

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»

Допущено к защите:

Заведующий кафедрой геофизики,

д.т.н., профессор

_____ / Р.А. Валиуллин

«__» _____ 20__ г.

Руководитель:

д.ф.-м.н., старший преподаватель

_____ / О.Р. Привалова

Консультант:

д.ф.-м.н., главный специалист

_____ / И.Ф. Сайфуллин

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 ПОСТАНОВКА ЗАДАЧИ	5
2 РЕШЕНИЕ	6
2.1 Установка и настройка LSF	6
2.2 Поддержка API для LSF в серверной части Scheduler	7
2.3 Поддержка команд, направляемых напрямую из Scheduler на кластер	8
2.4 Тестирование	10
ЗАКЛЮЧЕНИЕ	14
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	15

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: время, процессор, ядра, узлы [1].

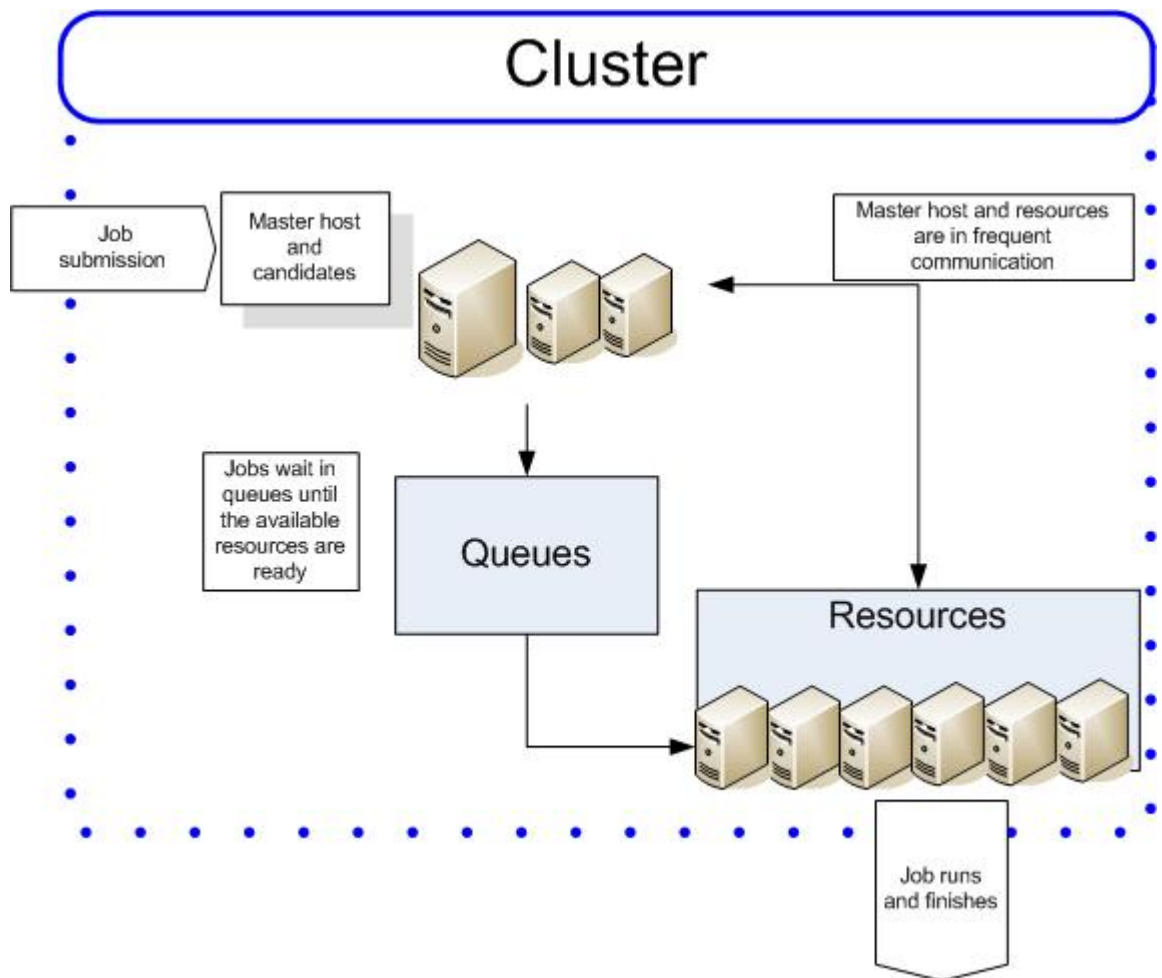


Рис. 1: Кластер LSF

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 ПОСТАНОВКА ЗАДАЧИ

Постановка задачи:

1. Установить и настроить IBM LSF;
2. Поддерживать API для LSF в серверной части Scheduler;
3. Поддерживать команды, направляемые напрямую из Scheduler на кластер;
4. Протестировать.

На блок-схеме 1.1 изображены отношения между элементами. Каждый элемент не знает о элементах за элементом, с которым он связан. Каждый элемент служит абстракцией.

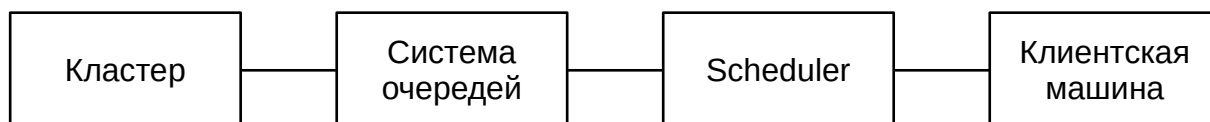


Рис. 1.1: Блок-схема

2 РЕШЕНИЕ

2.1 Установка и настройка LSF

Настройка значений полей в конфиг-файле `install.config` [2]:

```
LSF_ADMINS="lsfadmin"  
LSF_TOP="/usr/share/lsf"  
LSF_ADD_SERVERS="hostm hostb hostc hostd"  
LSF_MASTER_LIST="hostm hostd"  
LSF_ADD_CLIENTS="hoste hostf"  
LSF_CLUSTER_NAME="cluster1"  
CONFIGURATION_TEMPLATE="HIGH_THROUGHPUT"
```

Пояснение полей:

LSF_ADMINS: имена пользователей администраторов LSF;

LSF_TOP: полный путь директории установки LSF;

LSF_ADD_SERVERS: узлы, которые могут ставить задания в очередь и выполнять задания;

LSF_MASTER_LIST: узел сервера LSF, который действует как всеобщий координатор для кластера. В каждом кластере есть один главный узел, который выполняет планирование и отправку всех заданий из очереди в узлы для выполнения;

LSF_ADD_CLIENTS: узлы, которые могут только ставить задания в очередь;

LSF_CLUSTER_NAME: имя кластера LSF;

CONFIGURATION_TEMPLATE: шаблон конфигурации для определения начальной конфигурации нового кластера [3, 4].

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
```

2.2 Поддержка API для LSF в серверной части Scheduler

За основу взяты шаблоны задач и скрипты bash системы очередей Torque для поддержки LSF. Шаблоны задач и скрипты Torque переписаны для LSF. Созданы скрипты bash

```
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:

```
qsub job_file
-->
bsub < job_file
```

Переписывание шаблонов задач с Torque на LSF:

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

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

```

_tplNODETYPE_="#BSUB -m \"$NODETYPE\"
TOTALCORES = NNODES * CORES

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

#PBS -N _tplMODEL_
#PBS -l walltime=150:00:00
#PBS -d _tplDIR_
-->
#BSUB -J _tplMODEL_
#BSUB -W 150:00
#BSUB -cwd _tplDIR_

```

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:

Ключ:	Значение
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
}

```

2.4 Тестирование

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

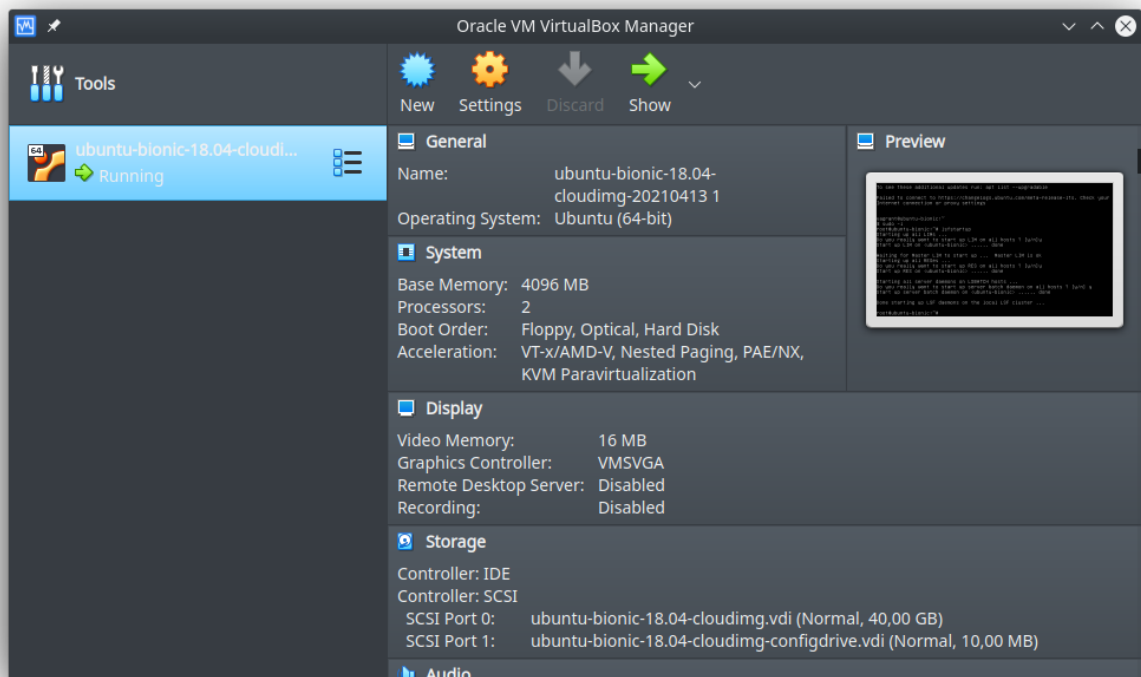
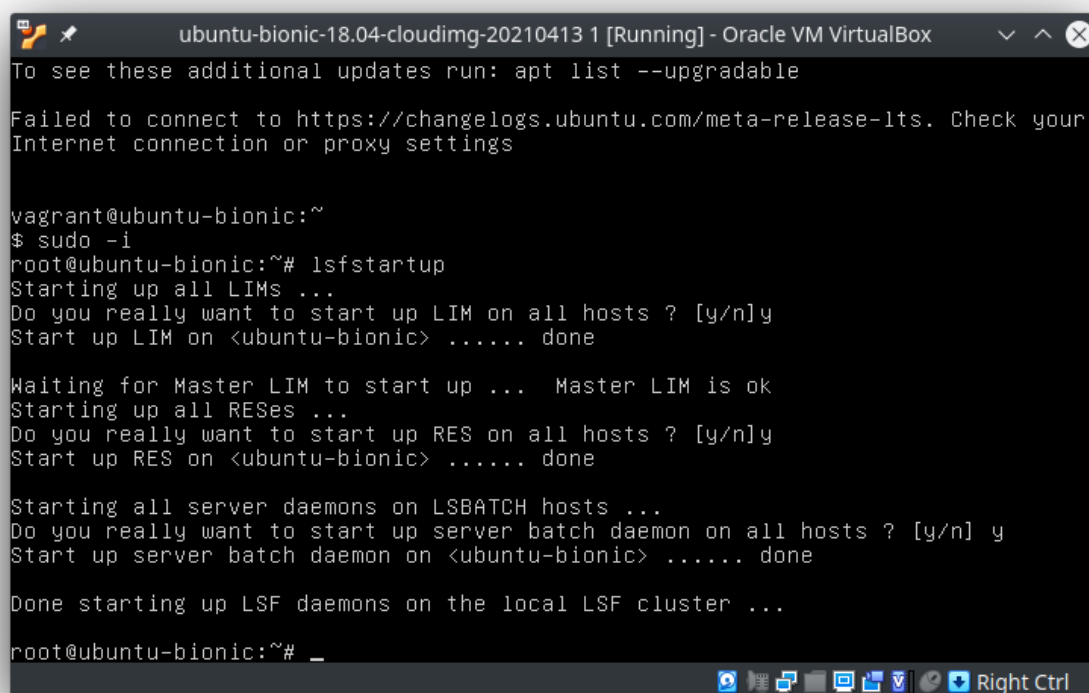


Рис. 2.1: VirtualBox

A terminal window titled 'ubuntu-bionic-18.04-cloudimg-20210413 1 [Running] - Oracle VM VirtualBox'. The terminal shows the execution of 'lsfstartup' as root. It prompts for confirmation to start LIMs and RESes on all hosts, both of which are confirmed with 'y'. It also prompts for confirmation to start the server batch daemon on all hosts, confirmed with 'y'. The process concludes with 'Done starting up LSF daemons on the local LSF cluster ...'.

```
ubuntu-bionic-18.04-cloudimg-20210413 1 [Running] - Oracle VM VirtualBox
To see these additional updates run: apt list --upgradable

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your
Internet connection or proxy settings

vagrant@ubuntu-bionic:~
$ sudo -i
root@ubuntu-bionic:~# lsfstartup
Starting up all LIMs ...
Do you really want to start up LIM on all hosts ? [y/n]y
Start up LIM on <ubuntu-bionic> ..... done

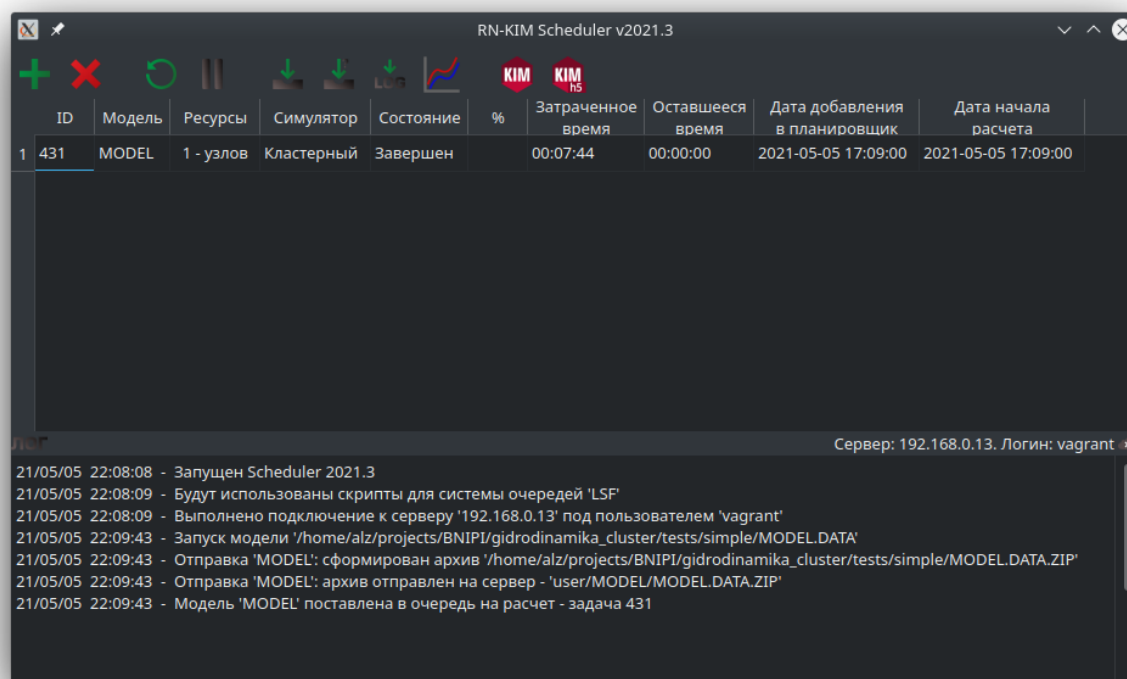
Waiting for Master LIM to start up ... Master LIM is ok
Starting up all RESeS ...
Do you really want to start up RES on all hosts ? [y/n]y
Start up RES on <ubuntu-bionic> ..... done

Starting all server daemons on LSBATCH hosts ...
Do you really want to start up server batch daemon on all hosts ? [y/n] y
Start up server batch daemon on <ubuntu-bionic> ..... done

Done starting up LSF daemons on the local LSF cluster ...

root@ubuntu-bionic:~# _
```

Рис. 2.2: Виртуальная машина. Изображен запуск LSF

The image shows the RN-KIM Scheduler v2021.3 interface. At the top is a toolbar with icons for adding, deleting, refreshing, pausing, downloading, and uploading. Below this is a table with columns: ID, Модель, Ресурсы, Симулятор, Состояние, %, Затраченное время, Оставшееся время, Дата добавления в планировщик, and Дата начала расчета. A single task with ID 431 is listed, with state 'Завершен'. At the bottom is a log window showing the execution steps of the scheduler, including connection to the server and file transfers.

ID	Модель	Ресурсы	Симулятор	Состояние	%	Затраченное время	Оставшееся время	Дата добавления в планировщик	Дата начала расчета
1 431	MODEL	1 - узлов	Кластерный	Завершен		00:07:44	00:00:00	2021-05-05 17:09:00	2021-05-05 17:09:00

Сервер: 192.168.0.13. Логин: vagrant

21/05/05 22:08:08 - Запущен Scheduler 2021.3
21/05/05 22:08:09 - Будут использованы скрипты для системы очередей 'LSF'
21/05/05 22:08:09 - Выполнено подключение к серверу '192.168.0.13' под пользователем 'vagrant'
21/05/05 22:09:43 - Запуск модели '/home/alz/projects/BNIP/гидродинамика_cluster/tests/simple/MODEL.DATA'
21/05/05 22:09:43 - Отправка 'MODEL': сформирован архив '/home/alz/projects/BNIP/гидродинамика_cluster/tests/simple/MODEL.DATA.ZIP'
21/05/05 22:09:43 - Отправка 'MODEL': архив отправлен на сервер - 'user/MODEL/MODEL.DATA.ZIP'
21/05/05 22:09:43 - Модель 'MODEL' поставлена в очередь на расчет - задача 431

Рис. 2.3: Скриншот Scheduler. Тип расчета модели: кластерный

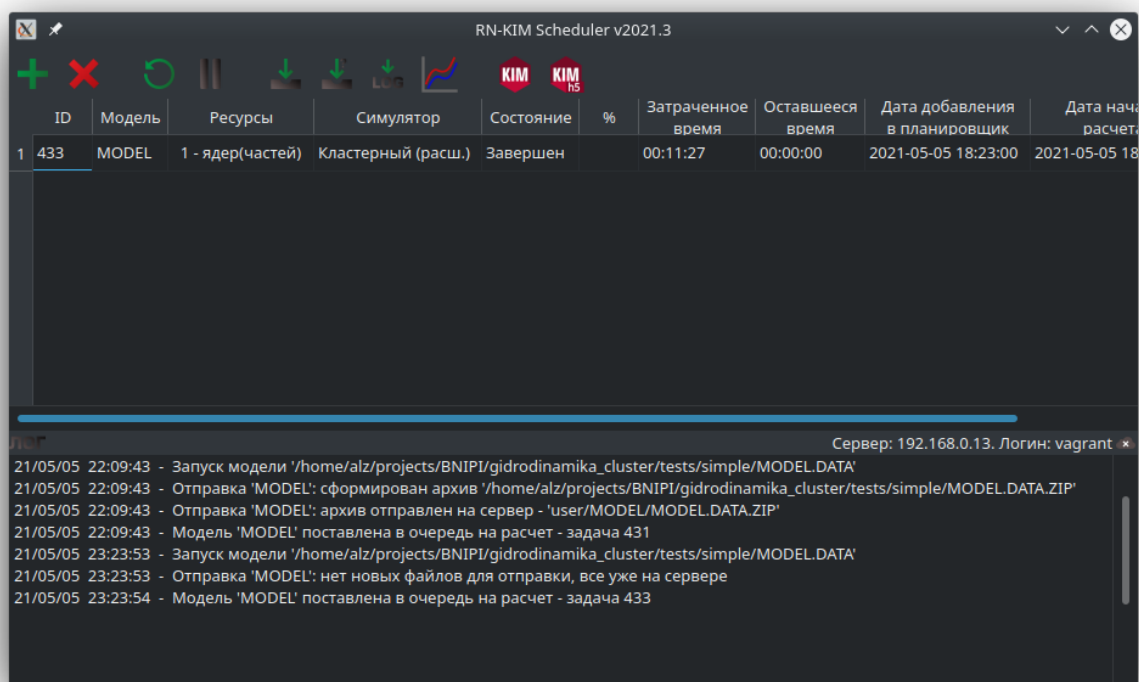


Рис. 2.4: Скриншот Scheduler. Тип расчета модели: кластерный (расш.)

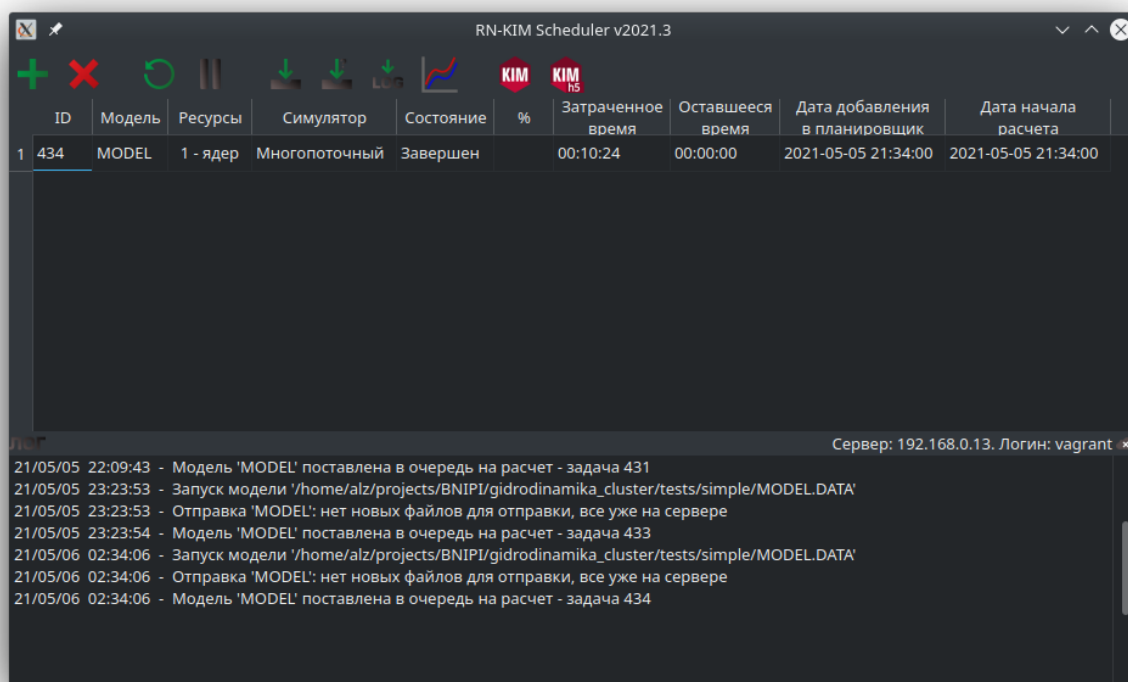


Рис. 2.5: Скриншот Scheduler. Тип расчета модели: многопоточный

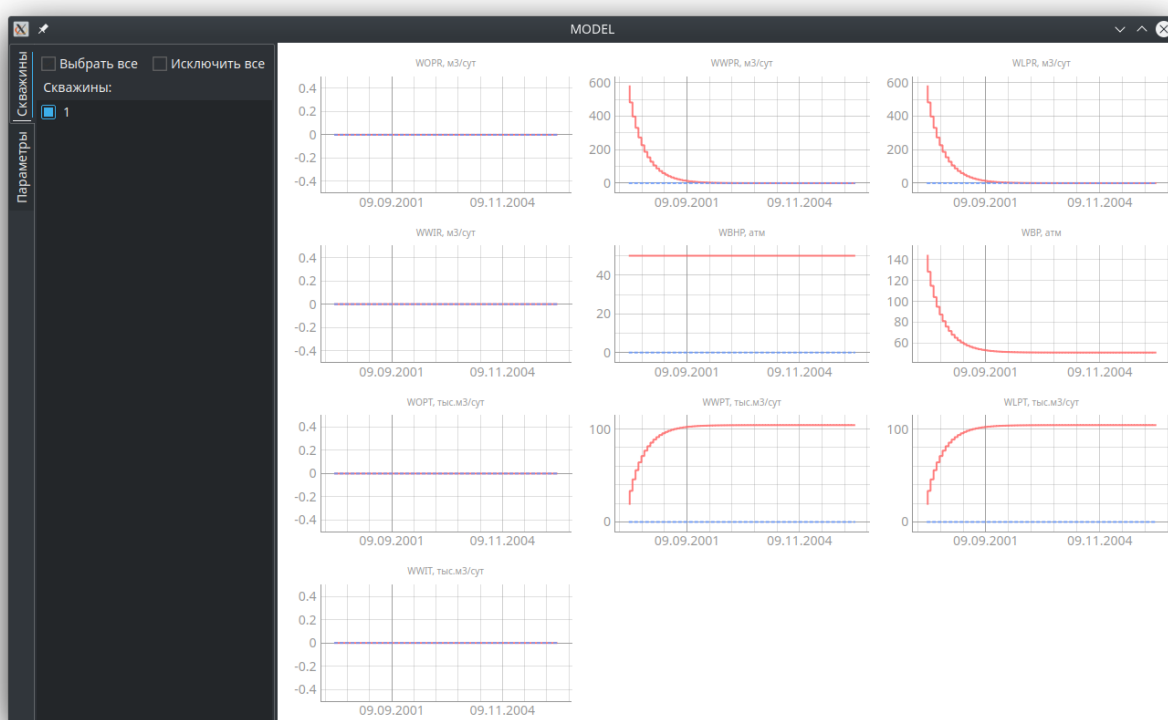


Рис. 2.6: Скриншот Scheduler. Рассчитанные кривые модели

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 поддерживаются команды на Python, направляемые напрямую из Scheduler на кластер. Создана и предоставлена виртуальная машина VirtualBox с операционной системой Ubuntu 18.04 с сервером. Создана и предоставлена документация по настройке LSF.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. LSF User Manual [Электронный ресурс] / — URL: <https://hpc.llnl.gov/banks-jobs/running-jobs/lsf-user-manual>
2. IBM Spectrum LSF V10.1 documentation: Planning your installation [Электронный ресурс] / — URL: <https://www.ibm.com/docs/en/spectrum-lsf/10.1.0?topic=linux-planning-your-installation>
3. IBM Spectrum LSF V10.1 documentation: Introduction to IBM Spectrum LSF [Электронный ресурс] / — URL: <https://www.ibm.com/docs/en/spectrum-lsf/10.1.0?topic=overview-lsf-introduction>
4. About IBM Platform LSF: Host types and host models [Электронный ресурс] / — URL: https://www.bsc.es/support/LSF/9.1.2/lsf_users_guide/index.htm?host_types_models_about.html~main