volclava 用户手册

Product Name : volclava

Product Version : 1.0.0

Release Date : 2024.11.26

Contributor： @李明泽（limingze.jiayou@bytedance.com）
@舒光波 （shuguangbo@bytedance.com）
@亓楠 (qinan.cn@bytedance.com)
@周灵汛（zhoulingxun@bytedance.com）

目录

1. 简介

openlava 是 100%免费、开源、兼容 IBM LSF 的工作负载调度器，支持各种高性能计算和分析应用。

openlava 由 LSF 的早期版本开源而来，由于 openlava 的命令行和文件格式与大多数 LSF 功能相兼容,因此用户和管理员都将非常熟悉 openlava 的操作。

当前 volclava 1.0.0 基于 openlava2.0 版本进行了二次开发，修复了一些明显的缺陷,并增加了必要功能支持,可以覆盖 EDA 业务基本功能需求，推荐在小规模集群（小于百台节点），业务对调度性能不高的场景使用。

2. 安装及配置

可单独参见 volclava 安装及配置文档

3. 常用基本命令

基本命令	用法
bsub	提交任务到 volclava
bjobs	查看任务状态和基本信息
bkill	杀死未完成的任务
bqueues	查看队列状态和基本信息
bhosts	查看机器状态及基本信息
lshosts	查看机器静态资源状态
lsload	查看机器动态负载状态

其中 bsub 和 bjobs 是普通用户主要使用的指令。

bkill 用于 job 的终止控制，有时使用。

bhosts/lshosts/lsload/bqueues 则用户机器和队列的信息获取，管理员常用。

4. 使用案例

4.1 提交 job

bsub 命令最重要参数在于指定资源用量，包括 cpu 和 memory，以防止过多的资源 reserve 导致资源浪费，过少的资源 reserve 导致机器资源耗尽。

- **bsub -q [QueueName]** :指定队列，如果不指定，则任务会提交到默认队列（一般是 normal）

```
[root@master-test etc]# bsub -q normal sleep 10
Job <22> is submitted to queue <normal>.
```

- **bsub -Is**： 投递任务的时候用 shell 模式启动一个终端，并将任务投递到上面以交互式运行。启动需要交互式的工具，将标准输出打印到当前窗口，或者为了阻塞式运行任务（任务运行期间 bsub 不退出）

```
[root@master-test etc]# bsub -Is date
Job <23> is submitted to default queue <normal>.
<<Waiting for dispatch ...>>
<<Starting on cmp2-test>>
Thu Oct 31 14:20:38 CST 2024
```

- **bsub -o [FileName]：**保存任务的标准输出到指定的文件，这个模式和”-Is”相冲突，但是可以和”-e”叠加使用

```
[root@master-test etc]# bsub -o /tmp/stdout.txt sleep 100
Job <24> is submitted to default queue <normal>.
```

- **bsub -e [FileName]：**保存任务的标准错误到指定的文件，这个模式和”-Is”相冲突，但是可以和”-o”叠加使用

```
[root@master-test etc]# bsub -e /tmp/stderror.txt sleep 100
Job <25> is submitted to default queue <normal>.
```

- **bsub -n [Number] :** 指定为当前任务保留多少个 slots

```
[root@master-test etc]# bsub -n 8 sleep 10
Job <26> is submitted to default queue <normal>.
```

- **bsub -R [ResourceString] :** 指定为当前任务的资源需求

```
[root@master-test etc]# bsub -R "rusage[mem=1024]" sleep 100
Job <27> is submitted to default queue <normal>.
```

注：不支持 多个-R 请求资源，即不支持 bubub -R "xxx" -R "xxx"，这类情况需要合并书写

正确写法：

bsub -R "select[tmp>10240&&mem>=1024]" -Is date

bsub -R "select[tmp>10240] rusage[mem=1024]" -Is date

```
[root@master-test etc]# bsub -R "select[tmp>10240&&mem>=1024]" -Is date
Job <28> is submitted to default queue <normal>.
<<Waiting for dispatch ...>>
<<Starting on cmp2-test>>
Thu Oct 31 14:24:09 CST 2024
[root@master-test etc]# bsub -R "select[tmp>10240] rusage[mem=1024]" -Is date
Job <29> is submitted to default queue <normal>.
<<Waiting for dispatch ...>>
<<Starting on cmp2-test>>
Thu Oct 31 14:24:19 CST 2024
```

以下是一些典型的使用示例：

4.1.1 GUI 界面任务

- 当前任务为 verdi（图形界面），任务投递到队列 interactive，预计需要 4 个 cpu 核和 10G 内存，则任务投递方式为：

bsub -q interactive -n 4 -R "rusage[mem=10240]" verdi

参数说明：

-q interactive：提交到 interactive 队列，此处只能提交到自己有权限的队列

-n 4：指定任务需要 4 个 slots（即 4 线程），若未指定，默认分配一个 slots。

-R "rusage[mem=10240]"：为本任务预留 10G 内存

注： 请提前 module load 对应的 EDA 工具

4.1.2 交互式任务

- 当前任务为 pt_shell，需要和 pt_shell 工具进行命令行交互，任务投递到 projectA 队列,预计需要 4 个 cpu 核和 100G 内存，则任务投递方式为：

bsub -q projectA -n 4 -Is -R "rusage[mem=102400]" pt_shell

参数说明：

-q projectA：提交到 projectA 队列，此处只能提交到自己有权限的队列

-n 4：指定任务需要 4 个 slots（即 4 线程），若未指定，默认分配一个 slots。

-Is：启动的任务是交互式任务，远程机器上的命令输出会映射到当前机器上。

-R "rusage[mem=102400]" ：为本任务预留 100G 内存

注： 请提前 module load 对应的 EDA 工具

4.1.3 常规 EDA 任务

- 当前任务为 liberate，任务投递到 projectA 队列，需要保存标准输出和标准错误，预计 2 个 cpu 核足够， 要求投递的机器剩余内存大于 100G，剩余 swap 大于 100G， 剩余 tmp 空间大于 30G，则任务投递方式为：

bsub -q projectA -n 2 -o stdout.log -e stderror.log -R "select[mem>=102400 && swap>=102400 && tmp>=30720]" liberate

参数说明：

-q projectA : 提交到 projectA 队列，此处只能提交到自己有权限的队列

-n 2：指定任务需要 2 个 slots（即 2 线程），若未指定，默认分配一个 slots。

-o :保存任务的标准输出到指定的文件，这个模式和”-Is”相冲突

-e :保存任务的标准错误到指定的文件，这个模式和”-Is”相冲突

-R “select[mem>=102400 && swap>=102400 && tmp>=30720]” : 选择剩余内存大于 100G，剩余 swap 大于 100G， 剩余 tmp 空间大于 30G 的机器

注： 请提前 module load 对应的 EDA 工具

说明：

select 能够选择当前满足条件的机器，但是不能预占机器上的资源，请使用 rusage 命令预占资源。

4.2 查询 job 信息

4.2.1 bjobs

查看当前用户所有的未完成 job。

```
[root@master-test etc]# bjobs
JOBID   USER    STAT  QUEUE      FROM_HOST   EXEC_HOST   JOB_NAME   SUBMIT_TIME
30      root    RUN   normal     master-test cmp2-test   sleep 1000 Oct 31 14:26
31      root    RUN   normal     master-test cmp1-test   sleep 1000 Oct 31 14:27
32      root    RUN   normal     master-test cmp1-test   sleep 1000 Oct 31 14:27
33      root    PEND  normal     master-test             sleep 1000 Oct 31 14:29
```

4.2.2 bjobs -a

查看当前用户在一段时间内所有的 job，包括已完成和未完成的 job

```
[root@master-test etc]# bjobs -a
JOBID   USER    STAT  QUEUE      FROM_HOST   EXEC_HOST   JOB_NAME   SUBMIT_TIME
30      root    RUN   normal     master-test cmp2-test   sleep 1000 Oct 31 14:26
31      root    RUN   normal     master-test cmp1-test   sleep 1000 Oct 31 14:27
32      root    RUN   normal     master-test cmp1-test   sleep 1000 Oct 31 14:27
33      root    PEND  normal     master-test             sleep 1000 Oct 31 14:29
20      root    DONE  normal     master-test cmp2-test   date       Oct 31 14:19
21      root    EXIT  normal     master-test cmp2-test   /bin/bash  Oct 31 14:19
```

4.2.3 bjobs -UF [JobId] :

```
[root@master-test ~]# bjobs -UF 36

Job <36>, User <root>, Project <default>, Status <RUN>, Queue <normal>, Command <sleep 100>

Thu Oct 31 14:30:45: Submitted from host <master-test>, CWD <$HOME>;
Thu Oct 31 14:30:51: Started 1 Task(s) on Host(s) <cmp2-test>, Allocated 1 Slot(s) on Host(s) <cmp2-test>, Ex
ecution Home </root>, Execution CWD </root>;
Thu Oct 31 14:31:41: Resource usage collected. MEM: 3 Mbytes SWAP: 236 Mbytes; PGID: 38871; PIDs: 38871 38874
 38876;

 MEMORY USAGE:
 MAX MEM: 3 Mbytes;  AVG MEM: 3 Mbytes

 SCHEDULING PARAMETERS:
           r15s   r1m  r15m   ut      pg    io   ls    it    tmp    swp    mem
 loadSched   -     -     -     -       -     -    -     -     -      -      -
 loadStop    -     -     -     -       -     -    -     -     -      -      -

 RESOURCE REQUIREMENT DETAILS:
 Combined:
 Effective:
```

4.2.4 bjobs -l [JobId]:

```
[root@master-test ~]# bjobs -l 36

Job <36>, User <root>, Project <default>, Status <RUN>, Queue <normal>, Command <sleep 100>
Thu Oct 31 14:30:45: Submitted from host <master-test>, CWD <$HOME>;
Thu Oct 31 14:30:51: Started on <cmp2-test>, Execution Home </root>, Execution CWD </root>;
Thu Oct 31 14:31:41: Resource usage collected.
                     MEM: 3 Mbytes;  SWAP: 236 Mbytes
                     PGID: 38871;  PIDs: 38871 38874 38876

 MEMORY USAGE:
 MAX MEM: 3 Mbytes;  AVG MEM: 3 Mbytes

 SCHEDULING PARAMETERS:
           r15s   r1m  r15m   ut      pg    io   ls    it    tmp    swp    mem
 loadSched   -     -     -     -       -     -    -     -     -      -      -
 loadStop    -     -     -     -       -     -    -     -     -      -      -

 RESOURCE REQUIREMENT DETAILS:
 Combined:
 Effective:
```

4.2.5 bjobs -o

来显示指定的字段名字,当前 bjobs -o 支持的参数:

"JOBID", "USER", "STAT", "QUEUE", "FROM_HOST", "EXEC_HOST", "JOB_NAME", "SUBMIT_TIME",

"PROJ_NAME", "CPU_USED", "MEM", "SWAP", "PIDS", "START_TIME", "FINISH_TIME"

```
[root@master-test etc]# bjobs -o 'jobid stat QUEUE EXEC_HOST USER SUBMIT_TIME FROM_HOST JOB_NAME PROJ_NAME CP
U_USED MEM SWAP PIDS START_TIME FINISH_TIME'
JOBID STAT QUEUE EXEC_HOST USER SUBMIT_TIME FROM_HOST JOB_NAME PROJ_NAME CPU_USED MEM SWAP PIDS START_TIME FI
NISH_TIME
30 RUN normal cmp2-test root Oct 31 14:26 master-test sleep 1000 default 000:00:-2.00 0 0 - 10/31-14:27:01 -
31 RUN normal cmp1-test root Oct 31 14:27 master-test sleep 1000 default 000:00:00.00 1996 31404 107488 10/31
-14:27:11 -
32 RUN normal cmp1-test root Oct 31 14:27 master-test sleep 1000 default 000:00:00.00 2104 31624 107490 10/31
-14:27:11 -
```

4.2.6 bjobs -o -json

结合-o 使用，可以将输出内容转换成 json 格式；不可以脱离-o 单独使用

```
[root@master-test etc]# bjobs -o 'jobid stat QUEUE EXEC_HOST USER SUBMIT_TIME FROM_HOST JOB_NAME PROJ_NAME CP
U_USED MEM SWAP PIDS START_TIME FINISH_TIME' -json
{
        "COMMAND":      "bjobs",
        "JOBS": 3,
        "RECORDS":      [{
                        "JOBID":        "30",
                        "STAT": "RUN",
                        "QUEUE":        "normal",
                        "EXEC_HOST":    "cmp2-test",
                        "USER": "root",
                        "SUBMIT_TIME":  "Oct 31 14:26",
                        "FROM_HOST":    "master-test",
                        "JOB_NAME":     "sleep 1000",
                        "PROJ_NAME":    "default",
                        "CPU_USED":     "000:00:00.00",
                        "MEM":  "3632",
                        "SWAP": "241788",
                        "PIDS": "38846,38848,38849",
                        "START_TIME":   "10/31-14:27:01",
                        "FINISH_TIME":  "-"
```

4.3 终止 job

4.3.1 bkill 0

杀死当前用户所有的 job

```
[root@master-test ~]# bkill 0
Job <34> is being terminated
Job <35> is being terminated
Job <36> is being terminated
```

注：volclava 集群管理员账号使用该命令时会杀死集群内所有的 job

4.3.2 bkill [JobId]

强制杀死指定的 job

```
[root@master-test ~]# bkill 38
Job <38> is being terminated
```

4.4 查询队列

4.4.1 bqueues

常用于查询队列 running 以及 pending 情况

```
[root@master-test etc]# bqueues
QUEUE_NAME      PRIO STATUS          MAX JL/U JL/P JL/H NJOBS  PEND   RUN  SUSP
short           35   Open:Active       -    -    -    -     2     1     1     0
normal          30   Open:Active       -    -    -    -     1     1     0     0
```

4.4.2 bqueues -u [User]

查看指定用户可用队列,即有权限提交的队列

```
[root@master-test etc]# bqueues -u volclava
QUEUE_NAME      PRIO STATUS          MAX JL/U JL/P JL/H NJOBS  PEND   RUN  SUSP
short           35   Open:Active       -    -    -    -     0     0     0     0
normal          30   Open:Active       -    -    -    -     1     0     1     0
```

5. 常见问题及解决方案

5.1 如何查看 job PEND 的原因

5.1.1 bjobs -p

bjobs -p 命令可过滤出当前用户所有 pending 的 job,且显示 pending 原因

如下图 pend 的原因为：New job is waiting for scheduling（job 正在等待调度）

```
[root@master-test etc]# bjobs -p
JOBID   USER    STAT  QUEUE      FROM_HOST      JOB_NAME          SUBMIT_TIME
42      root    PEND  normal     master-test    sleep 100         Oct 31 14:36
New job is waiting for scheduling: 1 host;
```

5.1.2 bjobs -l [JobId]

该命令可以查看指定 job pending 的原因

```
[root@master-test etc]# bjobs -l 44

Job <44>, User <root>, Project <default>, Status <PEND>, Queue <normal>, Command <sleep 100>
Thu Oct 31 14:37:07: Submitted from host <master-test>, CWD </ic/software/tools/volclava/etc>;
 PENDING REASONS:
 New job is waiting for scheduling: 1 host;

 SCHEDULING PARAMETERS:
           r15s   r1m  r15m   ut      pg    io   ls    it    tmp    swp    mem
 loadSched   -     -     -     -       -     -    -     -     -      -      -
 loadStop    -     -     -     -       -     -    -     -     -      -      -

 RESOURCE REQUIREMENT DETAILS:
 Combined:
 Effective:
```

5.2 常见 job PEND 的原因及解决方案

5.2.1 job 在等待投递

- New job is waiting for scheduling

任务在投递出去以后，运行起来之前，需要等待 volclava 调度分配，会有一小段等待时间。

5.2.2 queue 限制了每个用户最大可用 slots 数目，用户任务数目超限

- User has reached the per-user job slot limit of the queue

例如 test 队列设置每个用户最多使用 30 个 slots，按每个 job 占用 1 个 slots 算，用户投递超过 30 个 jobs 后，新的 job 就会变成 PEND 状态。

此时可以等待旧的 job 完成，或者把任务投递到同项目的其它 queue 中

5.2.3 cpu 不能满足需求

- Not enough job slot(s);

申请了多个 slots，但是 server 上可用 slots 都不能满足需求。

这类情况请评估资源申请是否有误，以及是否必须。若不是必须可以适当调小 **bsub -n** 指定的的数值，若是必要的资源可联系 volclava 管理员配置专用资源

5.2.4 mem 不能满足需求

- Job requirements for reserving resource (mem) not satisfied

申请了较大的 memory，但是 server 上可用的 memory 都不能满足需求。

这类情况请评估资源申请是否有误，以及是否必须。若不是必须可以适当调小 **bsub -R "rusage [mem=xxx]"** 指定的数值，若是必要的资源可联系 volclava 管理员配置专用资源

5.3 Job 异常退出的原因

bjobs -l [JobId] 可以查询指定 job 的详细信息，其中包含 job 的退出码

```
[root@master-test ~]# bjobs -l 59

Job <59>, User <root>, Project <default>, Status <EXIT>, Queue <normal>, Command <test.sh>
Thu Oct 31 14:51:07: Submitted from host <master-test>, CWD <$HOME>;
Thu Oct 31 14:51:12: Started on <cmp2-test>, Execution Home </root>, Execution CWD </root>;
Thu Oct 31 14:51:12: Exited with exit code 127. The CPU time used is 0.0 seconds.

 SCHEDULING PARAMETERS:
           r15s   r1m  r15m   ut      pg    io   ls    it    tmp    swp    mem
 loadSched   -     -     -     -       -     -    -     -     -      -      -
 loadStop    -     -     -     -       -     -    -     -     -      -      -

 RESOURCE REQUIREMENT DETAILS:
 Combined:
 Effective:
```

如果 Exit code 小于 128，说明是 job 的 command 本身执行出错，是工具问题，而非系统问题，更详细的原因需要去看工具 log。

如果 Exit code 大于 128，说明 job 本身遇到异常，比如被 volclava、系统或者用户 kill，此时可以找 volclava 管理员一起来 debug 原因。

6. 功能列表

6.1 已支持功能

功能分类	命令示例	说明
作业提交	bsub -q	指定队列
	bsub -P	指定项目
	bsub -m	指定主机
	bsub -J	指定作业名称，指定作业数组的索引
	bsub -Ip / -I / -Is	交互任务
	bsub -n	指定 slots 数量
	bsub -E	前处理命令

	bsub -Ep	后处理命令
	bsub -e / -eo	标准错误重定向到指定文件
	bsub -o / -oo	标准输出重定向到指定文件
	bsub -W	限制运行时间 （run_limit）
	bsub -M	限制内存
	bsub -C	限制 core file 大小
	bsub -c	限制任务 CPU 时间（cpu_limit ）
	bsub -D	限制数据大小（data_limit）
	bsub -F	限制文件数限制 （file_limit）
	bsub -S	限制栈（stack_limit）
	bsub -v	限制交换分区（swap_limit）
	bsub -R "rusage[mem=10000]"	内存保留
	bsub -R "select[swp > 1000]"	选择符合资源需求的机器
	bsub -pack	批量提交任务 -pack
作业查询	**bjobs**	显示当前用户 pending、running 和 suspended 作业的信息
	bjobs -a	显示所有状态下的作业信息，包括最近完成的作业
	bjobs -l	长格式。以多行格式显示每个作业的详细信息。
	bjobs -w	宽格式。显示作业信息而不截断字

		段
	bjobs -u	显示指定用户或用户组提交的作业
	bjobs -r	显示正在运行的作业
	bjobs -p	显示 pending 的作业，以及导致 pending 原因
	bjobs -s	显示 suspended 的作业，以及导致 suspending 原因
	bjobs -P	显示属于指定项目的作业
	bjobs -d	显示有关最近完成的作业的信息
	bjobs -m	显示分派到指定主机的作业
	bjobs -q	显示指定队列中的作业
	bjobs -J	显示具有指定作业名称的作业或作业数组的信息
	bjobs [job_id[index]]	指定 bjobs 显示的作业或作业数组
	bjobs -UF	显示未格式化的作业详细信息
	bjobs -o [*format]*	设置自定义输出格式
	bjobs -o [*format]* -json	以 JSON 格式显示自定义输出
	bhist	显示作业的历史信息
作业控制	bkill [job_id]	终止指定作业
	bkill -0	杀死当前用户的所有作业
	bstop	暂停作业

	bresume	恢复作业
查询队列信息	bqueues	
查询主机组信息	**bmgroup**	
查询用户组信息	**bugroup**	
显示主机状态及 slots 信息	**bhosts**	
显示主机的负载信息	lsload	
显示主机及其静态资源信息	**lshosts**	

附录：

附一、变更历史

日期	版本	变更描述
2024.11.26	Volclava 1.0	Volclava 1.0 派生自 Openlava 2.0. 添加了以下功能和问题修复： • 多项功能支持：bjobs -UF；bjobs -o/-json；bsub -pack；bsub -Ep；等等。 • 多处错误修复：设置 “!” 时 MXJ 与 maxCpus 不相等；lshosts -l 出现段错误； • sbatchd 因超过 1000 个任务而受阻；在 RPM 安装中前缀无法用于自定义目录； • 修复主机空闲时却达到作业槽位限制的问题。 在相关文件中将新项目名称定义为 volclava
