

УДК 004.4

## Методология построения пользовательского интерфейса вычислительного кластера

Д. В. Беляков

*Лаборатория информационных технологий,  
Объединённый институт ядерных исследований,  
ул. Жолио-Кюри 6, Дубна, Московская область, Россия, 141980*

Email: dmitry@jinr.ru

Современные научные исследования ставят перед учёными задачи, требующие для своего численного решения применения различных методов и алгоритмов, реализация которых возможна только с использованием высокопроизводительных и/или параллельных вычислений. Применение вычислительных кластеров или суперкомпьютеров для численного решения научных задач стало повсеместной практикой в научно-исследовательских и образовательных организациях.

Элементами любого вычислительного кластера являются высокопроизводительные серверы и быстрый интерконнект, образующие основные компоненты кластера — вычислительное поле, систему хранения и обработки данных, пользовательские интерфейсы. Для своей работы вычислительный кластер требует применения ряда сервисов — систему управления ресурсами (менеджер ресурсов и планировщик задач), систему распространения прикладного программного обеспечения (ПО), систему аутентификации и авторизации. К дополнительным сервисам можно отнести систему мониторинга вычислительного поля, систему сбора/анализа статистики использования ресурсов (биллинг).

Одним из основных компонентов вычислительного кластера является Пользовательский интерфейс, который представляет собой связующее звено, организующее работу пользователей с прикладным ПО, системой управления ресурсами и системой хранения и обработки данных.

В работе рассмотрены функции Пользовательского интерфейса и предъявляемые к ним технические требования. Выполнен анализ типичных методов организации и разработана методология построения Пользовательского интерфейса вычислительного кластера. Представлен реализованный подход к организации работы пользователей с вычислительным кластером на примере Пользовательского интерфейса гетерогенной платформы HybridIT (ЛИТ, ОИЯИ).

**Ключевые слова:** вычислительный кластер, пользовательский интерфейс

### 1. Введение

Современные научные исследования формируют ряд сложных задач, численное решение которых требует применения высокопроизводительных и/или параллельных вычислений. Роль вычислительных кластеров или суперкомпьютеров в реализации задач моделирования, обработки и анализа данных экспериментов существенна. При этом к ресурсам вычислительного кластера предъявляются достаточно высокие требования — высокопроизводительные серверы и быстрый интерконнект. Другим не менее важным аспектом достижения высокой производительности вычислительного кластера является применение специализированного прикладного программного обеспечения (ПО) для выполнения расчётов, что, в свою очередь, существенно влияет на «порог вхождения» для рядового пользователя.

Обеспечение работы пользователей с прикладным ПО, системой управления ресурсами и системой хранения и обработки данных реализуется в среде Пользовательского интерфейса (ПИ). Возможности такого взаимодействия пользователя с вычислительным кластером определяются спектром доступных функций ПИ. Чем понятнее эти функции, тем комфортнее становится работа пользователя с вычислительным кластером, что решает задачу достижения требуемой высокой производительности и, в конечном итоге, помогает пользователям более успешно решать свои исследовательские задачи.

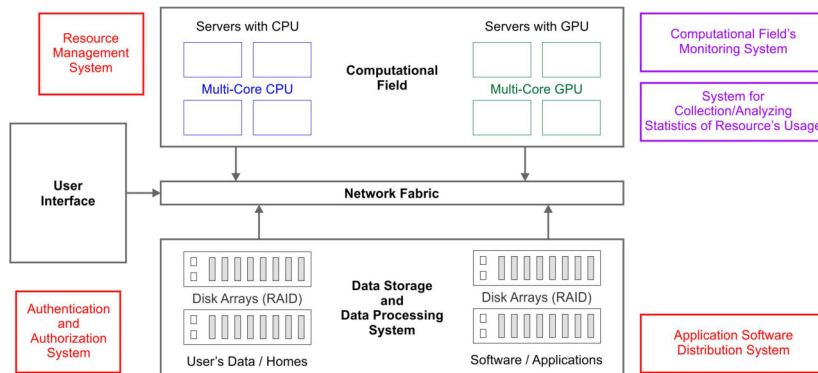


Рис. 1. Структура вычислительного кластера

В работе рассматриваются различные методы организации (способы построения) ПИ, на основе анализа существующих решений разработана Методология построения ПИ. Приведено описание реализованного подхода на примере Гетерогенной платформы HybriLIT (ЛИТ, ОИЯИ).

## 2. Структура вычислительного кластера

Вычислительный кластер [1] строится на основе блока высокопроизводительных серверов, соединённых в локальную сеть с помощью быстрого интерконнекта. На аппаратном уровне серверы и локальная сеть образуют основные компоненты кластера — вычислительное поле (ВП), систему хранения и обработку данных (СХиОД), пользовательский интерфейс (ПИ). Работа кластера обеспечивается на программном уровне, который состоит из ряда сервисов: системы управления ресурсами (менеджер ресурсов и планировщик задач), системы распространения прикладного ПО, системы аутентификации и авторизации. Вспомогательными сервисами могут выступать: система мониторинга вычислительного поля, система сбора/анализа статистики использования ресурсов (биллинг). Структура вычислительного кластера приведена на рисунке 1.

В составе вычислительного поля кластера могут применяться различные вычислительные архитектуры — многоядерные процессоры (Multi-Core CPU), а также графические ускорители (Multi-Core GPU), что позволяет применять различные методы и алгоритмы параллельного программирования для численного решения прикладных программ.

Система хранения и обработки данных представляет собой одну или несколько сетевых файловых систем, построенных на основе дисковых массивов по технологии RAID (Redundant Array of Independent Disks), и подключённых ко всем серверам кластера. СХиОД применяется для размещения файлов разных категорий: прикладного ПО, домашних директорий пользователей, входных и выходных данных при выполнении расчётов.

Пользовательский интерфейс является программно-аппаратной средой, организующей работу пользователей с ВП и СХиОД кластера. В этой среде выполняется редактирование текстов программ, подготовка исполняемых (бинарных) файлов, взаимодействие с планировщиком задач, а также работа с пакетами прикладного ПО.

Система управления ресурсами служит для формирования счётных очередей, осуществляет запуск пользовательских задач, обеспечивает наиболее оптимальную загрузку доступных серверов ВП, исключает нерабочие серверы из счётных очередей.

Система распространения прикладного ПО отвечает за предоставление пакетов прикладных программ на все сервера ВП и ПИ. Для коммерческих программных продуктов в составе данной системы могут применяться различные схемы распространения лицензий (выдача разрешения на запуск ПО менеджером лицензий, локальный файл лицензии и т.п.).

Система аутентификации и авторизации используется для распознавания и наделения разрешёнными правами пользователей (по логину/паролю, ключу, токenu, сертификату и т.п.).

Система мониторинга применяется для наблюдения за состоянием ВП, фиксации текущей загрузки вычислительных ресурсов и локальной сети, что помогает оперативно выявлять и реагировать на внештатные ситуации в работе кластера.

Система сбора/анализа статистики использования ресурсов (биллинг) позволяет оценивать общую востребованность вычислительной функции кластера за определённый период времени.

### 3. Функции Пользовательского интерфейса

В структуре вычислительного кластера Пользовательский интерфейс представляет не менее важную роль, чем другие компоненты. Он является программно-аппаратной средой, в которой происходит взаимодействие пользователей с вычислительным кластером. Возможности рабочей среды ПИ определяются набором доступных функций. При этом, с одной стороны эти функции должны обеспечивать работу пользователей, а с другой стороны не способствовать снижению политики безопасности, что создаёт некие трудности в области системного администрирования.

Работа пользователей с вычислительным кластером описывается набором типовых задач:

- соединение с рабочей средой ПИ;
- редактирование текстов программ;
- загрузка входных файлов с данными в свою домашнюю директорию;
- настройка переменных программной среды (shell environment) для работы с необходимыми компиляторами и библиотеками;
- подготовка исполняемых (бинарных) файлов;
- просмотр загрузки счётных очередей кластера;
- запуск пользовательской задачи (постановка в счётную очередь);
- получение результатов счёта (лог файлов, выходных файлов);
- выгрузка выходных файлов на свой рабочий компьютер;
- интерактивная работа с пакетами прикладных программ;

Анализ типовых задач формирует технические требования, предъявляемые к функциям ПИ. Рабочая среда должна обеспечивать работу следующих функций:

- организацию интерактивной работы пользователя;
- аутентификацию и авторизацию пользователя;
- доступ к домашней директории пользователя;
- минимальный набор текстовых редакторов;

- настройку переменных программной среды (shell environment) для работы с доступными компиляторами и библиотеками;
- копирование (загрузка и выгрузка) файлов с данными;
- работу с планировщиком задач;
- работу с пакетами прикладных программ, в том числе и в графическом режиме;

Таким образом, для работы пользователей с вычислительным кластером, рабочая среда ПИ должна предоставлять ряд сервисов системного и программного уровней. Например, реализация рабочей среды ПИ возможна на основе следующих пакетов: SSH (интерактивный сеанс), Kerberos & LDAP (аутентификация и авторизация пользователей), NFS (сетевые домашние директории), vi/nano/mcedit (текстовые редакторы), source/modulefiles (настройка переменных программной среды), SCP (копирование файлов), SLURM (планировщик задач), X11/RDP (работа в графическом режиме).

## **4. Методы организации Пользовательского интерфейса**

### **4.1. Анализ методов организации ПИ с точки зрения сетевой безопасности**

На практике существует несколько методов построения Пользовательского интерфейса. Например, в качестве ПИ может выступать один или несколько серверов Вычислительного поля, однако такое решение имеет достаточно низкий уровень безопасности. Поскольку все серверы ВП и СХиОД должны быть в едином адресном пространстве, т.е. в этом случае иметь публичные IP-адреса, то присутствует высокая вероятность сетевых атак из внешней сети на кластер.

Следующим методом построения может служить применение блока выделенных серверов, предоставляющих необходимые сервисы, настроенные в соответствии с набором технических требований к функциям ПИ. Данный блок серверов должен поддерживать достаточно жесткую политику доступа из внешней сети, т.е. иметь открытыми только необходимые порты, а все другие серверы кластера должны отвергать любые внешние соединения, допуская только соединения с IP-адресов ПИ. В этом случае риск сетевых атак из внешней сети будет минимальным. Однако такое решение требует постоянного контроля за состоянием уровня безопасности всех серверов кластера.

Вариантом предыдущего метода является применение сетевого пограничного брандмауэра и перенос серверов, реализующих функции ПИ, в отдельную сетевую зону DMZ (Demilitarized Zone, демилитаризованная зона) — за пределы внутренней сети научной организации. Однако такое решение требует настройки пограничного брандмауэра, серверов, размещённых в DMZ, а также других серверов кластера. С точки зрения сетевой безопасности это оптимальное решение, т.к. все серверы, к которым требуется доступ из внешней сети, находятся за пределами внутренней сети. Но поскольку ПИ является одним из компонентов вычислительного кластера, то такое «разбиение» кластера на части создаёт трудности в настройке и системном администрировании.

Ещё одним методом построения является размещение всех серверов кластера в отдельной не маршрутизируемой сети, например, использующей IP-адреса вида 10.0.0.0/8 в соответствии с документом RFC1918 (Address Allocation for Private Internets, Распределение адресов в частных IP-сетях). Для доступа к этой сети пользователи должны использовать промежуточный шлюз из внутренней сети в сеть вычислительного кластера. Для автоматизации процесса соединения может применяться технология SSH ProxyJump. Данное решение максимально защищает кластер от внешних сетевых атак, но усложняет процесс соединения с кластером для пользователей.

#### 4.2. Анализ методов организации ПИ с точки балансировки нагрузки

Поскольку вычислительный кластер ориентирован на решение широкого спектра задач различных пользователей, то рабочая среда Пользовательского интерфейса должна поддерживать одновременную работу многих пользователей. В самом простом случае рабочая среда ПИ может быть представлена только одним сервером, при этом его ресурсы (ядра процессора и оперативная память) будут разделяться между всеми интерактивно работающими пользователями. Очевидно, что при одновременном использовании (запуске) ресурсоёмких пакетов прикладных программ может сложиться ситуация, когда работа одних пользователей станет фактически невозможной при работе других пользователей, т.е. возникнет “конфликт интересов” из-за нехватки ресурсов.

Для преодоления таких ситуаций используют несколько серверов (обычно от 5 или более), в зависимости от общего количества зарегистрированных пользователей и/или различных видов (характера) работы пользователей. При этом вопрос о выборе сервера для своей работы решает пользователь самостоятельно и опытным путем методом перебора. В случае высокой нагрузки на конкретном сервере пользователь вынужден выполнять действия по завершению интерактивного сеанса и открытию нового соединения на другой сервер.

Применение технологии Round-Robin DNS (круговой DNS) в соответствии с документом RFC1794 (DNS Support for Load Balancing, Балансировка нагрузки с помощью сервиса доменных имён) позволяет реализовать автоматический выбор сервера при соединении с рабочей средой ПИ. Данная технология основана на циклическом переборе IP-адресов, присвоенных одному доменному имени. С точки зрения пользователя рабочая среда ПИ представлена одним сервером, с которым он соединяется по доменному имени. SSH-клиент на стороне пользователя запрашивает у сервиса DNS соответствующий доменному имени IP-адрес и получает в ответ очередной адрес из набора IP-адресов, приписанных этому доменному имени. Такой метод позволяет организовать единую точку входа в рабочую среду ПИ, представив пул серверов с разными IP-адресами в виде одного доменного имени. Существенным недостатком метода является отсутствие проверки доступности серверов, т.к. это лежит за рамками функций сервиса DNS. В случае сбоев в работе конкретного сервера (нет доступа по сети, выключено питание, выполняется профилактика и т.п.) сервис DNS всё равно сообщит SSH-клиенту IP-адрес, что приведёт спустя 1-2 минуты к ошибке Connection timeout (вышло время соединения). Таким образом, возможна ситуация, когда одни пользователи смогут соединиться с рабочей средой ПИ, а другие не смогут. Очевидно, что это сильно снижает качество обслуживания пользователей.

### 5. Пользовательский интерфейс Гетерогенной вычислительной платформы HybriLIT

Гетерогенная вычислительная платформа HybriLIT [2] входит в состав Многофункционального информационно-вычислительного комплекса (МИВК) ЛИТ ОИЯИ. Платформа включает в себя суперкомпьютер «Говорун», применяемый для выполнения массивно-параллельных вычислений с использованием различных типов вычислительных архитектур (Multi-Core CPU, Multi-Core GPU), и учебно-тестовый полигон, применяемый для отладки пользовательских задач и используемый в образовательных целях.

Особое внимание уделяется вопросам развития платформы — за всё время существования был выполнен ряд модернизаций, в ходе которых неоднократно было расширено вычислительное поле, увеличена ёмкость системы хранения и обработки

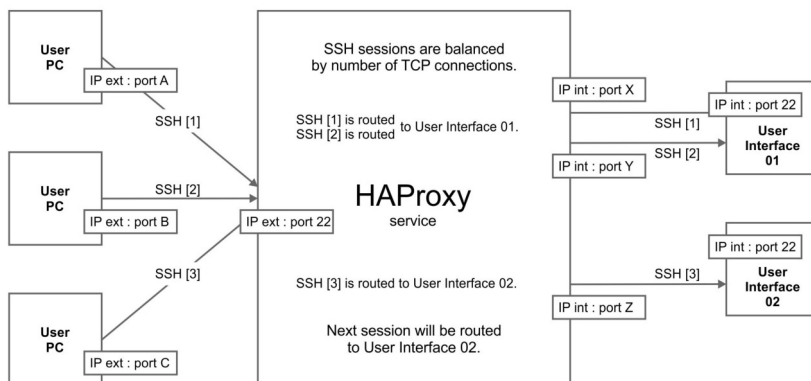


Рис. 2. Структура Пользовательского интерфейса

данных, разработаны и внедрены новые сервисы системного и прикладного уровней. По мере возникновения потребностей развивался и модернизировался Пользовательский интерфейс, пройдя путь от одного сервера, совмещённого с СХиОД, до полноценной компоненты платформы. Текущая структура ПИ приведена на рисунке 2.

Пользовательский интерфейс Гетерогенной вычислительной платформы Hybrid-IT построен на основе пула виртуальных машин (VM), размещённых в едином адресном пространстве вместе с ВП и СХиОД. Пользователи соединяются с ПИ по протоколу SSH, который маршрутизируется с помощью сервиса HAProxy, имеющего внешний IP-адрес (в сети ОИЯИ) и внутренний IP-адрес (в сети платформы). Балансировка нагрузки осуществляется автоматически, по количеству открытых TCP-соединений к каждой VM, таким образом, чтобы количество SSH-сеансов было равномерно распределено по всем VM.

Каждая VM предоставляет рабочую среду ПИ [3], в которой пользователь осуществляет свою рабочую деятельность. После соединения с VM пользователь получает доступ к своей домашней директории, может настраивать переменные программной среды (shell environment) с помощью пакета modulefiles для работы с доступными компиляторами и библиотечками, размещёнными в специализированном репозитории CVMFS, редактировать тексты программ, копировать файлы с данными, осуществлять подготовку исполняемых (бинарных) файлов. В этой среде пользователь взаимодействует с планировщиком SLURM, с помощью которого выполняет постановку задач в счётные очереди, а затем получает результаты счёта (лог файлы, выходные файлы).

### 5.1. Методология построения Пользовательского интерфейса

Общие принципы организации Пользовательского интерфейса вычислительного кластера можно сформулировать в виде ряда тезисов:

1. Пользовательский интерфейс должен состоять из нескольких серверов или виртуальных машин, размещённых в одном адресном пространстве с вычислительным полем и системой хранения и обработки данных.

2. Все компоненты вычислительного кластера, включая Пользовательский интерфейс, должны иметь IP-адреса из немаршрутизируемой сети (например, вида 10.0.0.0/8).
3. Доступ из внешней сети (например, локальной сети организации) должен осуществляться через единую точку входа по протоколу SSH (TCP порт 22), при этом все другие протоколы должны быть недоступны (все другие TCP порты должны быть закрыты).
4. Ключевым элементом Пользовательского интерфейса — является сервис проксирования SSH соединений (например, HAProxy), который обеспечивает балансировку нагрузки на серверы ПИ по количеству интерактивных SSH-сеансов.

## 6. Заключение

На основе анализа различных методов построения Пользовательского интерфейса и опыта эксплуатации Гетерогенной вычислительной платформы HybriLIT была разработана Методология построения Пользовательского интерфейса вычислительного кластера.

Приведённые общие принципы построения ПИ можно успешно применять для различных многопользовательских информационных систем, обеспечивая высокий уровень безопасности и балансировку нагрузки.

## Литература

1. Беляков Д.В., Бутенко Ю.А., Валя М., Воронцов А.С., Заикина Т.Н., Зуев М.И., Киракосян М.Х., Матвеев М.А., Подгайный Д.В., Стрельцова О.И., Торосян Ш.Г. Гетерогенная платформа HybriLIT // Бюллетень «Новости ОИЯИ». — 2019. — № 2. — С. 19-25.
2. Гетерогенная платформа «HybriLIT». — URL: <http://hlit.jinr.ru/> (Дата обращения: 01.03.2024).
3. Dmitry D.V., Nechaevskiy A.V., Pelevanuk I.S., Podgainy D.V., Stadnik A.V., Streltsova O.I., Vorontsov A.S., Zuev M.I. "Govorun" Supercomputer for JINR Tasks // CEUR Workshop Proceedings, ISSN:1613-0073, Изд.:CEUR Workshop Proceedings. — 2020. — Vol. 2772. — С. 1-12. - URL: <https://ceur-ws.org/Vol-2772/1-12-paper-1.pdf>

UDC 004.4

## Methodology for building an User Interface for a computing cluster

D. V. Belyakov

*Laboratory of Information Technologies  
Joint Institute for Nuclear Research  
Joliot-Curie 6, Dubna, Moscow region, 141980, Russia*

Email: [dmitry@jinr.ru](mailto:dmitry@jinr.ru)

Modern scientific research poses problems for scientists that require the usage of various methods and algorithms for their numerical solutions, the implementation of which is only possible using high performance and/or parallel computing. The usage of computing clusters or supercomputers for the numerical solutions of scientific problems has become a widespread practice in research and educational organizations.

The elements of any computing cluster are high-performance servers and fast interconnect, which form the main cluster's components — a computation field, a data storage and data processing system, and user interfaces. For its operation, a computing cluster requires the usage of a number of services - a resource management system (resource manager and task scheduler), an application software distribution system, and an authentication and authorization system. Additional services include a computational field's monitoring system and a system for collection/analyzing statistics of resource's usage (billing).

One of the main components of computing cluster is the User Interface, which is the link that organizes the work of users with application software, a resource management system, and a data storage and data processing system.

The work examines the functions of the User Interface and technical requirements for them. An analysis of typical methods of organization was carried out and a methodology for building an User Interface for computing cluster was developed. An implemented approach to organizing the work of users with computing cluster is presented using the example of the User Interface of the HybriLIT Heterogeneous Platform (LIT, JINR).

**Key words and phrases:** HPC, User Interface