместная разработка эксперимента ALICE и специалистов ИТ-центра FAIR (Дармштадт, Германия)

ALICE — A Large Ion Collider Experiment, тяжелоионный эксперимент на LHC

AliEN — система управления загрузкой эксперимента ALICE в среде грид

AliROOT — фреймворк для базового физического кода в эксперименте ALICE

AMS — Alpha-Magnetic Spectrometer, астрофизический эксперимент на космическом челноке Shuttle (STS91) в 1998 г.

AMS-02 — Alpha-Magnetic Spectrometer, астрофизический эксперимент на МКС

AOD — Analysis Object Data, формат приведенных данных в ФВЭ и ЯФ, используемых для физического анализа

APF — Autopilot Factory, система запуска пилотных заданий

API — Application Programming Interface, интерфейс прикладного программирования

ARC — ATLAS Resource Control, программный модуль, разработанный в консорциуме NDGF, для доступа к ресурсам консорциума

Athena — фреймворк для базового физического кода в эксперименте ATLAS

AthenaMP — фреймворк Athena Multi-Process, развитие фреймворка Athena, версия, поддерживающая многоядерность

ATLAS — A Toroidal LHC ApparatuS, один из двух универсальных экспериментов на LHC

BaBar — эксперимент в области ФВЭ на ускорителе SLAC

Blue Brain — международный проект по моделированию работы мозга и созданию модели работы мозга человека (проект использует СК по всему миру)

BNL — Brookhaven National Laboratory, Брукхейвенская национальная лаборатория (США), также центр уровня T1 для эксперимента ATLAS ($\sim$ 22–24 % всех выделенных коллаборации ресурсов грид)

CASTOR — CERN Advanced Storage Manager, гибридная иерархическая система хранения данных

CDF — эксперимент в области ФВЭ на ускорителе тэватрон

CE — Computing Element, один из элементов грид-инфраструктуры, предназначенный для выполнения вычислительных задач пользователей

CERN — см. ЦЕРН

CMS — Compact Muon Solenoid, один из двух универсальных экспериментов на LHC

COMPASS — эксперимент в области ФВЭ на суперпротонном ускорителе ЦЕРН

Condor-G — менеджер ресурсов, доступных в среде грид

CRIC — Computing Resource Information Catalog, второе поколение информационной системы AGIS

D0 — эксперимент в области ФВЭ на ускорителе тэватрон

DAOD — Derived Analysis Object Data, формат, совместимый с форматом AOD и получаемый путем выборки отдельных событий согласно определенным критериям (например, маска триггера высокого уровня), используется для физического анализа данных

DDM — Distributed Data Management, система управления данными при распределенной модели обработки данных

DESC — Dark Energy Science Collaboration, научное сообщество по поиску «темной материи» на основе исследований, проводимых на телескопе LSST

DST — Data Summary Tape, формат приведенных данных ФВЭ и ЯФ, как правило, в формате DST записываются результаты работы программы реконструкции событий (см. также ESD)

DTN — Data Transfer Node, в суперкомпьютерной архитектуре это узел, имеющий доступ к интернет

DUNE — Deep Underground Neutrino Experiment, международный эксперимент, планируемый на базе ускорителя в Фермилаб. Установка изначально будет располагаться в ЦЕРН (так называемый этап protoDUNE), а в дальнейшем — в шахте штата Северная Дакота (США)

EC2 — Elastic Compute Cloud, коммерческий сервис компании Amazon для проведения облачных вычислений

EGEE — Enabling Grids for E-science in Europe, европейский проект развертывания грид-систем для научных исследований

EGI — European Grid Initiative, один из трех консорциумов, входящих в WLCG

EPFL — Ecole Polytechnic Federale de Lausanne, Политехнический институт в Лозанне (Швейцария), один из ведущих технических университетов в мире

ESD — Event Summary Data, формат приведенных данных в области ФВЭ и ЯФ, как правило, в ESD-формате записываются результаты работы программы реконструкции событий. До работы коллайдера LHC этот формат, как правило, назывался DST (Data Summary Tape)

FAIR — Facility for Antiproton and Ion Research, будущий ускоритель в GSI, ориентированный на тяжелоионные и антипротонные исследования

FTS — File Transfer Service, пакет программ для передачи файлов между центрами обработки LHC

FTS3 — третье поколение пакета FTS

GAUDI — проект ЦЕРН по разработке единого подхода к созданию фреймворков для экспериментов на LHC

GCE — Google Compute Engine, платформа облачных вычислений компании Google

Geant4, G4 — четвертое поколение пакета программ моделирования (Geant), широко используемого в экспериментах в области ФВЭ и ЯФ для моделирования электроники, детекторов и физических процессов (в последние годы этот пакет также используется для приложений в области биоинформатики и ядерной медицины)

GSC — Ground Space Computers, станции AMS-02, установленные непосредственно внутри периметра центра MSFC NASA (штат Алабама, США)

GSI — Центр им. Гельмгольца по исследованию тяжелых ионов (Дармштадт, Германия)

HENP — High Energy and Nuclear Physics (см. ФВЭ и ЯФ)

HEP — High Energy Physics (см. ФВЭ)

HITS — формат данных, полученных в результате оцифровки событий, произведенных моделированием методом Монте-Карло (RDO)

HLT — High Level Trigger, система отбора событий «высшего» уровня, производит окончательный выбор событий для записи на носители (диск и лента) и последующей обработки и физического анализа

HMSF — Hybrid Metadata Storage Framework, гибридное хранилище метаданных

HPC — High Performance Computing, суперкомпьютеры

HTC — High Throughput Computing, грид

HTML — HyperText Markup Language, язык гипертекстовой разметки

HTTP — Hypertext Transfer Protocol, протокол передачи гипертекста

HS06 — HEP SPECint (см. MIPS), дословно — количество миллионов инструкций кода ФВЭ, характеристика производительности вычислительного узла

IP — Internet Protocol, межсетевой протокол

JEDI — Jobs Execution and Definition Interface, динамическая система запуска задач

JIRA — ПО, используемое для учета этапов разработки, отладки и фиксации ошибок большими командами разработчиков ПО

L3 — эксперимент в области ФВЭ на LEP

LAN — Local Area Network, локальная вычислительная сеть

LCF — Leadership Class Facilities, группа суперкомпьютеров в национальных лабораториях США: ALCF (Аргоннской национальной лаборатории: Theta, Mira), OLCF (Ок-Риджской национальной лаборатории: Summit, Titan), NERSC (Национальной лаборатории им. Э. Лоуренса в Беркли), а также CK в Ливерморской национальной лаборатории им. Э. Лоуренса

LEP — Large Electron–Positron collider, ускоритель в ЦЕРН в 1989–2000 гг.

LHC — Большой адронный коллайдер, ускорительный комплекс в ЦЕРН

LHCb — специализированный эксперимент в области ФВЭ на LHC

LHCONE — LHC Open Network Environment, дополнительная (к LHCOPN) вычислительная сеть для связи центров разных уровней консорциума WLCG

LHCOPN — LHC Optical Private Network, глобальная вычислительная сеть, связывающая центры первого уровня WLCG (Tier-1) с центром нулевого уровня (Tier-0, ЦЕРН)

LSF — Load Sharing Facility, система пакетной обработки заданий

LSST — Large Synoptic Survey Telescope, международный проект по созданию телескопа в Южной Америке (Чили) для исследований в области астрономии, стоимость проекта более 1 млрд долларов США

megaPanDA — система управления потоком заданий и загрузкой, созданная в НИЦ КИ (см. также PanDA и WDMS)

MIPS — Million Instructions per Second, характеристика производительности вычислительного узла в миллионах операций в секунду (данная характеристика в 2005–2008 гг. была в ФВЭ заменена на HS06)

MLlib — масштабируемая библиотека для среды Apache Spark, содержащая API для основных языков программирования

MonALISA — Monitoring Agents using a Large Integrated Services Architecture, пакет программ, созданный в рамках проекта MONARC для мониторирования (и в начале для моделирования поведения) распределенных вычислительных систем

MONARC — Models of Networked Analysis at Regional Centres for LHC Experiments, модель обработки и анализа данных экспериментов на LHC в региональных ВЦ

MoU — Memorandum of Understanding, меморандум о взаимопонимании

MSFC — Marshall Space Flight Center, Центр космических полетов им. Дж. Маршалла NASA (штат Алабама, США)

MySQL — реляционная база данных со свободным кодом

NASA — National Aeronautics and Space Administration, Национальное управление по аэронавтике и исследованию космического пространства

NGE — Natural Generic Engineering, естественная генная инженерия

NorduGrid — Nordic Grid, проект развертывания грид-систем для научных исследований в Норвегии, Дании, странах Балтии, Украине, Швейцарии, Словении

NoSQL — Non relational, not only relational, технологии баз данных на основе графов, имеющих существенные отличия от моделей, используемых в традиционных реляционных СУБД с доступом к данным средствами языка SQL

NTUP — табличный формат данных в ФВЭ и ЯФ, как правило, в формате ROOT (см. ROOT), данные в формате NTUP используются в физическом анализе

NWS — Network Weather Service, специальный сервис для сбора и хранения информации о состоянии WAN в эксперименте ATLAS

Objectivity — коммерческая БД, созданная одноименной компанией. Позволила решить вопрос временного/постоянного хранения для коммерческих приложений. Опыт использования Objectivity для хранения метаинформации и ввода/вывода данных для приложений в области ФВЭ и ЯФ был неудачным

ORACLE — коммерческая реляционная БД

OSG — Open Science Grid, проект развертывания грид-систем для научных исследований в США

PanDA — Production and Distributed Analysis Workload Management System, система управления загрузкой для обработки и анализа данных

PBS — Portable Batch System, система пакетной обработки

Phedex — система управления данными эксперимента CMS

POCC — Payload Operations and Control Center, центр AMS-02 в ЦЕРН по сбору данных и телеметрии, контролю работы детектора на борту МКС

PS — Proton Synchrotron, кольцевой ускоритель в ЦЕРН

RAW — «сырые» (неприведенные) наборы данных, получаемые с экспериментальной установки, содержат адрес канала считывания и величину, полученную непосредственно с электроники считывания

RDIG — Russian Data Intensive Grid, проект развертывания грид-систем для научных исследований в России

RDO — Raw Data Object, формат данных, «сырые» данные, полученные в результате моделирования методом Монте-Карло

RHIC — Reliativistic Heavy Ion Collider, коллайдер тяжелых ионов в BNL (США)

RO — Read Only, метод доступа к информации на чтение, не предполагает изменения информации

Rucio — второе поколение системы управления данными эксперимента ATLAS

RW — Read Write, метод доступа к информации на чтение и запись, предполагает изменение информации (иногда удаление информации также относится к этой группе)

SAGA — Simple API for Grid Applications, фреймворк для работы задач пилотов с различными локальными системами пакетной обработки, разработка Ратгерского университета (США)

SDC — Software for Distributed Computing, лаборатория департамента ИТ в ЦЕРН

SE — Storage Element, один из компонентов грид-инфраструктуры, предназначенный для хранения данных

SLAC — Stanford Linear Accelerator, линейный ускоритель в одноименной лаборатории (Калифорния, США)

SLURM — Simple Linux Utility for Resource Management, менеджер ресурсов для кластеров вычислительных узлов под управлением операционной системы Linux

SOC — Science Operations Center, центр AMS-02 по обработке данных в ЦЕРН

Spark (Apache Spark) — программный каркас с открытым исходным кодом для распределенной обработки неструктурированных или слабоструктурируемых данных

SPS — Super Proton Synchrotron, кольцевой ускоритель в ЦЕРН

SQL — Structured Query Language, язык структурированных запросов

SRM — Storage Resource Manager, протокол доступа к постоянному дисковому хранилищу данных

SSO — Single Sign-On Management, система аутентификации пользователей

SW — Software, программное обеспечение

SW&C — Software and Computing, один из проектов в исследованиях в области ФВЭ и ЯФ под общим руководством одного/двух координаторов

TDR — Technical Design Report, документ, описывающий основные компоненты, функции и характеристики экспериментальной установки или одной из ее подсистем. Компьютинг, как правило, рассматривается как одна из подсистем

Tier-0, T0 — гридовский центр нулевого уровня в классификации WLCG, таким центром является только ЦЕРН. Tier-0 используется для экспресс-обработки данных и их постоянного хранения

Tier-1, T1 — гридовский центр первого уровня в классификации WLCG, существуют 13 центров по всему миру. Tier-1 используются для обработки и анализа данных, а также для их долгосрочного хранения

Tier-2, T2 — гридовский центр второго уровня в классификации WLCG, существует ~ 140 центров по всему миру. Tier-2 используются для анализа данных и их временного хранения

Tier-3, T3 — гридовский центр третьего уровня в классификации WLCG, существует ~ 200 центров, включая центры в университетах. Вычислительный ресурс Tier-3 не является гарантированным и может предоставляться на добровольной основе

TRT — Transition Radiation Tracker, детектор переходного излучения установки ATLAS

TWIKI — ПО (разработка ЦЕРН) на базе веб-платформ для хранения документации, инструкций и информации о проекте. Позволяет осуществлять контролируемый доступ к веб-страницам, в том числе их редактирование группами, ведущими работы над проектом

VO — Virtual Organization, понятие грид: совокупность институтов, университетов, групп, объединенных для решения общей задачи в режиме скоординированного использования распределенных вычислительных ресурсов, выделенных для данной виртуальной организации

WAN — Wide Area Network, глобальная вычислительная сеть

WDMS — Workload and Data Management System: система управления загрузкой, вычислительными ресурсами (потоком задач) и данными

WLCG — Worldwide LHC Computing Grid, консорциум, объединяющий грид-ресурсы, предназначенные для работ на LHC

WMS — Workload Management System, система управления потоком задач

WP — Work Packages, раздел работ в рамках проекта

СПИСОК ЛИТЕРАТУРЫ

1. LHC — The Large Hadron Collider. http://lhc.web.cern.ch/lhc.
2. *Aad G. et al. (ATLAS Collab.)* The ATLAS Experiment at the CERN Large Hadron Collider // J. Instrum. 2008. V. 3. P. S08003.
3. *Chatrchyan S. et al. (CMS Collab.)* The CMS Experiment at the CERN LHC // Ibid. P. S08004.
4. *Aamond K. et al. (ALICE Collab.)* The ALICE Experiment at the CERN LHC // Ibid. P. S08002.
5. FAIR Baseline Technical Report / Eds. H. H. Gutbrod et al. 2006.
6. XFEL. The European X-ray Free-Electron Laser Technical Design Report / Eds. M. Altarelli et al. DESY 2006-097. DESY, 2007.
7. *Trubnikov G. V. et al.* Project of the Nuclotron-Based Ion Collider Facility (NICA) at JINR // Proc. of EPAC-08, Genoa, 2008. P. 2581–2583.
8. *Aad G., Klimentov A. et al. (ATLAS Collab.)* Observation of a New Particle in the Search for the Standard Model Higgs Boson with the ATLAS Detector at the LHC // Phys. Lett. B. 2012. V. 716. P. 1–29.
9. *Ratchford J., Colombo U.* Megascience: UNESCO World Science Report. 1996.
10. *Markoff J.* A Deluge of Data Shapes a New Era in Computing. http://www.nytimes.com/2009/12/15/science/15books.html.
11. http://www.fourthparadigm.org
12. *Gray J.* eScience — A Transformed Scientific Method. Talk Given to the NRC-CSTB, Mountain View, CA, USA, Jan. 11, 2007; http://research.microsoft.com/en-us/um/people/gray/talks/NRC-CSTB_eScience.ppt.
13. *Таненбаум Э., ван Стеен М.* Распределенные системы. Принципы и парадигмы. СПб.: Питер, 2003. С. 876.
14. *Воеводин В. В., Воеводин Вл. В.* Параллельные вычисления. СПб.: БХВ-Петербург, 2002. С. 608.
15. *Говорун Н. Н.* Некоторые вопросы применения электронных вычислительных машин в физических исследованиях. Дис. . . . д-ра физ.-мат. наук. ОИЯИ, 10-4437. Дубна, 1969.
16. WLCG: Worldwide LHC Computing Grid. http://wlcg.web.cern.ch.
17. *Foster I., Kesselman K.* GRID: A Blueprint to the New Computing Infrastructure. Morgan Kaufman Publ., 1999. P. 690.
18. LHCOPN: LHC Optical Private Network. http://wlcg.web.cern.ch.
19. LHCONE: LHC Open Network Environment. http://lhcone.cern.ch.
20. *Климентов А.* К вопросу о федеративной организации распределенной ЦЕРН // Суперкомпьютеры. 2015. Т. 20. С. 26–28.
21. Проект Globus. http://toolkit.globus.org/toolkit/about.html.
22. *Ilyin V., Korenkov V., Kryukov A., Ryabov Yu., Soldatov A.* Russian Date Intensive Grid (RDIG): Current Status and Perspectives towards National Grid Initiative // Proc. of Intern. Conf. “Distributed Computing and Grid-Technologies in Science and Education”, GRID-2008. Dubna, 2008. P. 100–108.

23. *Воеводин Вл. В., Жуматий С. А.* Вычислительное дело и кластерные системы. М.: Изд-во Моск. ун-та, 2007. 150 с.
24. *Aderholz M. et al.* Models of Networked Analysis at Regional Centers for LHC Experiments (MONARC) — Phase 2. Report CERN/LCB 2000-001. http://monarc.web.cern.ch/MONARC.
25. *Campana S., Di Girolamo A., Elmsheuser J., Jezequel S., Klimentov A., Schovancova J., Serfon C., Stewart G., van der Ster D., Ueda I., Vaniachine A.* ATLAS Distributed Computing Operations: Experience and Improvements after 2 Full Years of Data-Taking // 19th Intern. Conf. on Computing in High Energy and Nuclear Physics (CHEP12). May 2012.
26. *Klimentov A. et al.* Extending ATLAS Computing to Commercial Clouds and Supercomputers // Proc. of Sci. 2014. V. ISGC2014. P. 034.
27. *Zarochentsev A., Kiryanov A., Klimentov A., Krasnopevtsev D., Hristov P.* Federated Data Storage and Management Infrastructure // J. Phys. Conf. Ser. 2016. V. 762, No. 1.
28. *Климентов А., Кирьянов А., Зароченцев А.* Российское озеро научных данных // Открытые системы. 2018. Т. 4. С. 33–38.
29. Load Sharing Facility. https://www-03.ibm.com/systems/spectrum-computing/products/lsf/index.html.
30. Portable Batch System. http://www.pbspro.org/.
31. HTCondor. Official site: https://research.cs.wisc.edu/htcondor/.
32. *Bagnaso S., Betev L., Buncic P. et al.* The ALICE Workload Management System: Status before the Real Data Taking // J. Phys. Conf. Ser. 2010. V. 219. P. 062004.
33. *Paterson S. K., Tsaregorodtsev A.* DIRAC Optimized Workload Management // J. Phys. Conf. Ser. 2008. V. 119, No. 6.
34. *Klimentov A. et al.* Next Generation Workload Management System for Big Data on Heterogeneous Distributed Computing // J. Phys. Conf. Ser. 2015. V. 608, No. 1. P. 012040.
35. *Cittolin S. et al.* A Remus Based Crate Controller for the Autonomous Processing of Multichannel Data Streams. CERN Preprint 81-07. 1976.
36. *Klimentov A. et al.* The Distributed DAQ System of Hadron Calorimeter Prototype. Preprint ITEP-18. 1989.
37. *Климентов А. А.* Создание комплекса автоматизированных стендов для проведения тестовых испытаний при производстве, сборке и запуске адронного калориметра установки ЛЗ на ускорителе ЛЕП. Дис. . . . канд. физ.-мат. наук. 01.04.01. Ин-т теор. и эксп. физики. М., 1991. РГБ ОД, 9 91-2/3642-7.
38. *Klimentov A. et al.* Computing Strategy of Alpha-Magnetic Spectrometer Experiment // Nucl. Instr. Meth. 2003. V. 502. P. 28–32.
39. *Klimentov A. et al.* AMS-02 Computing and Ground Data Handling // Proc. of Conf. “Computing in High Energy Physics”, Interlaken, Switzerland, Sept. 2004.
40. *Baud J.-P. et al.* CASTOR Status and Evolution // Proc. of Conf. “Computing in High Energy and Nuclear Physics” (CHEP-2003). CA, USA, 2003.
41. *Dobre M., Stratan C.* MONARC Simulation Framework // Proc. of the RoEduNet Intern. Conf.; Buletinul Stiintific al Universitatii “Politehnica” din Timisoara, Romania, Ser. “Automatica si Calculatoare Periodica Politechnica, Transactions on Automatic Control and Computer Science”. 2004. V. 49, No. 63. P. 35–42.

42. Европейский проект развертывания грид-систем для научных исследований — EGEE (Enabling Grids for E-science in Europe). http://www.eu-egee.org.
43. Проект по разработке фундаментальных грид-технологий, Альянс Globus. http://www.globus.org/.
44. ROOT: Data Analysis Framework. https://root.cern.ch.
45. *Costanzo D., Klimentov A. et al.* Metadata for ATLAS. Preprint ATLAS ATL-GEN-PUB-2007-001, ATL-COM-GEN-2007-001.
46. *Lassnig M.* Using Machine Learning Algorithms to Forecast Network and System Load Metrics for ATLAS Distributed Computing. Talk given at Conf. "Computing in High Energy and Nuclear Physics", San Francisco, USA, Oct. 2016.
47. *Titov M., Zaruba G., Klimentov A., De K.* A Probabilistic Analysis of Data Popularity in ATLAS Data Caching // J. Phys. Conf. Ser. 2012. V. 396, No. 3.
48. perfSONAR. http://www.perfsonar.net/.
49. *Klimentov A., Titov M.* ATLAS Data Transfer Request Package (DaTRI) // Proc. of the 18th Intern. Conf. on Computing in High Energy and Nuclear Physics (CHEP-2010); J. Phys. Conf. Ser. 2010.
50. *Schovancova J., Di Girolamo A., Fkiaras A., Mancinelli V.* Evolution of HammerCloud to Commission CERN Compute Resources // Proc. of the 23rd Intern. Conf. on Computing in High Energy and Nuclear Physics, Sofia, 2018.
51. *Anisenkov A., Klimentov A., Kuskov R., Wenaus T.* ATLAS Grid Information System // J. Phys. Conf. Ser. 2011. V. 331. P. 072002.
52. *Pradillo M. et al.* Consolidating WLCG Topology and Configuration in the Computing Resource Information Catalogue // Conf. "Computing in High Energy and Nuclear Physics", San Francisco, CA, USA, Oct. 2016.
53. *Oleynik D., Petrosyan A., Garonne V., Campana S. (ATLAS Collab.).* DDM DQ2 Deletion Service, Implementation of Central Deletion Service for ATLAS Experiment // Proc. of the 5th Intern. Conf. "Distributed Computing and Grid-Technologies in Science and Education" (GRID-2012). Dubna, 2012. P. 189–194.
54. *Robertson L.* LHC Data Analysis Will Start on the Grid. What's Next? Talk given at Conf. "Computing in High Energy and Nuclear Physics" (CHEP-2009), Prague, March 2009.
55. *Аристотель.* Метафизика. М.: Эксмо, 2016.
56. *Martin H. S. C., Jha S., Howorka S., Coveney P. V.* Determination of Free Energy Profiles for the Translocation of Polynucleotides through α-Hemolysin Nanopores Using Non-Equilibrium Molecular Dynamics Simulations // J. Chem. Theory Computation. 2009. V. 5, Iss. 8. P. 1955–2192.
57. *De K., Jha S., Klimentov A., Oleynik D., Wells J. et al.* High-Throughput Computing on High-Performance Platforms: A Case Study. arXiv:1704.00978.
58. Top-500, ноябрь 2019 г. https://www.top500.org/lists/2016/11/.
59. *Stewart G.* Evolution of Computing and Software at LHC: From Run 2 to HL-LH // Conf. "Computing in High Energy and Nuclear Physics", Okinawa, Japan, Apr. 2015.
60. *Borodin M., De K., Garcia Navarro J., Golubkov D., Klimentov A., Maeno T., Vaniachine A.* Scaling up ATLAS Production System for the LHC Run 2 and Beyond: Project ProdSys2 // J. Phys. Conf. Ser. 2015. V. 664, No. 6. P. 062005.

61. *Laycock P. J., Ozturk N., Beckingham M., Henderson R., Zhou L.* Derived Physics Data Production in ATLAS: Experience with Run 1 and Looking Ahead // J. Phys. Conf. Ser. V. 513, No. 3.
62. http://www.wired.com/magazine/2013/04/bigdata
63. *Panzer-Steindel B.* Introduction to CERN Computing // Летняя лекция ЦЕРН 2015 г.
64. Материалы рабочего совещания "BigData Processing and Analysis Challenges", 29–31 янв. 2015 г., НИЦ "Курчатовский институт", Москва. https://indico.cern.ch/event/364112/.
65. *Grigorieva M. A., Golosova M. V., Gubin M. Y., Klimentov A. A., Osipova V. V., Ryabinkin E. A.* Evaluating Non-Relational Storage Technology for HEP Metadata and Meta-Data Catalog // J. Phys. Conf. Ser. V. 762, No. 1.
66. Django Software Foundation. https://www.djangoproject.com/foundation/.
67. ds.j3. https://d3js.or.
68. *De K., Klimentov A., Schovancova J., Wenaus T.* The New Generation of the ATLAS PanDA Monitoring System // Proc. of Sci. 2014. V. ISGC2014. P. 035.
69. *Korchuganova T., Padolski S., Wenaus T.* ATLAS BigPanDA Monitoring and Its Evolution. Talk given at the 7th Intern. Conf. "Distributed Computing and Grid-Technologies in Science and Education", Dubna, 2016.
70. *Foster I., Zhao Y., Raicu I., Lu S.* Cloud Computing and Grid Computing 360-Degree Compared. https://arxiv.org/pdf/0901.0131.pdf.
71. *Sevior M.* Belle Monte-Carlo Production on the Amazon EC2 Cloud // Intern. Conf. ISGC, Taipei, Taiwan, Apr. 2009.
72. Google Compute Engine Portal. https://cloud.google.com/products/compute-engine.
73. HTCondor Project. http://research.cs.wisc.edu/htcondor.
74. CVMFS Portal. http://cernvm.cern.ch/portal/filesystem.
75. Пакет программ для управления виртуальными машинами CERNVM. https://cernvm.cern.ch.
76. Amazon EC2. http://aws.amazon.com/ec2/pricing.
77. Топ-500 суперкомпьютеров. https://en.wikipedia.org/wiki/TOP500.
78. Intern. Conf. "Supercomputers-2016", Salt Lake City, Utah, USA, 2016. http://sc16.supercomputing.org.
79. Geant4. http://geant4.cern.ch.
80. SAGA-Python (Simple API for Grid Applications). http://saga-project.github.io/saga-python/.
81. *Calafiura P. et al.* Running ATLAS Workloads within Massively Parallel Distributed Applications Using Athena Multi-Process Framework (AthenaMP) // Conf. "Computing in High Energy and Nuclear Physics", Okinawa, Japan, Apr. 2015.
82. Проект Gaudi. http://gaudi.web.cern.ch/gaudi/.
83. AliROOT. ALICE Offline Project. https://alice-offline.web.cern.ch.
84. *Al-Turani M. et al.* ALFA: The New ALICE-FAIR Software Framework // J. Phys. Conf. Ser. 2015. V. 664. P. 072001.
85. *Аулов В. А., Климентов А. А., Машинистов Р. Ю., Недолужко А. В., Новиков А. М., Пойда А. А., Тертычный И. С., Теслюк А. Б., Шарко Ф. С.* Интеграция гетерогенных вычислительных инфраструктур для анализа данных геномного секвенирова-

ния // Матем. биология и биоинформатика. 2016. Т. 11, вып. 2. С. 205–213. doi: 10.17537/2016.11.205.

86. *Skryabin K. G., Prokhortchouk E. B., Mazur A. M., Boulygina E. S., Tsygankova S. V., Nedoluzhko A. V., Rastorguev S. M., Matveev V. B., Chekanov N. N., Goranskaya D. A., Teslyuk A. B., Gruzdeva N. M., Velikhov V. E., Zaridze D. G., Kovalchuk M. V.* Combining Two Technologies for Full Genome Sequencing of Human // Acta Nat. 2009. V. 1, No. 3. P. 102–107.
87. *Schubert M., Ermini L., Sarkissian C. D., Jonsson H., Ginolhac A., Schaefer R., Martin M. D., Fernandez R., Kircher M., McCue M., Willerslev E., Orlando L.* Characterization of Ancient and Modern Genomes by SNP Detection and Phylogenomic and Metagenomic Analysis Using PALEOMIX // Nat. Protoc. 2014. V. 9. P. 1056–1082.
88. *ATLAS Collab.* Search for a Charged Higgs Boson Produced in the Vector-Boson Fusion Mode with Decay $H^{\pm} \to W^{pm}Z$ Using pp Collisions at $\sqrt{s} = 8$ TeV with the ATLAS Experiment // Phys. Rev. Lett. 2015. V. 114, No. 23. P. 231801.
89. *Klimentov A., Grigorieva M., Kiryanov A., Zarochentsev A.* BigData and Computing Challenges in High Energy and Nuclear Physics // J. Instrum. 2017. V. 12; doi: 10.1088/1748-0221/12/06/C06044.
90. *Klimentov A. et al.* A Roadmap for HEP Software and Computing R&D for the 2020s. arXiv:1712.06982 [physics.comp-ph]. Dec. 2017.
91. *Klimentov A. et al.* ATLAS & Google —- “Data Ocean” R&D Project. CERN Preprint ATL-SOFT-PUB-2017-002. 2017.
92. *Lassnig M., Klimentov A., De K. et al.* ATLAS & Google — The Data Ocean Project // 23rd Intern. Conf. on Computing in High Energy and Nuclear Physics (CHEP-2018), Sofia, July 9–13, 2018.
93. *Zarochentsev A., Kiryanov A., Klimentov A., Krasnopevtsev D., Hristov P.* Federated Data Storage and Management Infrastructure // J. Phys. Conf. Ser. 2016. V. 762, No. 1.