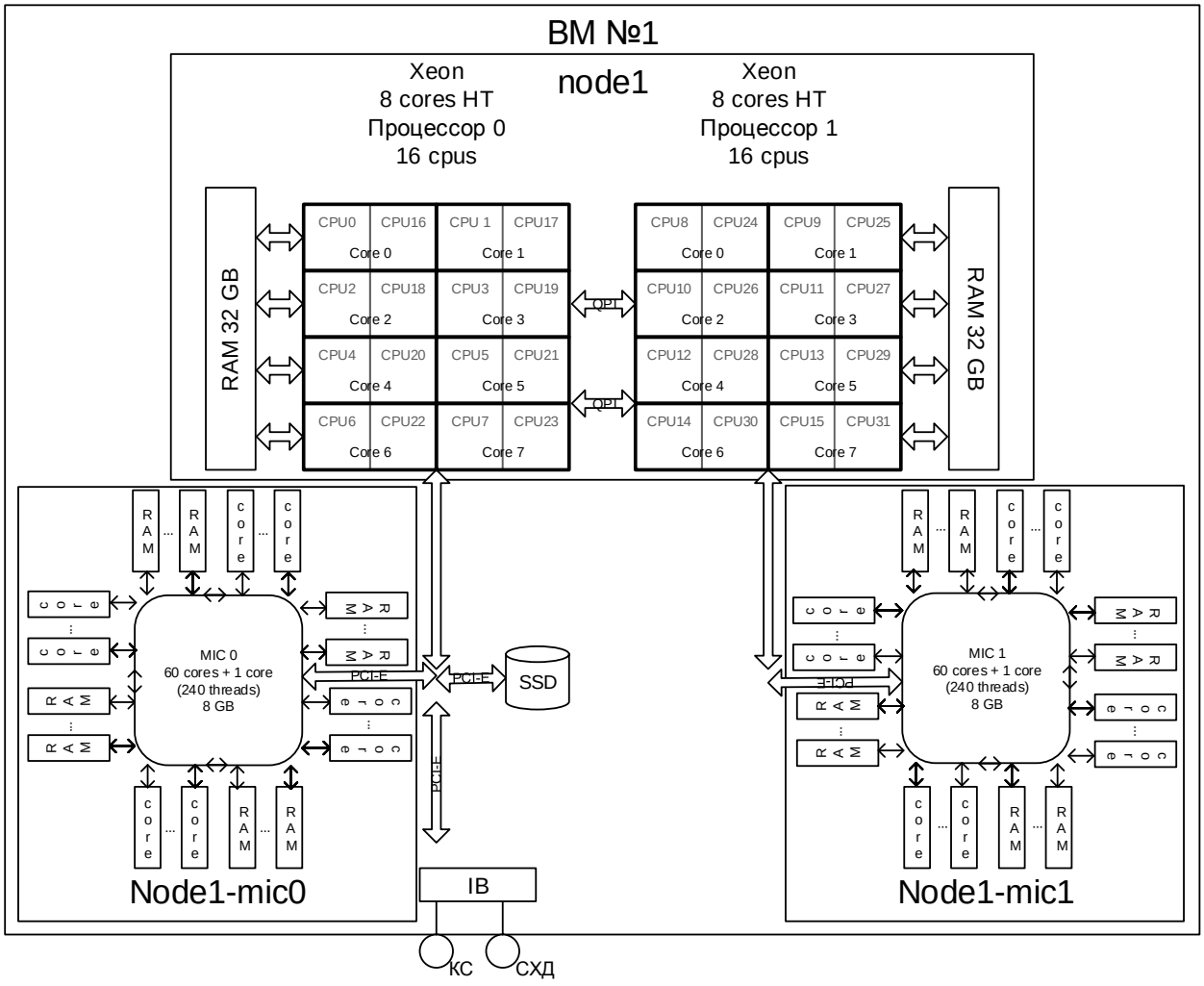
Описание интерфейса пользователя,
предназначенного для работы с интеловской гибридной архитектурой
суперЭВМ (СК), где вместе с процессорами **Intel Xeon** используются
сопроцессоры **Intel Xeon Phi**.

Описание СК МВС-10П

Вычислительный Модуль (ВМ) содержит 2 процессора (процессорных чипа) Xeon E5-2690 (далее Xeon) с 8-ю ядрами каждый (с hypertreading – 16 логических cpus) и 2 сопроцессора Intel Xeon Phi (далее MIC), имеющих по 61 ядра (с поддержкой 4-х одновременных потока (threads) на ядро. Минимальный ресурс, выдаваемый параллельному заданию, - 1 ВМ.



Сервер очередей различает 2 варианта Вычислительного Узла (ВУ):
- host-узел с 16-ю процессорами (cpus) Xeon (2 процессора по 8 ядер, 32 логических процессора cpus) (nodeX).
- MIC-узел с 4-мя процессорами (cpus) MIC (1 процессор MIC с 61 ядром) (nodeX-micY)

Без указания об использовании MIC процессоров, сервер очередей распределяет MPI-задачу только на Xeon часть ВМ, т.е. процессоры nodeX. При использовании процессоров MIC, сервер очередей распределяет MPI-

задачу сначала на Xeon часть первого ВМ (nodeX), потом на первый MIC (nodeX-mic0), потом на 2-й MIC (nodeX-mic1), потом на следующий ВМ nodeX+1, nodeX+1.mic0, nodeX+1.mic1, и т.д.

Таблица 1

Параметр	Значение для ВМ	
Процессоры	2 x Xeon E5-2690	2 x Intel Xeon Phi 7110X
Частота	2,9 ГГц	1,1 ГГц
Теоретическая пиковая производительность одного процессора	2 x 185,6 GFlops	2 x 1,0736 TFlops
Оперативная память	64 ГБ	2 x 8 ГБ

Таблица 2

Параметр	Значение для СК	
Вычислительный модуль ВМ	2 x Xeon (*nodeX*)	2 x MIC (*nodeX-mic0,nodeX-mic1*)
Коммуникационная сеть (MPI)	Между ВМ Infiniband FDR 56Гбит/с, fat-tree, неблокируемая	Внутри ВМ PCI-e, Intel QPI
MPI-процессов на ВУ по-умолчанию	**16**	**4 каждый MIC**
Доступная оперативная память на MPI-процесс	<4 ГБ	<2ГБ
Дисков	1 SSD 120ГБ. Недоступно для пользователя	
Транспортная сеть (files)	Через IB+10Gb Ethernet	Через PCI, IB+10Gb Ethernet
Системы хранения:	/home1 (/nethome)	NetApp HA cluster узел 1
	/home2	NetApp HA cluster узел 2
	/home3 (/gpfs/NETHOME)	NetApp HA cluster узел 2
	/home4	NetApp parallel cluster

Частоту процессора Intel Xeon Phi 7110X можно узнать командой на ВМ */opt/intel/mic/bin/micinfo*

Наглядное расположение процессорных ядер можно показать утилитой на ВМ
/opt/software/intel/impi/4.1.1.036/bin64/cpuinfo

Варианты запуска задания пользователя

Для работы через очередь СУППЗ необходимо загрузить модуль СУППЗ:
module load launcher/suppz

1) Если не используются MIC в режиме MPI, то привычная команда остаётся прежней:

```
mpirun –np 32 myprog
```

Программа myprog транслирована для Xeon, запускается на 32-х процессорах, занимая 2 ВМ. MPI-процессы запускаются только на Xeon, использование MIC возможно в режиме offload. 2 ВМ будут зарезервированы только для запущенной задачи, и два MIC в каждом узле будут доступны только для этой задачи, и не будут распределяться другим.

Используя конфигурационный файл со списком ВУ и количеством MPI-процессов на узел (`-machinefile`) можно изменить число MPI-процессов на один ВУ:

```
mpirun –np 32 –machinefile hosts myprog
```

файл `hosts`:

```
node1:8
node2:8
node3:8
node4:8
```

На каждом ВУ будет запущено по 8 MPI-процессов.

Тоже можно сделать с указанием опции `-ppn`

```
mpirun –np 32 –ppn 8 myprog
```

2) Что бы запустить задачу на MIC, нужно собрать два варианта кода программы:

```
mpicc –o myprog myprog.c
mpicc –o myprog.mic –mmic myprog.c
```

Один будет запущен на Xeon (`myprog`), другой на MIC (`myprog.mic`).
Запуск задачи производится с опцией `–mic`:

```
mpirun –np 48 –mic myprog
```

Программа займет 2 ВМ, разместив 32 процессов на Xeon (`myprog`) и по 4 процесса на каждом MIC (`myprog.mic`). На первого ВМ - 16 процессов Xeon и по 4 процесса на каждом MIC, на втором ВМ 16 процессов Xeon и по 4 на каждом MIC.

Количество процессов на ВУ можно изменить с помощью опции `–machinefile:`

```
mpirun -np 20 –machinefile hosts –mic myprog
```

hosts:

```
node1:8
node2:1
node3:1
node4:8
node5:1
node6:1
```

На Xeon первого ВМ будет запущено 8 процессов, на MIC первого выданного ВМ (nodeX-mic0) – 1 процесс, на втором MIC (nodeX-mic1) – 1 процесс, на втором ВМ 8 процессов на Xeon, по 1-му процессу на каждом MIC. В нашем случае node1,2,3 – первый ВМ, в котором node1 – ВУ на Xeon (nodeX), node2 – ВУ на 1-м MIC (nodeX-mic0), node3 – ВУ на 2-м MIC (nodeX-mic1), node4,5,6 – второй ВМ.

Количество процессов на ВУ можно изменить с помощью опций `–ppn`, `-mic-ppn`:

```
mpirun -np 20–ppn 8 –mic –mic-ppn 1 myprog
```

Постфикс исполняемого файла на MIC “.mic” можно переопределить с помощью ключа –mic-postfix. По умолчанию ключ равен “.mic”, если переменная окружения I_MPI_MIC_POSTFIX не определена:

```
mpirun -np 20 -mic –mic-postfix .mmm a.out

export I_MPI_MIC_POSTFIX=”.mmm”
mpirun -np 20 -mic a.out
```

В machine-файле можно указывать нулевое количество процессов, например:

hosts:

```
node1:8
node2:1
```

```
node3:0
node4:8
node5:1
node6:0

mpirun -np 18 –machinefile hosts –mic myprog
```

Процессов на nodeX-mic1 каждого ВМ запущено не будет.

Варианты запуска задания пользователя через скрипт.

Команда *mpirun* осуществляет постановку в очередь и запуск MPI-программ. При желании можно выполнять программы без MPI, а также самостоятельно запускать программы на выделенных в пакетном режиме ресурсах, воспользовавшись командой *mbatch*. Синтаксис новой команды аналогичен описанному. По команде *mbach* задача ставится в очередь, но, вместо запуска самой программы на каждом ВУ, будет выполнен указанный пользователем командный файл (shell script) на первом выделенном ВУ.

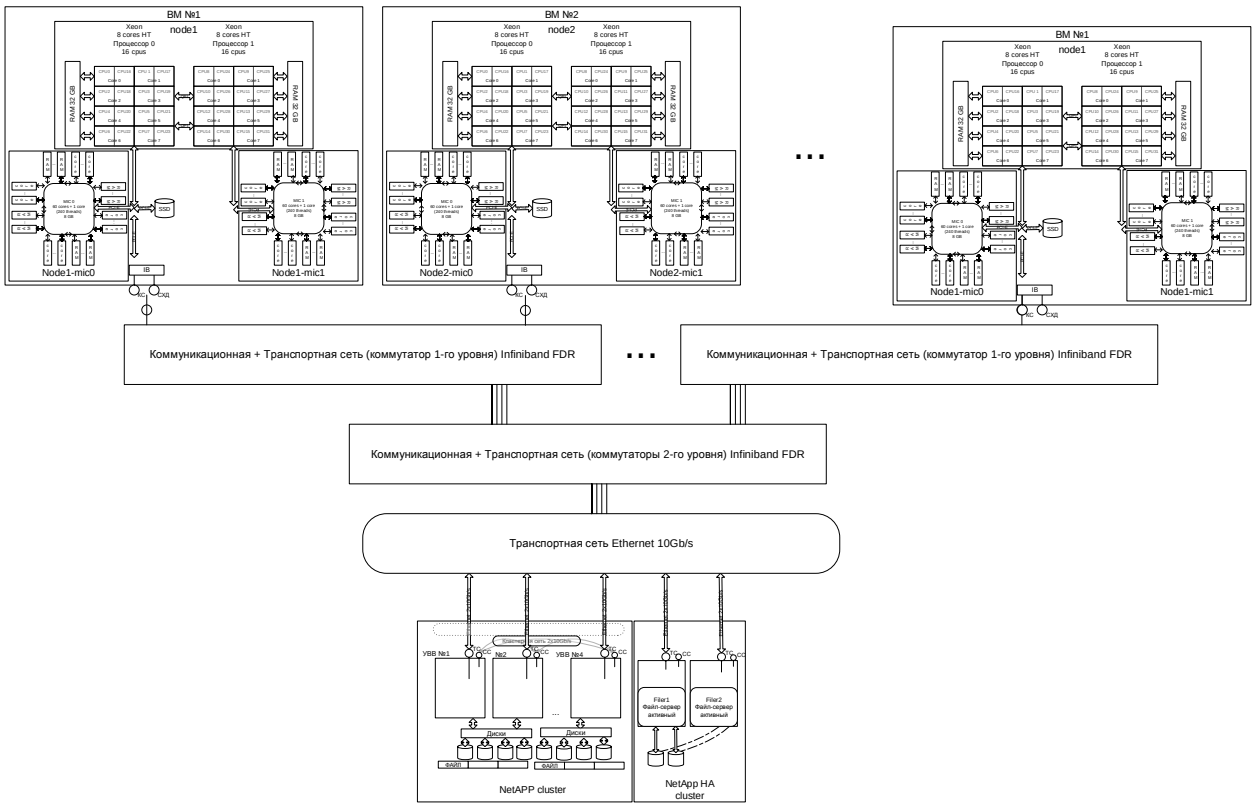
```
mbatch –np 48 runtask.sh
```

В окружение выполняющегося командного файла доступы следующие переменные:
nnodes – количество выделенных модулей;
nodeslist – список выделенных модулей;
nodes – сокращенный список модулей;
hosts_list – список выделенных модулей с учетом в формате *machinefile*;
Параметры из секций general и timerequest из паспорта задачи.

Пример:
[aladin@login bwltest]$ mbatch -np 17 ./test.sh
Trying head 10.65.0.1
Count of cpu is 48
Section [LocalDisks] not found, no LDM resource for your task
Section [LocalDisks] not found, no LDM resource for your task
Task "test.sh.1" queued successfully

Running task "test.sh.1" on following nodes:
node1 node2
Task "test.sh.1" started successfully, pid of manager is 62626
[aladin@login bwltest]$ cat test.sh.1/output
call_batch: run string - /usr/runmvs/bin/runtask
'/home1/aladin/JSCC/Works/TESTS/MVS10P/bwltest/test.sh.1/runmvs.bat'
'/home1/aladin/JSCC/Works/TESTS/MVS10P/bwltest/test.sh.1/.hosts'

'/common/runmvs/lock/tasks/aladin.test.sh.1/ret_code'
'/usr/runmvs/users/aladin/queue/test.sh.1'
Linux node1 2.6.32-220.el6.x86_64 #1 SMP Tue Dec 6 19:48:22 GMT 2011 x86_64 x86_64 x86_64 GNU/Linux
nnodes=2
nodeslist=node1
node2
nodes=node[1-2]
hosts_list=/home1/aladin/JSCC/Works/TESTS/MVS10P/bwltest/test.sh.1/.hosts
[aladin@login bwltest]$ cat test.sh.1/.hosts
node1:16
node2:1
[aladin@login bwltest]$


ВМ №1
node1
Xeon
8 cores HT
Процессор 0
16 cpus
Xeon
8 cores HT
Процессор 1
16 cpus
RAM 32 GB
RAM 32 GB
SSD
IB
Node1-mic0
Node1-mic1
ВМ №2
node2
Node2-mic0
Node2-mic1
Коммуникационная + Транспортная сеть (коммутатор 1-го уровня) Infiniband FDR
Коммуникационная + Транспортная сеть (коммутатор 1-го уровня) Infiniband FDR
Коммуникационная + Транспортная сеть (коммутаторы 2-го уровня) Infiniband FDR
Транспортная сеть Ethernet 10Gb/s
УВВ №1
№2
УВВ №4
Диски
RAID
Filer1
Файл-сервер
активный
Filer2
Файл-сервер
активный
NetAPP cluster
NetApp HA
cluster