МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «БАШКИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ КАФЕДРА ГЕОФИЗИКИ

КУРСОВАЯ РАБОТА ПО ПРОГРАММЕ МАГИСТРАТУРЫ

КАДЫРОВ АЛМАЗ ВЕНЕРОВИЧ

«РЕШЕНИЕ ЗАДАЧИ ПОДДЕРЖКИ ОЧЕРЕДИ ЗАДАЧ IBM LSF В КЛИЕНТЕ ЗАПУСКА РАСЧЕТОВ НА КЛАСТЕРЕ SCHEDULER»

Выполнил:

Магистрант 1 года очной формы обучения Направление подготовки – «Геология» Программа подготовки – «Цифровые технологии в петрофизике»

Руководитель:

старший преподаватель кафедры «Цифровые технологии в петрофизике»

Отру _____/ О.Р. Привалова

Консультант:

главный специалист

_____/ И.Ф. Сайфуллин

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
1 ОБЗОР IBM LSF.	4
1.1 Кластер	5
1.2 Задача	5
1.3 Слот задачи	6
1.4 Очередь	6
1.5 Ресурсы	7
1.6 Хосты	7
1.7 Хост отправки	8
1.8 Хост исполнения	8
1.9 Хост сервер	8
1.10 Хост клиент	9
1.11 Хост управления	9
2 ЗАДАЧА ПОДДЕРЖКИ ОЧЕРЕДИ ЗАДАЧ IBM LSF В КЛИЕНТЕ	
ЗАПУСКА РАСЧЕТОВ НА КЛАСТЕРЕ SCHEDULER	10
2.1 Установка и настройка LSF	10
2.2 Поддержка API для LSF в серверной части Scheduler	
2.3 Поддержка команд, направляемы напрямую из Scheduler на кластер	.15
2.4 Тестирование	18
ЗАКЛЮЧЕНИЕ	22
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ	23

ВВЕДЕНИЕ

IBM Spectrum LSF — это система очередей задач, позволяющая пользователям запускать задачи на кластере. Кластер состоит из множества вычислительных узлов, каждый из которых имеет набор процессоров и память. Пользователь отправляет задачу, в которой указана последовательность команд, которую он хочет запустить, вместе с описанием вычислительных ресурсов, необходимых для исполнения задачи: число узлов кластера, количество ядер процессора, необходимое количество оперативной памяти и необходимое время [1].

Система очередей задач позволяет распределить пользовательские задачи сети для расчетов гидродинамических моделей с различными запрашиваемыми ресурсами: кол-во ядер, кол-во и тип узлов.

Приложение клиент Scheduler позволяет пользователям рассчитывать на сервере кластере гидродинамические модели. В нем поддерживаются системы очередей: Torque, PBS Pro, Slurm. В рамках курсовой работы была поставлена задача поддержки системы очередей IBM Spectrum LSF, поскольку кластеры различаются и у них могут быть установлены различные системы очередей.

1 OБ3OP IBM LSF

Программное обеспечение IBM Spectrum LSF («LSF», сокращенно от load sharing facility — средства распределения нагрузки) является ведущим в обеспечением корпоративного LSF отрасли программным класса. распределяет работу по существующим разнородным компьютерным ресурсам для создания общей, масштабируемой и отказоустойчивой инфраструктуры, которая обеспечивает более быструю и надежную производительность рабочих нагрузок и снижает затраты. LSF балансирует нагрузку и распределяет ресурсы, а также обеспечивает доступ к этим ресурсам [2].

LSF предоставляет фреймворк управления ресурсами, которая учитывает необходимые ресурсы для задания, находит лучшие ресурсы для выполнения задания и отслеживает его выполнение. Задания всегда выполняются в соответствии с нагрузкой на хост и политикой планировщика.

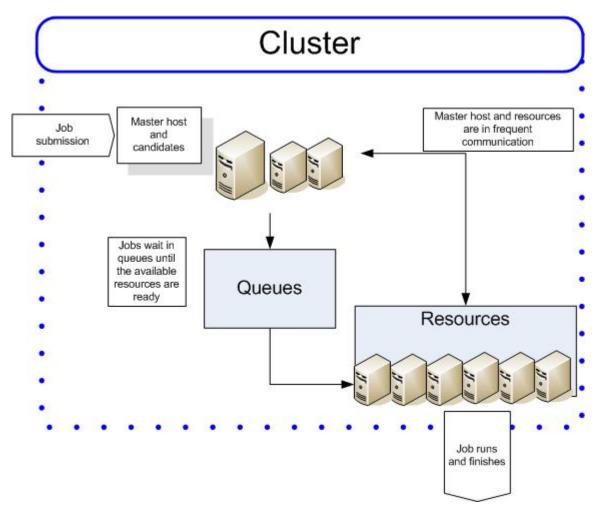


Рис. 1.1 – Кластер LSF

1.1 Кластер

Группа компьютеров (хостов), на которых запущен LSF, которые работают вместе как единое целое, объединяя вычислительную мощность, рабочую нагрузку и ресурсы. Кластер предоставляет образ одной системы для сети вычислительных ресурсов (рис. 1.1).

Хосты можно объединить в кластер несколькими способами. Кластер может содержать:

- Все хосты в единой административной группе;
- Все хосты в подсети.

1.2 Задача

Единица работы, выполняемая в системе LSF. Задача — это команда, которая отправляется LSF для выполнения. LSF планирует, контролирует и отслеживает задачу в соответствии с настроенными политиками.

Задачи могут представлять собой сложные задачи, сценарии моделирования, обширные вычисления или все, что требует вычислительной мощности.

1.3 Слот задачи

Слот задачи — это гнездо, которому в системе LSF назначается отдельная единица работы.

Хосты могут быть настроены с несколькими слотами, и вы можете отправлять задачи из очередей до тех пор, пока все слоты не будут заполнены. Вы можете соотнести слоты с общим количеством процессоров в кластере.

1.4 Очередь

Контейнер для рабочих мест во всём кластере. Все задачи ожидают в очередях, пока они не будут запланированы и отправлены на хосты.

Очереди не соответствуют отдельным хостам; каждая очередь может использовать все хосты серверов в кластере или заданное подмножество хостов серверов.

Когда вы отправляете задачу в очередь, вам не нужно указывать хост выполнения. LSF отправляет задачу на лучший доступный хост исполнения в кластере для выполнения этой задачи.

Очереди реализуют различные политики планирования задач и управления.

1.5 Ресурсы

Ресурсы — это объекты в вашем кластере, которые доступны для выполнения работы. Например, ресурсы включают, помимо прочего, хосты, слоты процессоров и лицензии.

1.6 Хосты

Хост — это отдельный компьютер в кластере.

У каждого хоста может быть более 1 процессора. Многопроцессорные хосты используются для выполнения параллельных задач. Многопроцессорный хост с единственной очередью считается отдельной машиной, в то время как коробка, полная процессоров, каждый из которых имеет свою собственную очередь процессов, рассматривается как группа отдельных машин [3].

Хосты в вашем кластере выполняют разные функции:

- Хост управления хост север LSF, который действует как всеобщий координатор для кластера, выполняя планирование и отправку всех заданий из очередей в хосты исполнения;
- Хост сервер хост, который отправляет и запускает задачи;
- Хост клиент хост, который только отправляет задачи и задания;
- Хост исполнения хост, на котором выполняются задачи и задания;
- Хост отправки хост, с которого отправляются задачи и задания.
 Команды:
- lsload просмотр нагрузки на хосты;
- lshosts просмотр информации о конфигурации хостов в кластере, включая количество процессоров, модель, тип и то, является ли хост клиентом или сервером;
- bhosts просмотр хостов пакетного сервера в кластере.

Совет: имена ваших хостов должны быть уникальными. Они не должны совпадать с именем кластера или какой-либо очередью, заданной для кластера.

1.7 Хост отправки

Хост, на котором задания отправляются в кластер.

Задания отправляются с помощью команды bsub или из приложения, которое использует API LSF.

Хосты клиенты и хосты серверы могут действовать как хосты отправки.

Команды:

- bsub отправить задачу;
- bjobs просмотр отправленных задач.

1.8 Хост исполнения

Хост, на котором выполняется задание. Может быть тем же, что и хост отправки. Все хосты исполнения являются хостами серверами.

Команды:

• bjobs — просмотр, где запущена задача.

1.9 Хост сервер

Хосты, которые могут отправлять и исполнять задания. Хост сервер запускает sbatchd для исполнения запросов к серверу и применения локальных политик.

Команды:

• lshosts — просмотр хостов серверов (server=Yes). Настройка: • Хосты сервера определяются в файле lsf.cluster.*cluster_name* путем указания значения 1 для server.

1.10 Хост клиент

Хосты, которые могут только отправлять задания в кластер. Хосты клиенты запускают команды LSF и действуют только как хосты отправки. Хосты клиенты не выполняют задания и не запускают демонов LSF (программы, работающие в фоновом режиме).

Команды:

- lshosts просмотр хостов клиентов (server=No). Настройка:
- Хосты клиенты определяются в файле lsf.cluster.cluster_name путем указания значения 0 для server.

1.11 Хост управления

Главный хост, где запускаются главный LIM и mbatchd. Хост сервер LSF, который действует как всеобщий координатор для этого кластера. В каждом кластере есть один главный хост, который выполняет планирование и отправку всех заданий из очередей в хосты исполнения. Если главный хост выходит из строя, другой сервер LSF в кластере становится главным хостом.

Все демоны LSF работают на главном хосте. LIM на главном хосте является главным LIM.

Команды:

- lsid просмотр имени главного хоста.
 Конфигурация:
- Главный хост это первый хост, указанный в файле lsf.cluster.cluster_name или определенный вместе с другими хостами кандидатами в главные хосты в LSF_MASTER_LIST в lsf.conf.

2 ЗАДАЧА ПОДДЕРЖКИ ОЧЕРЕДИ ЗАДАЧ IBM LSF В КЛИЕНТЕ ЗАПУСКА РАСЧЕТОВ НА КЛАСТЕРЕ SCHEDULER

Постановка задачи:

- Установить и настроить IBM LSF;
- Поддержать API для LSF в серверной части Scheduler;
- Поддержать команды, направляемы напрямую из Scheduler на кластер;
- Протестировать.

На блок-схеме на рис. 2.1 изображены отношения между элементами. Каждый элемент не знает о элементах за элементом, с которым он связан. Каждый элемент служит абстракцией.

Рис. 2.1 – Блок-схема

2.1 Установка и настройка LSF

Запланируйте установку, чтобы задать необходимые параметры для файла install.config [4].

Укажите главный администратор LSF (владеет файлами настроек и файлами лога LSF и EGO). Например,

LSF_ADMINS="lsfadmin"

Укажите общую директорию установки LSF. Например,

LSF_TOP="/usr/share/lsf"

Укажите хосты LSF (хост управления, хосты кандидаты в хосты управления, хосты серверы и хосты клиенты). Например,

LSF ADD SERVERS="hostm hostb hostc hostd"

LSF MASTER LIST="hostm hostd"

LSF_ADD_CLIENTS="hoste hostf"

Важно: не используйте имя какого-либо хоста, пользователя или группы пользователей в качестве имени вашего кластера.

Укажите хосты сервера LSF, которые являются кандидатами на роль хоста управления для кластера, если вы устанавливаете новый хост, который будет динамически добавлен в кластер. Например,

LSF_MASTER_LIST="hosta hostb"

Укажите имя кластера, содержащее не более 39 символов без пробелов. Например,

LSF CLUSTER NAME="cluster1"

Если вы устанавливаете LSF Standard Edition, выберите шаблон настройки для начальной настройки вашего нового кластера. Например,

CONFIGURATION_TEMPLATE="HIGH_THROUGHPUT"

Выберите один из следующих шаблонов в зависимости от типа задач, которые будет выполнять ваш кластер:

- DEFAULT укажите этот шаблон для кластеров со смешанной рабочей нагрузкой. Эта конфигурация может обслуживать различные типы рабочих нагрузок с хорошей производительностью, но не настроена для конкретного типа кластера;
- PARALLEL укажите этот шаблон для кластеров, в которых выполняются большие параллельные задачи. Эта конфигурация предназначена для длительных параллельных задач, а не для кластеров, которые в основном выполняют короткие задания из-за более длительного времени отчетности для каждой задачи;
- HIGH_THROUGHPUT этот шаблон используется для кластеров, которые в основном выполняют короткие задания, где более 80\% заданий завершаются в течение одной минуты. Такая высокая текучесть задач требует, чтобы LSF был более отзывчивым и быстродействующим, но по мере того, как демоны становятся более загруженными, будет использовать больше ресурсов.

Значение полей:

- LSF_ADMINS: имена пользователей администраторов LSF;
- LSF TOP: полный путь директории установки LSF;
- LSF_ADD_SERVERS: хосты сервера, которые могут ставить задания в очередь и выполнять задания;
- LSF_MASTER_LIST: главный хост, который действует как всеобщий координатор для кластера. В каждом кластере есть один главный узел, который выполняет планирование и отправку всех заданий из очередей в хосты исполнения;
- LSF_ADD_CLIENTS: хосты клиенты, которые могут только ставить задания в очередь;
- LSF_CLUSTER_NAME: имя кластера LSF;
- CONFIGURATION_TEMPLATE: шаблон конфигурации для определения начальной конфигурации нового кластера [2, 3].

Создание пользователя для администратора LSF и запуск установки LSF:

\$ sudo -i

adduser lsfadmin

./lsfinstall -f install.config

Запуск LSF:

source /usr/share/lsf/conf/profile.lsf

lsfstartup

Введите следующие команды, чтобы использовать кластер LSF, установленный в каталоге /usr/share/lsf, и настроить демоны LSF для автоматического запуска во время запуска машины [5]:

cd /usr/share/lsf/10.1/install

./hostsetup --top="/usr/share/lsf" --boot="y"

2.2 Поддержка API для LSF в серверной части Scheduler

Shell-скрипты формируют файл с информацией для запуска задачи, который запускается командой bsub — она считывает файл задачи, параметры в виде строк начинающихся с #BSUB и запрашивает в системе очередей необходимые ресурсы, после чего задача ставится в очередь и, когда запрошенные ресурсы освобождаются, исполняется остальная часть.

јоb_file — это shell-скрипт с прописанными директивами #BSUB в начале файла.

За основу взяты шаблоны задач и скрипты bash системы очередей Тогque для поддержки LSF. Шаблоны задач и скрипты Torque переписаны для LSF. Созданы shell скрипты, которые формируют файл с информацией для запуска задачи

```
run_rnkim_decomp_mpi_lsf.sh,
run_rnkim_mpi_lsf.sh,
run_rnkim_omp_lsf.sh
и шаблоны задач
template_rnkim_decomp_lsf,
template_rnkim_decomp_mpi_lsf,
template_rnkim_mpi_lsf,
template_rnkim_omp_lsf
для LSF.
Переписывание скриптов bash с Torque на LSF:
Отправка задачи job_file в очередь
qsub job_file
-->
bsub < job_file
```

В параметр -т — конкретные хосты, группы хостов, вычислительные единицы — передаются теги или типы узлов.

```
$NODETYPE
```

```
-->
_tmplNODETYPE_="#BSUB -m \"$NODETYPE\""
```

Переписывание шаблонов задач с Torque на LSF:

В параметр -n — задает кол-во тасков в задаче — передается кол-во ядер в узле $\mbox{tmplCORES}$.

В параметр -R — задает строку ресурсов — передается кол-во узлов 1 [6].

```
#PBS -l nodes=1_tmplNODETYPE_:ppn=_tmplCORES_
-->
#BSUB -n _tmplCORES_ -R "span[hosts=1]"
tmplNODETYPE
```

В параметр -n — задает кол-во тасков в задаче — передается кол-во всех ядер в узлах _tmplTOTALCORES_.

В параметр -R — задает строку ресурсов — передается кол-во ядер на узел _tmplCORES_.

```
#PBS -l nodes=_tmplNNODES__tmplNODETYPE_:ppn=_tmplCORES_
-->
#BSUB -n _tmplTOTALCORES_ -R "span[ptile=_tmplCORES_]"
_tmplNODETYPE_
```

TOTALCORES = NNODES * CORES

В параметр -notify — запрашивает уведомление пользователя, когда задание достигает любого из указанных состояний — передаются состояния программы.

В параметр -R — отправляет письмо по указанному адресу электронной почты — передается адрес электронной почты.

```
#PBS -m ea
#PBS -M <usermail>
-->
#BSUB -notify "exit done"
#BSUB -u <usermail>
```

В параметр -R — присваивает указанное имя заданию — передается имя модели.

В параметр -W — устанавливает ограничение времени выполнения задания — передается период 150 часов.

В параметр -cwd — задает текущую рабочую директорию для выполнения задания — передается путь директории.

```
#PBS -N _tmplMODEL_

#PBS -l walltime=150:00:00

#PBS -d _tmplDIR_
-->

#BSUB -J _tmplMODEL_

#BSUB -W 150:00

#BSUB -cwd _tmplDIR
```

2.3 Поддержка команд, направляемы напрямую из Scheduler на кластер

Команды в Scheduler, относящиеся к конкретной системе очередей, хранятся в значениях ключей в словаре (тип данных на Python). Значениям соответствуют либо ссылки на исполняемые на сервере скрипты, либо команды для системы очередей, либо ссылки на методы обработки. Для LSF добавлены следующие значения ключей:

```
QsysCMD.RUN_OMP:
```

"\$RNKIMPATH/scripts/run_rnkim_omp_lsf.sh",

QsysCMD.RUN_MPI:

"\$RNKIMPATH/scripts/run_rnkim_decomp_mpi_lsf.sh",

QsysCMD.RUN_MPI_ADV:

"\$RNKIMPATH/scripts/run_rnkim_mpi_lsf.sh",

QsysCMD.DEL_TASK: "bkill",

QsysCMD.GET_STAT: "bjobs -json -o 'jobid user stat job_name submit time start time finish time error file output file effective resreq slots'",

QsysCMD.GET_STAT_MTHD: lambda str_jobs: f"bjobs -json -o 'jobid user stat job_name submit_time start_time finish_time error_file output_file effective resreq slots' {str_jobs}",

QsysCMD.PARSE_ID_MTHD: lambda strout: int(strout[strout.find('<') + 1:strout.find('>')]),

```
QsysCMD.UPDT_JSTAT_MTHD: self._update_jstats_lsf
```

Метод _update_jstats_lsf обновляет статус моделей. Парсит JSON статуса модели и вызывает метод _pars_job_json_lsf для парсинга значений полей JSON статуса.

Метод _update_jstats_lsf парсит поля с значениями у JSON статуса и записывает ключ 'имя модели' со значением словарь состояния:

```
"JOBID":"1363",
       "USER":"vagrant",
       "STAT": "EXIT",
       "JOB NAME":"MODEL.DATA",
       "SUBMIT_TIME":"Jun 7 08:19",
       "START TIME":"Jun 7 08:19",
       "FINISH TIME":"Jun 7 08:19 L",
       "ERROR FILE":"",
       "OUTPUT FILE":"",
       "EFFECTIVE RESREQ": "select[type == local] order[r15s:pg]
span[ptile=2]",
       "SLOTS":"2"
     }
     -->
     model name:
       {
         JobStat.ACC NAME: str,
         JobStat.JOB NAME: str,
```

```
JobStat.OUT_PATH: str,
JobStat.ERR_PATH: str,
JobStat.JOB_STAT: ModelState,
JobStat.NUM_NODES: int,
JobStat.QUEUE_TIME: datetime,
JobStat.START_TIME: datetime,
JobStat.COMPL_TIME: datetime
```

Значение полей:

- JOBID: идентификатор задачи, является порядковым номером задачи;
- USER: имя пользователя, который отправил задачу в очередь;
- STAT: состояние исполнения задачи;
- JOB_NAME: имя задачи, содержащее имя модели;
- SUBMIT TIME: дата отправки задачи в очередь;
- START_TIME: дата запуска задачи;
- FINISH TIME: дата завершения задачи;
- ERROR_FILE: путь к файлу с выводом задачи;
- OUTPUT_FILE: путь к файлу с сообщениями задачи о ошибках;
- EFFECTIVE_RESREQ: запрошенные ресурсы для задачи: один узел или кол-во ядер в каждом узле;
- SLOTS: кол-во всех запрошенных ядер;
- model_name: имя модели;
- JobStat.ACC_NAME: имя пользователя, который отправил модель на кластер;
- JobStat.JOB_NAME: имя задачи расчета, содержащее имя модели;
- JobStat.OUT_PATH: путь к файлу с выводом расчета модели;
- JobStat.ERR_PATH: путь к файлу с сообщениями расчета модели о ошибках;
- JobStat.JOB_STAT: состояние расчета модели;
- JobStat.NUM_NODES: кол-во узлов, запрошенных для модели;

- JobStat.QUEUE TIME: дата отправки расчета модели в очередь;
- JobStat.START TIME: дата запуска расчета модели;
- JobStat.COMPL_TIME: дата завершения расчета модели.

2.4 Тестирование

Использован программный продукт виртуализации VirtualBox для тестирования. Сервер установлен на виртуальной машине VirtualBox с операционной системой Ubuntu Server 18.04. Клиент запускался в исходной машине и связывался с виртуальной машиной с сервером.

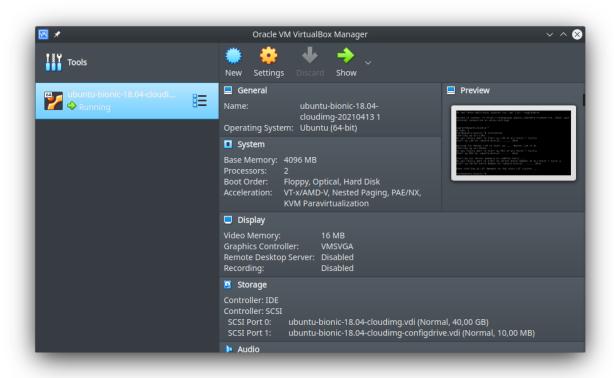


Рис. 2.2 – VirtualBox

Рис. 2.3 – Виртуальная машина. Изображен запуск LSF

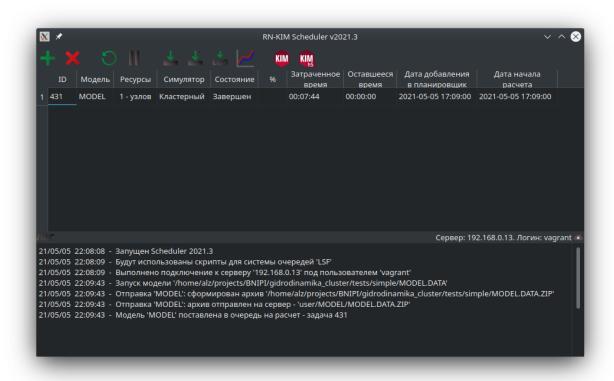


Рис. 2.4 – Скриншот Scheduler. Тип расчета модели: кластерный

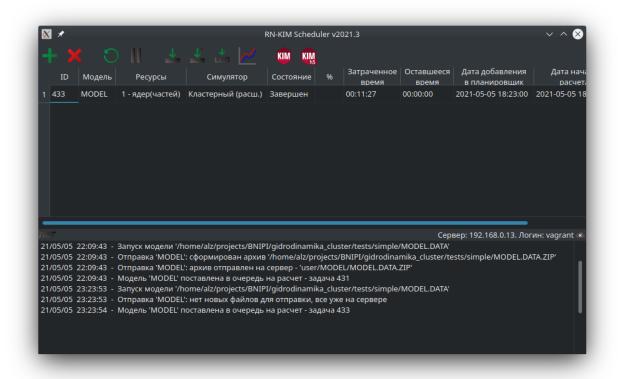


Рис. 2.5 – Скриншот Scheduler. Тип расчета модели: кластерный (расш.)

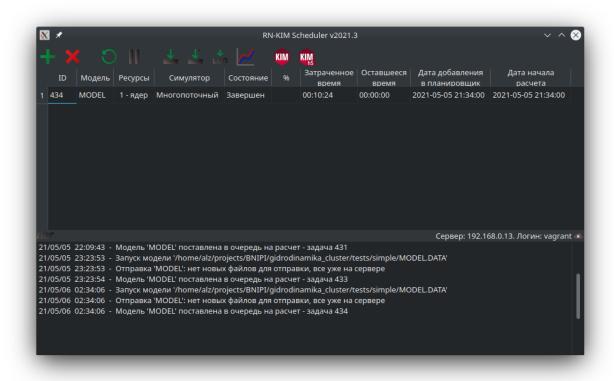


Рис. 2.6 – Скриншот Scheduler. Тип расчета модели: многопоточный

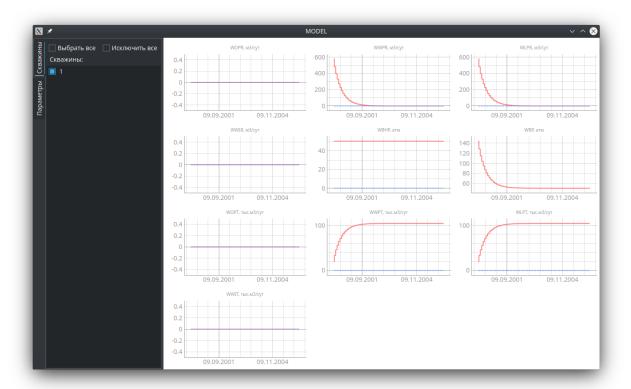


Рис. 2.7 – Скриншот Scheduler. Рассчитанные кривые модели

ЗАКЛЮЧЕНИЕ

Поддержана очередь задач IBM LSF в клиенте Scheduler: созданы скрипты bash и шаблоны задач для поддержки API для LSF в серверной части Scheduler и поддержаны команды на Python, направляемые напрямую из Scheduler на кластер. Создана и предоставлена виртуальная машина VirtualBox с операционной системой Ubuntu Server 18.04 с сервером. Создана и предоставлена документация по настройке LSF.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ

- 1. LSF User Manual [Электронный ресурс]. Режим доступа: https://hpc.llnl.gov/banks-jobs/running-jobs/lsf-user-manual. (Дата обращения: 21.06.2021).
- 2. Introduction to IBM Spectrum LSF [Электронный ресурс]: IBM Spectrum LSF V10.1 documentation. Режим доступа: https://www.ibm.com/docs/en/spectrum-lsf/10.1.0?topic=overview-lsf-introduction. (Дата обращения: 21.06.2021).
- Hosts [Электронный ресурс]: About IBM Platform LSF. Режим доступа:
 https://www.bsc.es/support/LSF/9.1.2/lsf_users_guide/hosts_about.html. (Дата обращения: 21.06.2021).
- 4. Planning your installation [Электронный ресурс]: IBM Spectrum LSF V10.1 documentation. Режим доступа: https://www.ibm.com/docs/en/spectrum-lsf/10.1.0?topic=linux-planning-your-installation. (Дата обращения: 21.06.2021).
- 5. Configuring a cluster [Электронный ресурс]: IBM Spectrum LSF V10.1 documentation. Режим доступа: https://www.ibm.com/docs/en/spectrum-lsf/10.1.0?topic=linux-configuring-cluster. (Дата обращения: 21.06.2021).
- 6. bsub [Электронный ресурс]: IBM Spectrum LSF command reference. Режим доступа: https://www.ibm.com/docs/en/spectrum-lsf/10.1.0? topic=reference-bsub. (Дата обращения: 21.06.2021).