

IsfMonitor 用户手册

Product Name : IsfMonitor

Product Version : V1.7

Release Date : 2025.03.26

Contact : [@李艳青](#) (liyanqing1987@163.com)

目录

一、简介	4
1.1 功能简介	4
1.2 快速了解	5
二、环境依赖	10
2.1 操作系统依赖	10
2.2 PYTHON 版本依赖	10
2.3 集群管理工具	10
三、工具安装及配置	11
3.1 工具下载	11
3.2 工具安装	12
3.3 工具配置	14
四、工具使用	15
4.1 数据采集 BSAMPLE	15
4.1.1 帮助信息	15
4.1.2 手工采样	16
4.1.3 自动采样 (定时)	16
4.1.4 数据库	17
4.2 数据展示 BMONITOR	19
4.2.1 工具载入	19
4.2.2 帮助信息	21
4.2.3 菜单栏	22
4.2.4 JOB 页	23
4.2.5 JOBS 页	24
4.2.6 HOSTS 页	25
4.2.7 LOAD 页	26
4.2.8 USERS 页	26
4.2.9 QUEUES 页	27
4.2.10 UTILIZATION 页	28
4.2.11 LICENSE 页	29
4.2.12 其它功能介绍	30
五、辅助工具	34
5.1 AKILL	34
5.2 PATCH	35
5.3 SEEDB	36
5.4 MEMPREDICTION	39
六、常见问题及解决	40
6.1 图形显示问题	40
6.2 JOBS 页信息缺失	40

6.3 LICENSE 页信息缺失	40
6.4 HOSTS 页和 LOAD 页中的 MEM 值为什么不一致.....	41
七、技术支持	42
附录	43
附 1. 变更历史.....	43

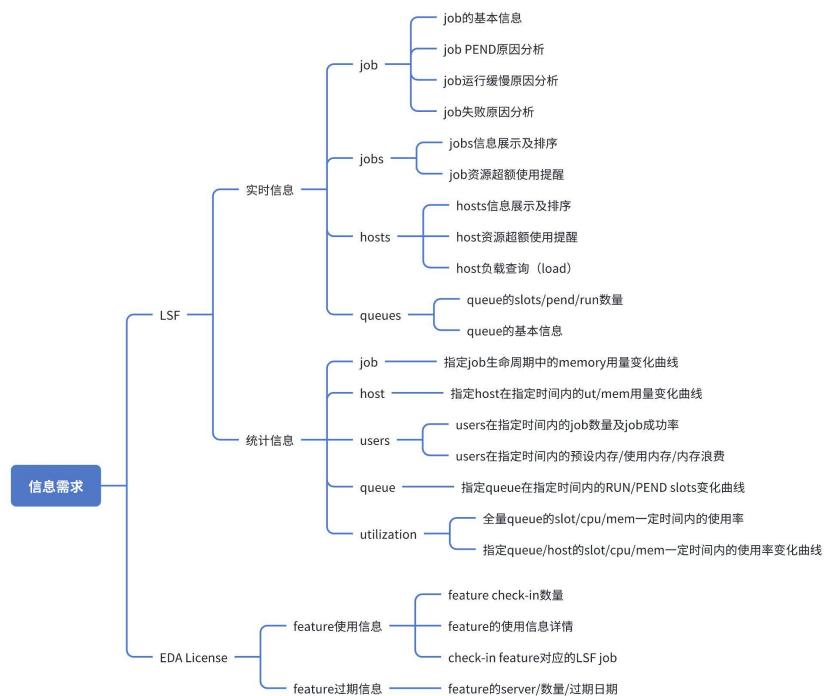
一、简介

1.1 功能简介

LSF 是 IBM 旗下的一款分布式集群管理系统软件，负责计算资源的管理和批处理作业的调度。它具有良好的可伸缩性和高可用性，支持几乎所有的主流操作系统，通常是高性能计算环境中重要的基础软件。

lsfMonitor 是一款适用于 LSF/volclava/openlava4 数据收集、分析及展示的开源工具，亦可用于 EDA license 实时信息检索，可以满足集成电路行业用户对于 LSF/license 的绝大部分信息需求。

在集成电路设计 Linux 环境中，关于 LSF 集群和 EDA license 使用状况，用户常常会面临如下信息需求：



lsfMonitor 就是为如上信息需求提供一站式解决方案。

1.2 快速了解

lsmMonitor 安装目录下的 monitor/bin 目录内，包含两个主脚本 bsample 和 bmonitor，bsample 用于信息采集，bmonitor 用于信息展示。

lsmMonitor 的正确使用姿势是：安装 -> 配置 -> bsample 采样 -> bmonitor 展示



bmonitor 支持“亮色”和“暗黑”两种界面模式。快速上手章节我们采用暗色系展示，工具使用章节则采用默认的亮色系展示。

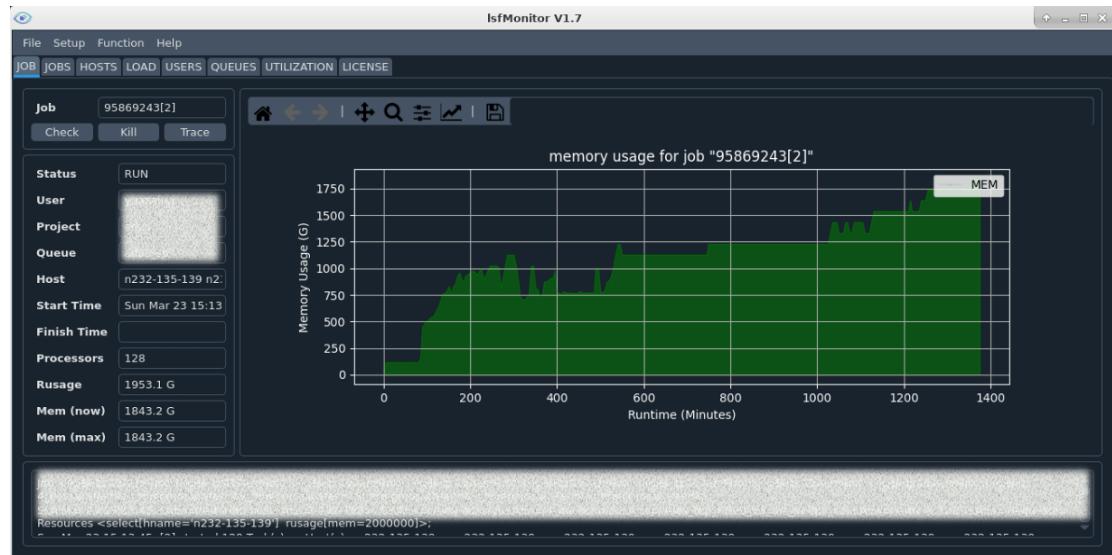
The table compares two screenshots of the lsmMonitor interface. Both screenshots show the 'JOBS' page with a list of tasks. The left screenshot is in light mode, and the right screenshot is in dark mode. The columns include Job ID, User, Status, Queue, Host, Started, Project, Stat, Usage (G), Mem (G), and Command.

Job	User	Status	Queue	Host	Started	Project	Stat	Usage (G)	Mem (G)	Command
1. 29841720	RUN	Interactive	i249-073-094	2020-12-04 12:36:47	1.5	0	vicode			
2. 31702687	RUN	Interactive	i249-073-094	2020-12-09 14:08:07	2	0	vicode			
3. 34055559	RUN	Interactive	i249-073-094	2020-12-13 19:18:18	2	0	vicode			
4. 25567517	RUN	Interactive	i249-073-094	2020-12-16 20:08:31	2	0.7	vicode +extensions-dir /icloudware...			
5. 40361452	RUN	Interactive	i249-073-094	2020-12-17 00:20:03	2	0	vicode			
6. 24243667	RUN	Interactive	i249-073-094	2020-12-20 22:22:18	2	0.5	vicode			
7. 64913954	RUN	Interactive	i249-073-094	2020-01-13 10:37:51	2	0	vicode-V			
8. 40372402	RUN	Interactive	i249-073-094	2020-01-23 10:50:49	2	0.8	vicode			
9. 77642490	RUN	Interactive	i249-073-094	2020-02-10 14:04:28	2	0	vicode-J			
10. 40712383	RUN	Interactive	i249-073-094	2020-02-16 14:03:45	2	0.1	vicode			
11. 68141139	RUN	Interactive	i249-073-094	2020-02-27 18:06:45	2	0.3	vicode			
12. 68621601	RUN	Interactive	i249-073-094	2020-03-01 10:40:26	2	0.4	vicode			
13. 60641079	RUN	Interactive	i249-073-094	2020-03-10 17:35:19	2	0.6	vicode			
14. 90995981	RUN	Interactive	i249-073-094	2020-03-11 10:21:51	2	0.7	vicode			
15. 91409028	RUN	Interactive	i249-073-094	2020-03-12 16:09:05	2	0.7	vicode			

lsmMonitor 共包含 JOB/JOBs/HOSTS/LOAD/USERS/QUEUES/UTILIZATION/LICENSE 八个页面，打开初始默认为 JOBS 页，下面来依次展示不同页面。

JOB 页：

展示单个 job 的信息。在 Job 文本框输入 jobid，点击 Check 按钮，左侧会显示 job 的关键信息，（如已采样）右侧会显示 job 生命周期内的内存用量曲线，下侧会显示 job 的详细信息。



JOBS 页：

展示所有 RUN 状态的 job 及其关键信息，也可以按照 Status/Queue/Host/User 来精准筛选 job。

Status: RUN

Queue: ALL

Host: ALL

User: [redacted]

Check

	Job	User	Status	Queue	Host	Started	Project	Slot	Rusage (G)	Mem (G)	Command
1	29841720	[redacted]	RUN	interactive	n249-073-094	2024-12-04 10:36:47	[redacted]	1	1.5	0	vscode
2	31702687	[redacted]	RUN	interactive	n249-073-094	2024-12-09 14:08:07	[redacted]	1	0.9	0	vscode
3	34055939	[redacted]	RUN	interactive	n249-073-094	2024-12-13 19:18:54	[redacted]	1	2	0	vscode
4	35367517	[redacted]	RUN	interactive	n249-073-094	2024-12-16 20:08:31	[redacted]	1	2	0.7	vscode --extensions-dir /ic/software/...
5	45366428	[redacted]	RUN	interactive	n249-073-094	2024-12-30 19:47:03	[redacted]	1	2	0	vscode
6	45436687	[redacted]	RUN	interactive	n249-073-094	2024-12-30 21:22:16	[redacted]	1	2	0.5	vscode
7	54913934	[redacted]	RUN	interactive	n249-073-094	2025-01-13 16:17:51	[redacted]	1	2	0	vscode -V
8	60372402	[redacted]	RUN	interactive	n249-073-094	2025-01-23 14:50:49	[redacted]	1	2	0.8	vscode
9	77642490	[redacted]	RUN	interactive	n249-073-094	2025-02-10 14:04:28	[redacted]	1	2	0	vscode ./
10	80712383	[redacted]	RUN	interactive	n249-073-094	2025-02-16 14:13:45	[redacted]	1	2	0.1	vscode
11	86141139	[redacted]	RUN	interactive	n249-073-094	2025-02-27 16:06:45	[redacted]	1	2	0.3	vscode .
12	88621601	[redacted]	RUN	interactive	n249-073-094	2025-03-05 15:45:26	[redacted]	1	2	0.4	vscode
13	90661079	[redacted]	RUN	interactive	n249-073-094	2025-03-10 17:15:19	[redacted]	1	2	3.6	vscode
14	90959561	[redacted]	RUN	interactive	n249-073-094	2025-03-11 10:21:51	[redacted]	1	2	2.8	vscode
15	91405028	[redacted]	RUN	interactive	n249-073-094	2025-03-12 16:09:05	[redacted]	1	2	0.7	vscode

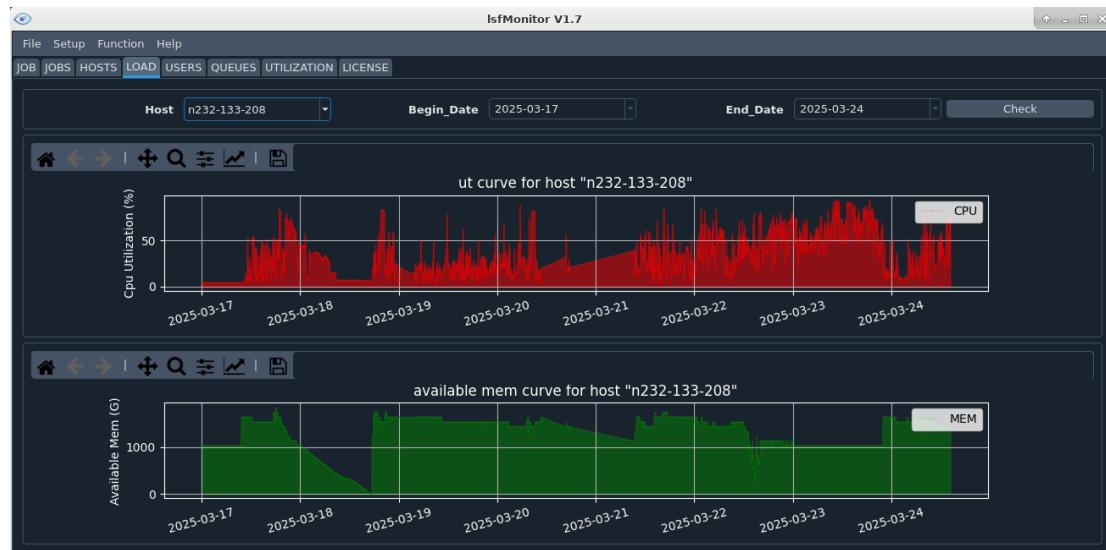
HOSTS 页：

展示所有 LSF 服务器的静态/动态信息，也可以按照 Status/Queue/MAX/MaxMem/Host 来精准筛选 host。

Host	Status	Queue	MAX	Njobs	Ut (%)	MaxMem (G)	aMem (G)	saMem (G)	MaxSwp (G)	Swp (G)	Tmp (G)
1 ic-lsfmaster1	ok		8	0	25	94	80	80	7	7	27
2 ic-lsfmaster2	ok		8	0	38	62	39	39	7	7	14
3 n232-129-012	ok		64	50	3	1945	1638	798	127	127	1504
4 n232-129-013	ok		96	48	2	1945	1228	1228	127	127	1071
5 n232-129-017	ok		64	50	8	1945	1740	1126	127	127	1508
6 n232-129-018	ok		64	59	2	1945	1740	1638	127	123	1507
7 n232-129-019	ok		64	48	5	1945	1638	1433	127	127	1505
8 n232-129-020	ok		64	54	13	1945	1945	1740	127	126	1507
9 n232-129-021	ok		64	61	3	1945	1638	361	127	127	1480
10 n232-129-022	ok		64	54	0	1945	1740	1536	127	127	1501
11 n232-129-023	ok		64	46	4	1945	1536	675	127	127	1491
12 n232-129-024	ok		64	52	78	1945	1536	1024	127	104	1503
13 n232-129-025	ok		64	51	32	1945	1433	1228	127	127	1504
14 n232-129-026	ok		64	51	9	1945	713	268	127	118	1071
15 n232-129-027	ok		128	4	1	1945	1536	1433	127	127	1509

LOAD 页：

展示指定服务器在指定时间段的 ut/mem 负载变化。



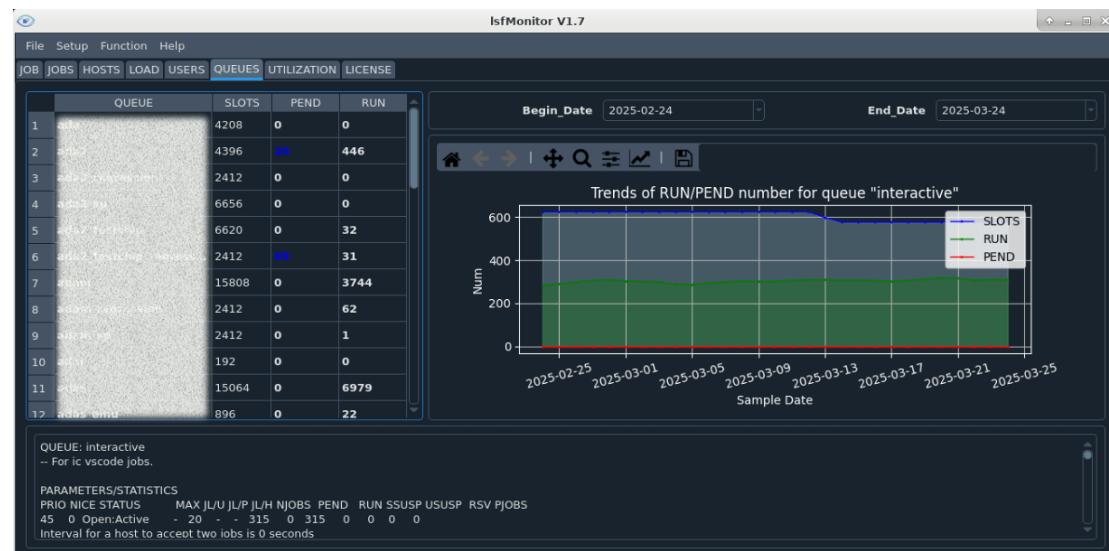
USER 页：

展示指定时间段内用户任务的基本汇总信息，包括 job 数目、成功率、预设内存、最大内存用量、内存浪费，也可以按照 Status/Queue/Project/User 来精准筛选用户数据。

User	Job_Num	Pass_Rate (%)	Total_Rusage_Mem (G)	Avg_Rusage_Mem (G)	Total_Max_Mem (G)	Avg_Max_Mem (G)	Total_Mem_Waste (G)	Avg_Mem_Waste (G)
1	710320	98.6	3131868.1	4.4	2325033.1	3.3	806835.0	1.1
2	2402	100.0	2472.7	1.0	1577.0	0.7	895.7	0.4
3	4669	100.0	23827.2	5.1	4059.8	0.9	19767.4	4.2
4	388016	100.0	747501.4	1.9	249686.7	0.6	497814.7	1.3
5	14	78.6	644.5	46.0	1040.1	74.3	-395.6	-28.3
6	20	80.0	1000.0	50.0	716.2	35.8	283.8	14.2
7	2334	99.8	23301.0	10.0	1529.8	0.7	21771.2	9.3
8	3	100.0	205.1	68.4	139.6	46.5	65.5	21.8
9	1660	100.0	628.2	0.4	870.9	0.5	-242.7	-0.1
10	8	75.0	3523.4	440.4	1414.5	176.8	2108.9	263.6
11	12	91.7	20.8	1.7	13.5	1.1	7.3	0.6
12	15	66.7	18.5	1.2	4.6	0.3	13.9	0.9
13	21	71.4	615.2	29.3	32.7	1.6	582.5	27.7
14	48	41.7	3243.0	67.6	473.0	9.9	2770.0	57.7

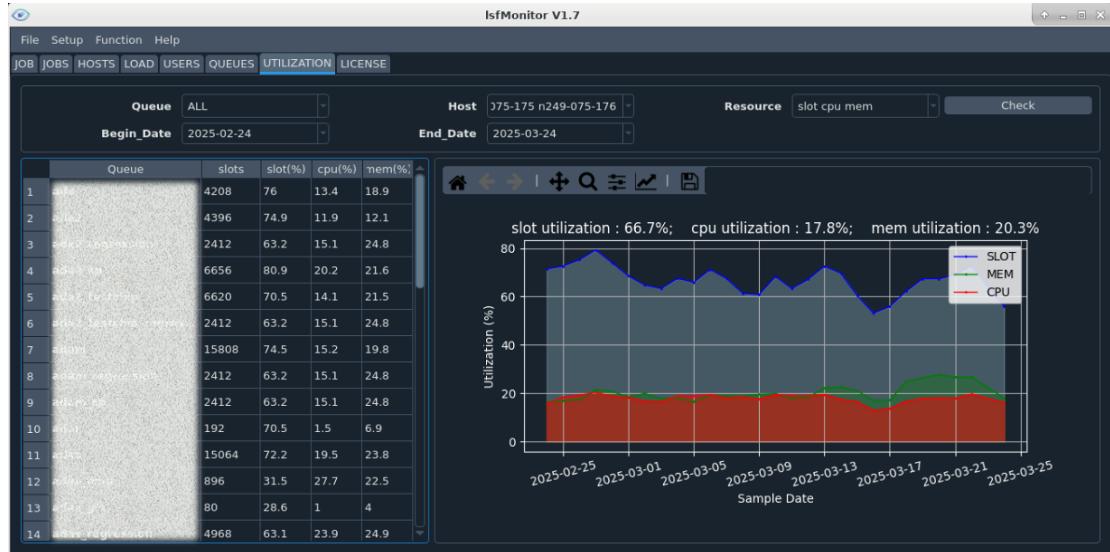
QUEUES 页：

展示所有 queue 瞬时的 SLOTS/PEND/RUN job 数目，如果点击指定的 queue，则右侧展示指定时间段内的 SLOTS/PEND/RUN job 数据变化趋势，下侧展示 queue 的详细信息。



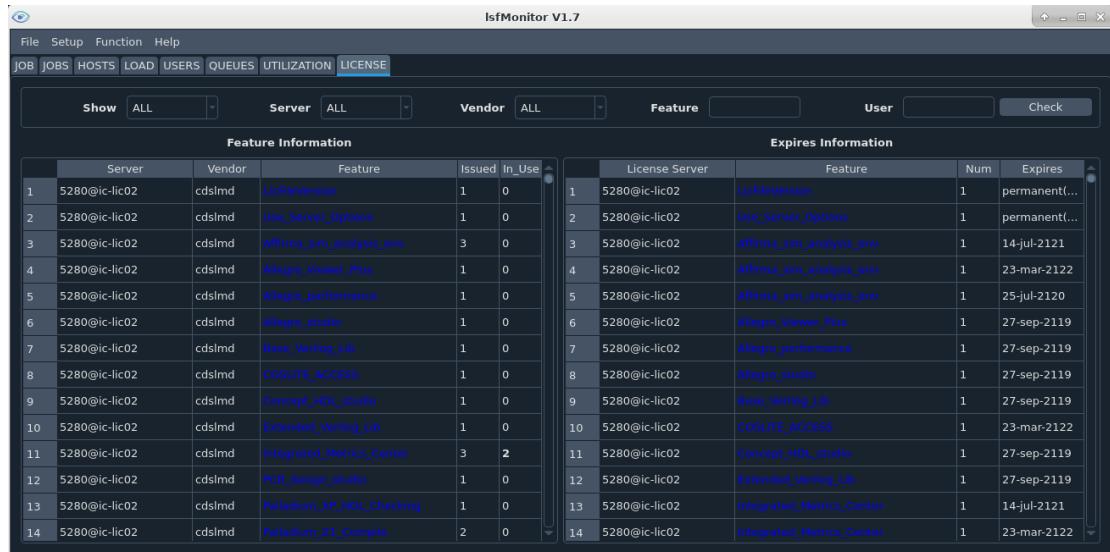
UTILIZATION 页：

展示指定时间段内所有 queue 的 slots/slots 利用率/cpu 利用率/mem 利用率信息，可以按照 Queue/Host 来指定逻辑单元，则会在右侧显示指定服务器单元在指定时间段内的利用率信息。



LICENSE 页：

会根据当前 Terminal 中用户的 **LM_LICENSE_FILE** 环境变量的设置，展示相关 license server 上所有 feature 的总量/用量信息。



二、环境依赖

2.1 操作系统依赖

lsmMonitor 的开发和测试操作系统为 **CentOS Linux release 7.9.2009 (Core)**，这也是 IC 设计常用的操作系统版本之一。

centos7/centos8/rocky8/redhat7/redhat8 理论上都可以运行，主要的潜在风险在于系统库版本差异可能会影响部分组件的正常运行。

2.2 python 版本依赖

lsmMonitor 基于 python 开发，其开发和测试的 python 版本为 **python3.8.8**。

不同版本的 python 可能会有 python 库版本问题，按照系统要求安装对应版本的 python 库即可解决。

2.3 集群管理工具

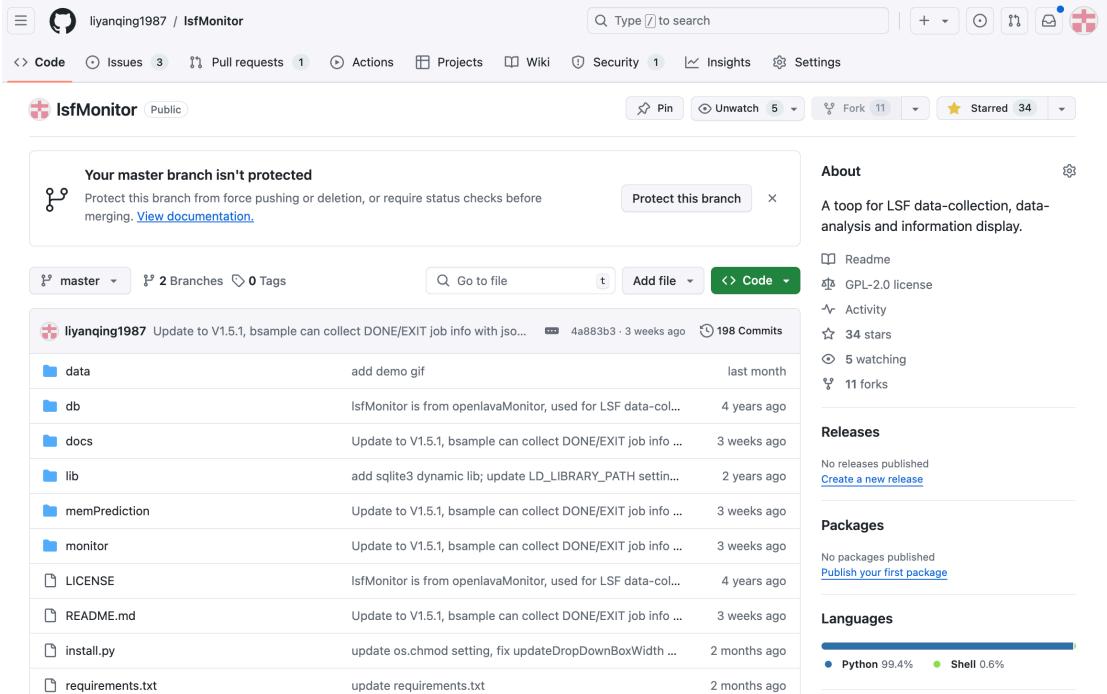
lsmMonitor 依赖 LSF/volclava/Oenlava 集群管理系统，暂不支持其它集群管理系统。

LSF 9.1.3 及以上的版本良好支持，volclava 支持良好，Openlava 几个版本间输出信息格式有一定差异，仅支持主流版本。

三、工具安装及配置

3.1 工具下载

lsfMonitor 的 github 路径位于 <https://github.com/liyanqing1987/lsfMonitor>

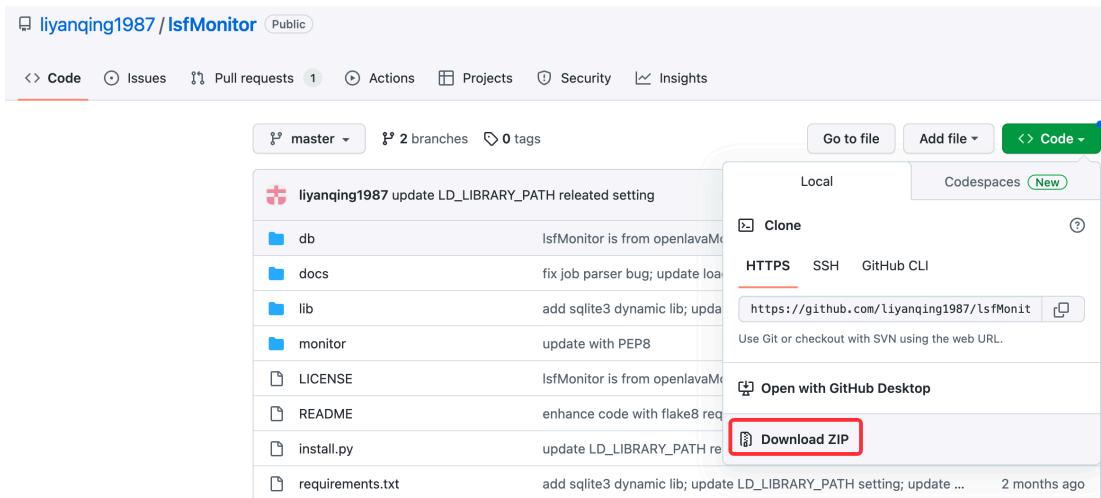


The screenshot shows the GitHub repository page for 'liyanqing1987 / lsfMonitor'. The repository is public and has 198 commits. The 'About' section describes it as a tool for LSF data-collection, data-analysis and information display. It includes links to the README, GPL-2.0 license, activity, and statistics like 34 stars, 5 watchers, and 11 forks. The 'Code' tab is selected, showing the master branch with 2 branches and 0 tags. The commit history lists several updates from the 'data', 'db', 'docs', 'lib', 'memPrediction', 'monitor', 'LICENSE', 'README.md', 'install.py', and 'requirements.txt' files.

可以采用“`git clone https://github.com/liyanqing1987/lsfMonitor.git`”的方式拉取源代码。

```
[liyanqing.1987@LQ2KW66YXF ~/test]#git clone  
git@github.com:liyanqing1987/lsfMonitor.git  
Cloning into 'lsfMonitor'...  
remote: Enumerating objects: 1320, done.  
remote: Counting objects: 100% (477/477), done.  
remote: Compressing objects: 100% (248/248), done.  
remote: Total 1320 (delta 267), reused 434 (delta 226), pack-  
reused 843 (from 1)  
Receiving objects: 100% (1320/1320), 44.45 MiB | 975.00 KiB/s,  
done.  
Resolving deltas: 100% (816/816), done.
```

也可以在 lsfMonitor 的 github 页面上，Code -> Download ZIP 的方式拉取代码包。



3.2 工具安装

工具安装之前，首先参照第二章“环境依赖”满足 lsfMonitor 的环境依赖关系。

安装包下的文件和目录如下。

```
[root@ic-admin2 tools]# cd lsfMonitor/
[root@ic-admin2 lsfMonitor]# ls -p
data/ db/ docs/ install.py lib/ LICENSE memPrediction/
monitor/ README.md requirements.txt
```

确认 python 版本正确（Python 3.8.8），并基于安装包中的 requirements.txt 安装 python 依赖库。（此步需要 root 权限并且支持 pip install，推荐在 IC 内网配置 pip 源实现）

```
[root@ic-admin2 lsfMonitor]# python3 --version
Python 3.8.8
[root@ic-admin2 lsfMonitor]#
[root@ic-admin2 lsfMonitor]# pip3 install -r requirements.txt
Looking in indexes: https://bytedpypi.byted.org/simple/
Collecting matplotlib==3.3.4 (from -r requirements.txt (line 1))
  Downloading
https://bytedpypi.byted.org/packages/pypi/matplotlib/matplotlib-
3.3.4-cp38-cp38-manylinux1_x86_64.whl (11.6 MB)
               11.6/11.6 MB
130.5 MB/s eta 0:00:00
```

```
...
Requirement already satisfied: python-dateutil>=2.1 in
/ic/software/tools/python3/3.8.8/lib/python3.8/site-packages (from
matplotlib==3.3.4->-r requirements.txt (line 1)) (2.8.2)
INFO: pip is looking at multiple versions of pyqt5 to determine
which version is compatible with other requirements. This could
take a while.
```

在安装目录下，使用命令“python3 install.py”安装 lsfMonitor。

```
[root@ic-admin2 lsfMonitor]# python3 install.py
>>> Check python version.
    Required python version : (3, 8)
    Current  python version : (3, 8)

>>> Generate script
"/ic/software/tools/lsfMonitor/monitor/bin/bmonitor".
>>> Generate script
"/ic/software/tools/lsfMonitor/monitor/bin/bsample".
>>> Generate script
"/ic/software/tools/lsfMonitor/monitor/tools/akill".
>>> Generate script
"/ic/software/tools/lsfMonitor/monitor/tools/check_issue_reason".
>>> Generate script
"/ic/software/tools/lsfMonitor/monitor/tools/patch".
>>> Generate script
"/ic/software/tools/lsfMonitor/monitor/tools/process_tracer".
>>> Generate script
"/ic/software/tools/lsfMonitor/monitor/tools/seedb".
>>> Generate script
"/ic/software/tools/lsfMonitor/monitor/tools/show_license_feature_
usage".

>>> Generate config file
"/ic/software/tools/lsfMonitor/monitor/conf/config.py".
>>> Install tool "memPrediction" ...

Done, Please enjoy it.
```

3.3 工具配置

安装目录下主要的配置文件为 monitor/conf/config.py，用于配置工具的一些基本设置和验证规则。

安装后默认配置如下，一般需要重新配置。

```
# Specify the database directory.  
db_path = "/ic/software/tools/lsfMonitor/db"  
  
# Specify lmstat path, example  
"/eda/synopsys/scl/2021.03/linux64/bin/lmstat".  
lmstat_path = "/ic/software/tools/lsfMonitor/monitor/tools/lmstat"  
  
# Specify lmstat bsub command, example "bsub -q normal -Is".  
lmstat_bsub_command = ""
```

db_path: 采样数据的数据库存放路径， 默认为 lsfMonitor 安装路径下的 db 目录。

如果 lsfMonitor 用版本管理工具管理，那么建议把 db_path 修改到独立的数据存放路径。

lmstat_path: lsfMonitor 通过工具 lmstat 获取 EDA license 信息， 此处用于配置 lmstat 工具的路径。

lmstat_bsub_command: lsfMonitor 一般在 Linux 环境的 login server 上运行， 而 login server 一般会通过 iptables 等方法设置禁止 lmstat 等 EDA 相关的工具运行， 所以执行 lmstat 的时候需要 bsub 出去， 此处用于指定执行 lmstat 时候的 bsub 命令。

一个使用的 demo 配置如下。

```
# Specify the database directory.  
db_path = "/ic/software/tools/lsfMonitor/db"  
  
# Specify lmstat path, example  
"/eda/synopsys/scl/2021.03/linux64/bin/lmstat".  
lmstat_path = "/ic/software/tools/lsfMonitor/monitor/tools/lmstat"  
  
# Specify lmstat bsub command, example "bsub -q normal -Is".  
lmstat_bsub_command = "bsub -q normal -Is"
```

四、工具使用

lsmMonitor 工具包括“数据采集”和“数据展示”两大部分，对应的执行脚本分别为 bsample 和 bmonitor，均位于 lsmMonitor 安装路径下的 monitor/bin 子目录中。

4.1 数据采集 bsample

4.1.1 帮助信息

bsample 用于采集 LSF/openlava 的 job/queue/host/load/user/utilization 信息。

```
[root@ic-admin2 lsfMonitor]# monitor/bin/bsample -h
usage: bsample.py [-h] [-j] [-m] [-q] [-H] [-l] [-u] [-U] [-UD]

optional arguments:
  -h, --help            show this help message and exit
  -j, --job              Sample (finished) job info with command
"bjobs -u all -d -UF".
  -m, --job_mem         Sample (running) job memory usage
information with command "bjobs -u all -r -UF".
  -q, --queue            Sample queue info with command "bqueues".
  -H, --host             Sample host info with command "bhosts".
  -l, --load              Sample host load (ut/tmp/swp/mem) info
with command "lsload".
  -u, --user              Sample user (finished) job info with
command "bjobs -u all -d -UF".
  -U, --utilization      Sample utilization (slot/cpu/mem) info
with command "lsload/bhosts/lshosts".
  -UD, --utilization_day
                        Count and save utilization-day info with
utilization data.
```

--help: 打印帮助信息。

--job: 采集 job 信息并存储。

--job_mem: 采集 job 的 MEM 用量信息并存储。

--queue: 采集 queue 信息并存储。

--host: 采集 host 信息并存储。 (bmonitor 暂时不需要)

--load: 采集 host load 信息并存储。

- user: 采集 user 信息并存储。
- utilization: 采集 slot/cpu/memory 的 utilization 信息。
- utilization_day: 根据 utilization 数据计算按天核算的 utilization 值。

注意，在 V1.5 及更早的版本中，“bsample -j”是采集 job MEM 用量信息，而在 V1.5.1 及以后版本中，这个功能变更为“bsample -m”。

4.1.2 手工采样

下面是一个手工采样示例，我们使用 monitor/bin/bsample 采样 job MEM 用量的信息。

```
[root@ic-admin2 lsfMonitor]# monitor/bin/bsample -m
>>> Sampling job mem usage info ...
      Done (5744 jobs).
```

4.1.3 自动采样（定时）

首先，建议 LSF 的 CLEAN_PERIOD 参数至少设置为 86400（一天），默认值一般为 3600（一小时），这个值过小的话容易导致部分采样中出现数据缺失。

我们推荐用 crontab 来自动定时采样，job/user 采样较慢所以一天两次即可，其它采样建议 5~10 分钟一次。下面是一个示例。（crontab -e）

```
SHELL=/bin/bash
PATH=/usr/local/bin:/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/ic/software/tools/lsf/10.1/linux3.10-glibc2.17-x86_64/bin
LSF_SERVERDIR=/ic/software/tools/lsf/10.1/linux3.10-glibc2.17-x86_64/etc
LSF_LIBDIR=/ic/software/tools/lsf/10.1/linux3.10-glibc2.17-x86_64/lib
LSF_BINDIR=/ic/software/tools/lsf/10.1/linux3.10-glibc2.17-x86_64/bin
LSF_ENVDIR=/ic/software/tools/lsf/conf
LSF_TOP=/ic/software/tools/lsf

# For lsfMonitor
10 11,23 * * * /ic/software/tools/lsfMonitor/monitor/bin/bsample -
```

```
j
*/5 * * * * /ic/software/tools/lsfMonitor/monitor/bin/bsample -m
*/5 * * * * /ic/software/tools/lsfMonitor/monitor/bin/bsample -q
*/5 * * * * /ic/software/tools/lsfMonitor/monitor/bin/bsample -H
*/5 * * * * /ic/software/tools/lsfMonitor/monitor/bin/bsample -l
30 11,23 * * * /ic/software/tools/lsfMonitor/monitor/bin/bsample -
u
*/10 * * * * /ic/software/tools/lsfMonitor/monitor/bin/bsample -U
55 23 * * * /ic/software/tools/lsfMonitor/monitor/bin/bsample -UD
```

请注意，`crontab` 中默认是没有任何环境的，所以需要在 `crontab` 中设置好 `PATH` 及 `LSF_*` 等变量，否则 `bsample` 中引用的 `bjobs` 等工具无法生效。这些变量可以通过如下方式获取。

```
[root@ic-admin2 lsfMonitor]# echo $PATH
/usr/local/bin:/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/ic/software/tools/lsf/10.1/linux3.10-glibc2.17-x86_64/bin
[root@ic-admin2 lsfMonitor]#
[root@ic-admin2 lsfMonitor]# env | grep "LSF_"
LSF_SERVERDIR=/ic/software/tools/lsf/10.1/linux3.10-glibc2.17-x86_64/etc
LSF_LIBDIR=/ic/software/tools/lsf/10.1/linux3.10-glibc2.17-x86_64/lib
LSF_BINDIR=/ic/software/tools/lsf/10.1/linux3.10-glibc2.17-x86_64/bin
LSF_ENVDIR=/ic/software/tools/lsf/conf
LSF_TOP=/ic/software/tools/lsf
```

4.1.4 数据库

`lsfMonitor` 支持多 LSF/openlava clusters，会根据 `cluster` 来存放数据，所以有可能在 `db_path` 下面看到多个 cluster 的采样数据目录。

测试环境的 `cluster` 信息为“`IC_CLUSTER`”。

```
[root@ic-admin2 lsfMonitor]# lsid
...
My cluster name is IC_CLUSTER
My master name is ic-lsfmaster1
```

所以采样目录为同名目录。

```
[root@ic-admin2 lsfMonitor]# ls db  
IC_CLUSTER
```

采样目录下的数据如下。

```
[root@ic-admin2 lsfMonitor]# ls -p db/IC_CLUSTER/  
host.db  job/  job_mem/  load.db  queue.db  user/  
utilization_day.db  utilization.db  
[root@ic-admin2 lsfMonitor]#  
[root@ic-admin2 lsfMonitor]# ls -p db/IC_CLUSTER/job/  
20240825  20240826  20240908  20240910  20240925  20240926  
20240927  
[root@ic-admin2 lsfMonitor]#  
[root@ic-admin2 lsfMonitor]# ls -p db/IC_CLUSTER/job_mem/  
73970000_73979999.db  90700000_90709999.db  
93550000_93559999.db ...  
[root@ic-admin2 lsfMonitor]#  
[root@ic-admin2 lsfMonitor]# ls -p db/IC_CLUSTER/user/  
20240825  20240826  20240908  20240910  20240925  20240926  
20240927
```

host.db: 记录 host 的静态信息，由“bsample -H”生成。

job/<date>: 记录 job 历史信息，文件为 json 格式，由“bsample -j”生成。

job_mem/*.db: 记录 job 的 mem usage 信息，由“bsample -m”生成。

load.db: 记录 host 的 load 信息，由“bsample -l”生成。

queue.db: 记录 queue 的 run/pending slot 信息，由“bsample -q”生成。

user/<date>: 记录用户的 job 关键信息，有“bsample -u”生成。

utilization_day.db: 记录 slot/cpu/mem 的 utilization 信息，按天汇聚，由“bsample -UD”生成。

utilization.db: 记录 slot/cpu/mem 的 utilization 信息，由“bsample -U”生成。

特别注意：

V1.3 版本开始，bsample 采样数据库的数据格式跟旧版本有变化，且不向前兼容，不能复用旧的数据库。

V1.4 版本开始，bsample 采样 queue.db 数据库格式再次有了变化，且不向前兼容，不能复用旧的数据库。

V1.4.2 版本开始，bsample 采样存储目录名从“monitor”切换为 LSF/openlava 的 cluster 名字，这个切换是无感的，唯一的影响是旧的“monitor”目录下的数据会失效，可以通过手工拷贝到新路径复用旧数据。

V1.5.1 版本开始，bsample 采样 job MEM 用量的存储目录名从“job”切换为“job_mem”，这个切换是无感的，唯一的影响是旧的“job”目录下的数据会失效，可以通过手工拷贝到“job_mem”路径下来复用旧数据。

4.2 数据展示 bmonitor

4.2.1 工具载入

lsfMonitor 的核心工具叫做 **bmonitor**，是一个图形界面工具，其载入方式有多种。

- 引用 bmonitor 绝对路径。
- 将 bmonitor 的路径加入到环境变量 PATH 中，直接执行 bmonitor 即可。
- 采用 modules 管理和加载环境，直接执行 bmonitor 即可。
- 将 bmonitor link 到 LSF 的 bsub 脚本路径中，直接执行 bmonitor 即可。

推荐最后一种方式，下面是具体效果。

```
[root@ic-admin2 lsfMonitor]# which bmonitor  
/ic/software/tools/lsf/10.1/linux2.6-glibc2.3-x86_64/bin/bmonitor
```

其启动效果如下所示（测试环境耗时约 20 秒，时间主要耗费在 LSF 实时/历史数据加载上）。

```
[root@ic-admin2 bin]# bmonitor  
[2024-11-08 10:08:40] lsfMonitor Version: V1.6 (2024.11.07)  
[2024-11-08 10:08:40]  
[2024-11-08 10:08:40] LSF (10.1.0.12)  
[2024-11-08 10:08:40] My cluster name is "IC_CLUSTER"  
[2024-11-08 10:08:40] My master name is "ic-lsfmaster1"  
[2024-11-08 10:08:40]  
[2024-11-08 10:08:40] Loading License information, wait a
```

```

moment ...
[2024-11-08 10:08:40] *Warning*: Not find any valid license
information.
[2024-11-08 10:08:40] Loading LSF bqueues information, wait a
moment ...
[2024-11-08 10:08:40] Loading LSF bhosts information, wait a
moment ...
[2024-11-08 10:08:40] Loading LSF busers information, wait a
moment ...
[2024-11-08 10:08:40] Loading LSF jobs information, wait a
moment ...
[2024-11-08 10:08:44] Loading LSF lshosts information, wait a
moment ...
[2024-11-08 10:08:44] Loading LSF host_queue information, wait a
moment ...
[2024-11-08 10:08:50] Loading LSF lsload information, wait a
moment ...
[2024-11-08 10:08:50] Loading LSF bhosts_load information, wait a
moment ...
[2024-11-08 10:08:50] Loading user history info, wait a moment ...
[2024-11-08 10:08:58] Loading LSF queue_host information, wait a
moment ...
[2024-11-08 10:09:05] Loading queue utilization info, wait a
moment ...
[2024-11-08 10:09:08] Loading resource utilization information,
wait a moment ...

```

lsfMonitor V1.7										
File		Setup		Function		Help				
JOB		JOBS		HOSTS		LOAD		USERS		QUEUES
Status		Queue		Host		User		Check		
Job	User	Status	Queue	Host	Started	Project	Slot	Rusage (G)	Mem (G)	Command
1 29841720	wangzhangwei	RUN	interactive	n249-073-094	2024-12-04 10:36:47		1.5	0	vscode	
2 31702687	wangzhangwei	RUN	interactive	n249-073-094	2024-12-09 14:08:07			0.9	vscode	
3 34055939	wangzhangwei	RUN	interactive	n249-073-094	2024-12-13 19:18:54		2	0	vscode	
4 35367517	wangzhangwei	RUN	interactive	n249-073-094	2024-12-16 20:08:31		2	0.7	vscode --extensions-dir /ic/software/...	
5 45366428	wangzhangwei	RUN	interactive	n249-073-094	2024-12-30 19:47:03		2	0	vscode	
6 45438687	wangzhangwei	RUN	interactive	n249-073-094	2024-12-30 21:22:16		2	0.5	vscode	
7 54913934	wangzhangwei	RUN	interactive	n249-073-094	2025-01-13 16:17:51		2	0	vscode -V	
8 60372402	wangzhangwei	RUN	interactive	n249-073-094	2025-01-23 14:50:49		2	0.8	vscode	
9 77642490	wangzhangwei	RUN	interactive	n249-073-094	2025-02-10 14:04:28		2	0	vscode ./	
10 80712383	wangzhangwei	RUN	interactive	n249-073-094	2025-02-16 14:13:45		2	0.1	vscode	
11 86141139	wangzhangwei	RUN	interactive	n249-073-094	2025-02-27 16:06:45		2	0.3	vscode .	
12 88621601	wangzhangwei	RUN	interactive	n249-073-094	2025-03-05 15:45:26		2	0.4	vscode	
13 90661079	wangzhangwei	RUN	interactive	n249-073-094	2025-03-10 17:15:19		2	3.6	vscode	
14 90959561	wangzhangwei	RUN	interactive	n249-073-094	2025-03-11 10:21:51		2	2.8	vscode	
15 91405028	wangzhangwei	RUN	interactive	n249-073-094	2025-03-12 16:09:05		2	0.7	vscode	

我们可以看到当前集群的基本信息，以及图形界面启动过程中加载数据的过程。

4.2.2 帮助信息

直接执行 bmonitor 会启动图形界面。

执行“bmonitor -h”则可以查看 bmonitor 的帮助信息， bmonitor 的参数主要用于初始化部分信息， 不过这些参数一般也可以在 bmonitor 启动后设置。

```
[root@ic-admin2 lsfMonitor]# bmonitor -h
usage: bmonitor.py [-h] [-j JOBID] [-u USER] [-f FEATURE] [-t
{JOB,JOBS,HOSTS,LOAD,USERS,QUEUES,UTILIZATION,LICENSE}] [--disable_license] [-d]

optional arguments:
  -h, --help            show this help message and exit
  -j JOBID, --jobid JOBID
                        Specify the jobid which show it's
                        information on "JOB" tab.
  -u USER, --user USER  Specify the user show how's job
                        information on "JOBS" tab.
  -f FEATURE, --feature FEATURE
  -t {JOB,JOBS,HOSTS,LOAD,USERS,QUEUES,UTILIZATION,LICENSE}, --tab
{JOB,JOBS,HOSTS,LOAD,USERS,QUEUES,UTILIZATION,LICENSE}
                        Specify current tab, default is "JOBS"
tab.
  --disable_license     Disable license check function.
  -d, --dark_mode       Enable dark mode on the main interface.
```

--help: 打印帮助信息。

--jobid: 指定 jobid， 用于切换到 JOB 页并直接显示指定 jobid 的信息， 此时其它页内容并不加载， 以加快 GUI 打开速度。

--user: 指定 user， 用于切换到 JOBS 页并显示指定用户的所有 job 信息。

--feature: 指定 license feature， 用于切换到 LICENSE 页并显示指定 license feature 的信息。

--tab {JOB, JOBS, HOSTS, QUEUES, LOAD, UTILIZATION, LICENSE}: 指定页面， 会将 bmonitor 打开到指定 GUI 页面。

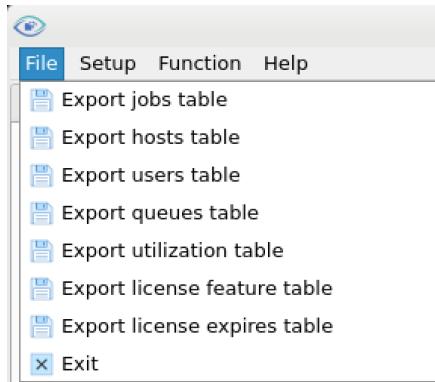
--disable_license: 启动的时候不执行 license 信息获取步骤， 以加快 GUI 打开速度。

--dark_mode: 启用暗黑主题模式。

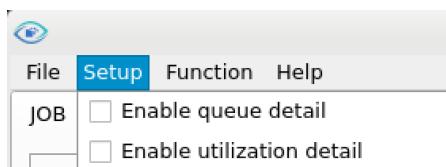
4.2.3 菜单栏

bmonitor 菜单栏包含 File, Setup, Function, Help 四部分。

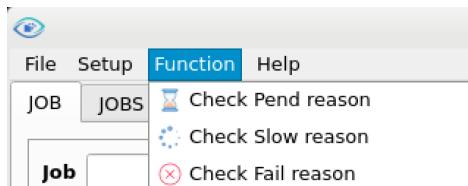
File: 包含 Export * table 功能和 Exit 功能。



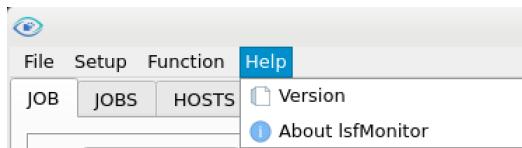
Setup: 包含“Enable queue detail”和“Enable utilization detail”两个复选框。



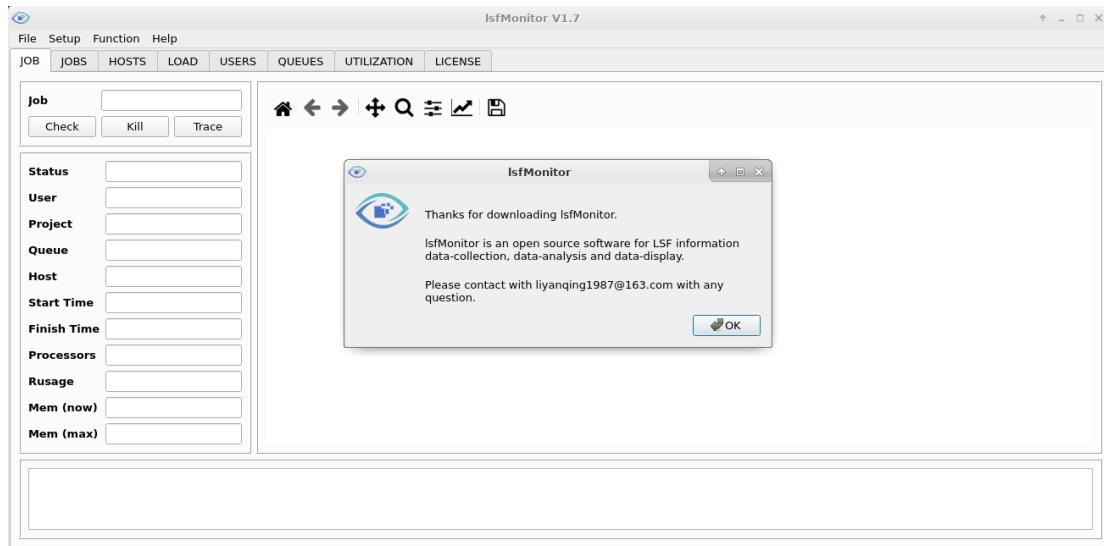
Function: 包含“Check Pend reason”、“Check Slow reason”和“Check Fail reason”三个功能。



Help: 包含“Version”和“About lsfMonitor”两个信息项。



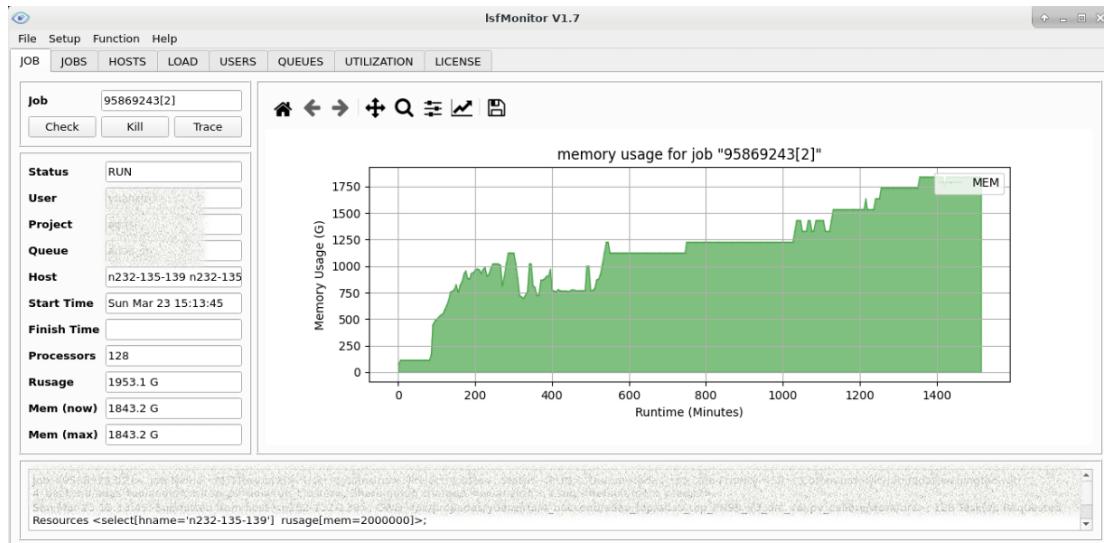
Help 菜单中的“About lsfMonitor”内容如下。



4.2.4 JOB 页

JOB 页主要用于查看指定 job 的详细信息，以及 job 内存用量的历史曲线。

在 Job 框输入 jobid，点击 Check 按钮，可以查看指定 job 的详细信息（来源于 bjob -UF <JOBID>）和 job 的内存用量曲线。



说明：

- 左上侧显示 job 的关键信息。
- 右上侧显示 job 在生命周期内的 memory 用量曲线，如未显示，可能是没有启动周期性的采样，或者采样了但是 job 的 runtime 太短未采到。

- 下侧显示 job 的详细信息。(通过 bjobs -UF <jobid>获取)
- 点击“Kill”按钮，在有权限的情况下会 kill 掉当前 job，
- 点击“Trace”按钮，会追踪当前 job pend/slow/fail 的原因。
- 在有 job 信息的情况下，在 Host 框中敲击回车，会跳转到 LOAD 页，并显示第一台 server 在 Start Time 和 Finish Time 之间的负载变化趋势。

4.2.5 JOBS 页

JOBS 页主要用于批量查看 jobs 的关键信息。

IsfMonitor V1.7										
File Setup Function Help										
JOB JOBS HOSTS LOAD USERS QUEUES UTILIZATION LICENSE										
Status	RUN	Queue	ALL	Host	ALL	Project	Slot	Rusage (G)	Mem (G)	Command
1	29841720	RUN	interactive	n249-073-094	2024-12-04 10:36:47			1.5	0	vscode
2	31702687	RUN	interactive	n249-073-094	2024-12-09 14:08:07			0.9	0.9	vscode
3	34055939	RUN	interactive	n249-073-094	2024-12-13 19:18:54		2	0	vscode	
4	35367517	RUN	interactive	n249-073-094	2024-12-16 20:08:31		2	0.7	vscode --extensions-dir /ic/software/...	
5	45366428	RUN	interactive	n249-073-094	2024-12-30 19:47:03		2	0	vscode	
6	45438687	RUN	interactive	n249-073-094	2024-12-30 21:22:16		2	0.5	vscode	
7	54913934	RUN	interactive	n249-073-094	2025-01-13 16:17:51		2	0	vscode -V	
8	60372402	RUN	interactive	n249-073-094	2025-01-23 14:50:49		2	0.8	vscode	
9	77642490	RUN	interactive	n249-073-094	2025-02-10 14:04:28		2	0	vscode ./	
10	80712383	RUN	interactive	n249-073-094	2025-02-16 14:13:45		2	0.1	vscode	
11	86141139	RUN	interactive	n249-073-094	2025-02-27 16:06:45		2	0.3	vscode .	
12	88621601	RUN	interactive	n249-073-094	2025-03-05 15:45:26		2	0.4	vscode	
13	90661079	RUN	interactive	n249-073-094	2025-03-10 17:15:19		2	3.6	vscode	
14	90959561	RUN	interactive	n249-073-094	2025-03-11 10:21:51		2	2.8	vscode	
15	91405028	RUN	interactive	n249-073-094	2025-03-12 16:09:05		2	0.7	vscode	

说明：

- Job 列的 jobid 左击，可以直接跳转到 JOB 页，并展示 job 的信息。(粗体字均可点击跳转，后同)
- 点击任意列标题，可以排序列内容。
- 如果 job Rusage 没有设，或者 Rusage 小于 Mem (job 的实际内存用量) 的值，Mem 值背景色会变红。
- 如果 Status 为 RUN，点击 RUN 单元格，bmonitor 会调用工具“Check Issue Reason”来查看 job SLOW 的原因。
- 如果 Status 为 PEND，点击 PEND 单元格，bmonitor 会调用工具“Check Issue Reason”来查看 job PEND 的原因。
- 如果 Status 为 EXIT，点击 EXIT 单元格，bmonitor 会调用工具“Check Issue Reason”来查看 job FAIL 的原因。

4.2.6 HOSTS 页

HOSTS 页主要用于查看 hosts 的静态和动态信息。

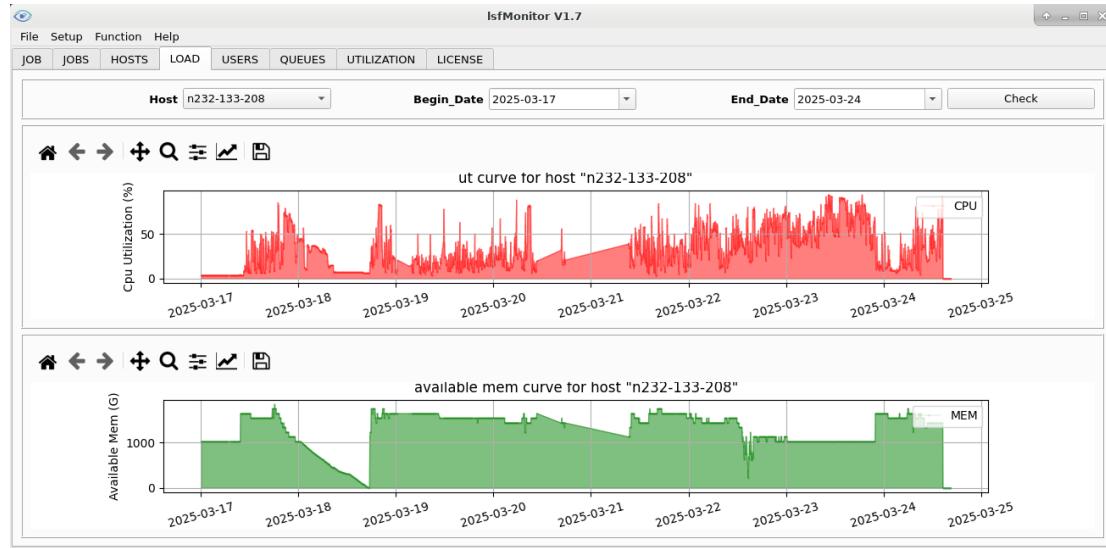
Host	Status	Queue	MAX	Njobs	Ut (%)	MaxMem (G)	aMem (G)	saMem (G)	MaxSwp (G)	Swp (G)	Tmp (G)
1 ic-lsfmaster1	ok		8	0	24	94	78	79	7	7	27
2 ic-lsfmaster2	ok		8	0	39	62	39	39	7	7	14
3 n232-129-012	ok		64	50	2	1945	1638	798	127	127	1504
4 n232-129-013	ok		96	48	2	1945	1228	1228	127	127	1071
5 n232-129-017	ok		64	50	14	1945	1740	1126	127	127	1508
6 n232-129-018	ok		64	59	2	1945	1740	1638	127	123	1507
7 n232-129-019	ok		64	48	5	1945	1638	1433	127	127	1505
8 n232-129-020	ok		64	54	13	1945	1945	1740	127	126	1507
9 n232-129-021	ok		64	61	4	1945	1638	361	127	127	1480
10 n232-129-022	ok		64	54	0	1945	1740	1536	127	127	1501
11 n232-129-023	ok		64	46	3	1945	1536	677	127	127	1491
12 n232-129-024	ok		64	52	85	1945	1536	1024	127	104	1503
13 n232-129-025	ok		64	51	35	1945	1433	1228	127	127	1504
14 n232-129-026	ok		64	53	9	1945	713	276	127	118	1071
15 n232-129-027	ok		128	4	1	1945	1536	1433	127	127	1509

说明：

- “aMem”是指系统的 available memory，即系统上实际的可用内存。
- “saMem”是指 scheduling available memory，即调度器所判断的可用内存，是 aMem 减去 rusage_mem 之后的内存值。
- MaxMem \geq aMem \geq saMem
- Host 列的 hostname 左击，可以跳转到 LOAD 页，展示指定 host 的 cpu 和 memory 历史用量曲线。
- Host 列的 hostname 右击，可以弹出“Open”和“Close”两个选项，分别用于 hopen 和 hclose 当前 host，生效的前提是当前用户具有 LSF 管理员权限。
- Njobs 列的数字左击，可以跳转到 JOBS 页，展示指定 host 上所有的 RUN jobs。
- 点击任意列标题，可以排序列内容。
- 如果 host 的 Status 异常 (unavail/unreach/closed_LIM)，Status 状态背景色会变红。
- 如果 host 的 Ut 使用率超过 90%，Ut 值背景色会变红。
- 如果 host 的 memory 可用容量不足 10%，Mem 值背景色会变红。
- 如果 host 的 tmp 可用量变为 0，Tmp 值背景色会变红。

4.2.7 LOAD 页

LOAD 页主要用于查看 host 的 ut 和 memory 负载信息。



4.2.8 USERS 页

USERS 页主要用于查看用户 job 相关的统计信息。

The table provides a detailed breakdown of user resource usage over the specified date range (March 23-24, 2025). The columns include User ID, Job Num, Pass Rate (%), Total_Rusage_Mem (G), Avg_Rusage_Mem (G), Total_Max_Mem (G), Avg_Max_Mem (G), Total_Mem_Waste (G), and Avg_Mem_Waste (G).

User	Job_Num	Pass_Rate (%)	Total_Rusage_Mem (G)	Avg_Rusage_Mem (G)	Total_Max_Mem (G)	Avg_Max_Mem (G)	Total_Mem_Waste (G)	Avg_Mem_Waste (G)
1	710320	98.6	3131868.1	4.4	2325033.1	3.3	806835.0	1.1
2	2402	100.0	2472.7	1.0	1577.0	0.7	895.7	0.4
3	4669	100.0	23827.2	5.1	4059.8	0.9	19767.4	4.2
4	388016	100.0	747501.4	1.9	249686.7	0.6	497814.7	1.3
5	14	78.6	644.5	46.0	1040.1	74.3	-395.6	-28.3
6	20	80.0	1000.0	50.0	716.2	35.8	283.8	14.2
7	2334	99.8	23301.0	10.0	1529.8	0.7	21771.2	9.3
8	3	100.0	205.1	68.4	139.6	46.5	65.5	21.8
9	1660	100.0	628.2	0.4	870.9	0.5	-242.7	-0.1
10	8	75.0	3523.4	440.4	1414.5	176.8	2108.9	263.6
11	12	91.7	20.8	1.7	13.5	1.1	7.3	0.6
12	15	66.7	18.5	1.2	4.6	0.3	13.9	0.9
13	21	71.4	615.2	29.3	32.7	1.6	582.5	27.7
14	48	41.7	3243.0	67.6	473.0	9.9	2770.0	57.7

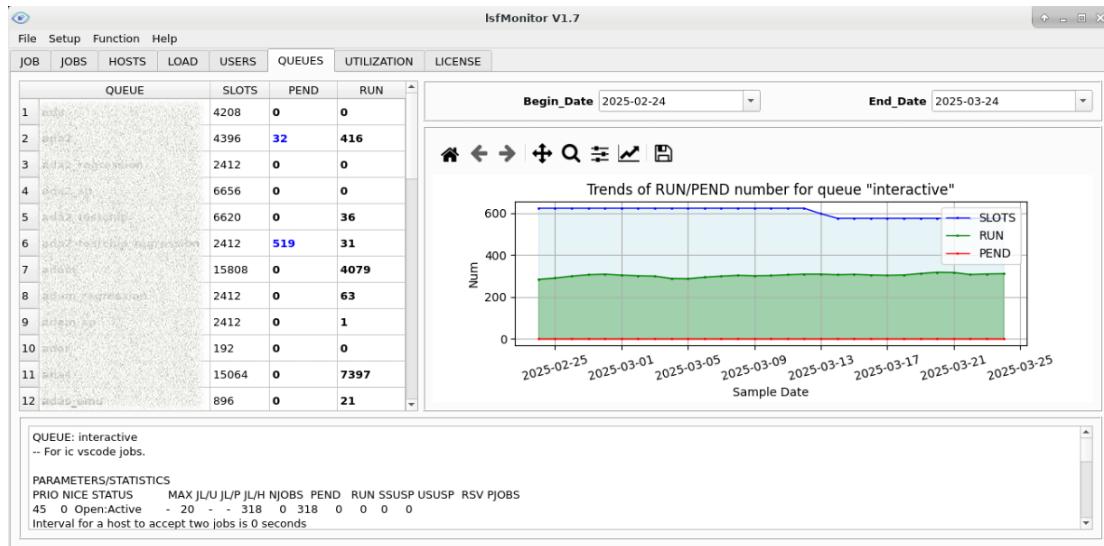
说明：

- Job_Num 是在指定时间段内用户所有 DONE/EXIT 任务的总量。
- Pass_Rate 是在指定时间段内用户 DONE 任务数量/job_Num 数量 的比值。
- Total_Mem_Waste 是在指定时间段内 Total_Rusage_Mem - Total_Max_Mem 的值，标识用户申请了但未使用而造成的内存浪费的量。

- 点击任意列标题，可以排序列内容。

4.2.9 QUEUES 页

QUEUES 页主要用于查看所有 queue 的实时和历史信息。

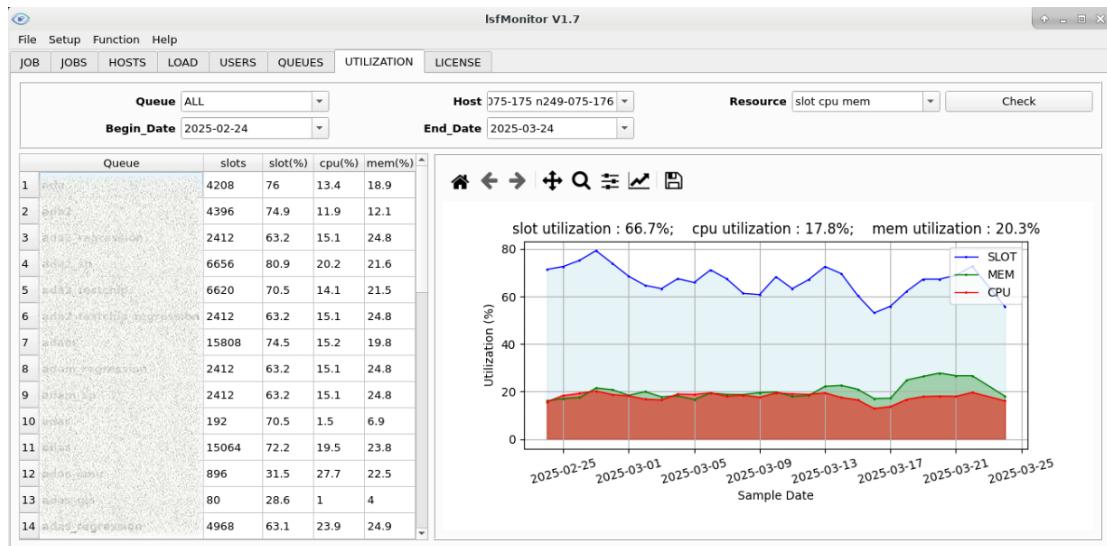


说明：

- 点击 QUEUE 列的队列名，可以展示 queue 的详细信息和 queue 中 SLOTS/RUN/PEND 数据的变化曲线。
- 点击 PEND 列的数字，可以跳转到 JOBS 页，展示指定 queue 上所有的 PEND jobs。（点击 RUN 列数字亦然）
- 如果想看更细致的数据，可以在菜单栏 Setup 中选中“Enable queue detail”选项，注意此时采样周期被默认重置为最近一周。
- 如果 queue 中 PEND 的 job 数目不为 0，数字会被红标。

4.2.10 UTILIZATION 页

UTILIZATION 页主要用来查看 slot/cpu/memory 等资源的使用率统计信息，信息周期默认为最近一个月。



说明：

- 点击 QUEUE 列的队列名，可以展示 queue 的 slot/cpu/mem 使用率的变化曲线。
- 右侧的曲线图是展示指定“Queue”在一定时间段内的资源（slot/cpu/mem）用量变化曲线。
- 点击任意列标题，可以排序列内容。
- 如果想看更细致的数据，可以在菜单栏 Setup 中选中“Enable utilization detail”选项，注意此时采样周期被默认重置为最近一周。
- 左侧和右侧的 utilization 统计值有可能会有所差别，尤其是在队列机器有变更（增加/减少）的情况，这是因为左侧结果是按照“ $\text{sum}(\text{服务器利用率})/\text{len}(\text{服务器数目})$ ”计算出来的，右侧结果是按照“ $\text{sum}(\text{整体按天汇聚利用率})/\text{天数}$ ”计算出来的，理论上左侧表格中的使用率信息更准确一些。

4.2.11 LICENSE 页

LICENSE 页主要用于查看 EDA license 的使用情况。

启动 lsfMonitor 前，需要保证当前 terminal 中 LM_LICENSE_FILE 环境变量配置正确，bmonitor 将环境变量 LM_LICENSE_FILE 作为唯一的 license server 设定来源。

Server	Vendor	Feature	Issued	In_Use	License Server	Feature	Num	Expires
1 5280@ic-lic02	cdslmd	LicFileVersion	1	0	1 5280@ic-lic02	LicFileVersion	1	permanent(...)
2 5280@ic-lic02	cdslmd	Use_Server_Options	1	0	2 5280@ic-lic02	Use_Server_Options	1	permanent(...)
3 5280@ic-lic02	cdslmd	Affirma_sim_analysis_env	3	0	3 5280@ic-lic02	Affirma_sim_analysis_env	1	14-jul-2121
4 5280@ic-lic02	cdslmd	Allegro_Viewer_Plus	1	0	4 5280@ic-lic02	Affirma_sim_analysis_env	1	23-mar-2122
5 5280@ic-lic02	cdslmd	Allegro_performance	1	0	5 5280@ic-lic02	Affirma_sim_analysis_env	1	25-jul-2120
6 5280@ic-lic02	cdslmd	Allegro_studio	1	0	6 5280@ic-lic02	Allegro_Viewer_Plus	1	27-sep-2119
7 5280@ic-lic02	cdslmd	Base_Verilog_Lib	1	0	7 5280@ic-lic02	Allegro_performance	1	27-sep-2119
8 5280@ic-lic02	cdslmd	COSLITE_ACCESS	1	0	8 5280@ic-lic02	Allegro_studio	1	27-sep-2119
9 5280@ic-lic02	cdslmd	Concept_HDL_studio	1	0	9 5280@ic-lic02	Base_Verilog_Lib	1	27-sep-2119
10 5280@ic-lic02	cdslmd	Extended_Verilog_Lib	1	0	10 5280@ic-lic02	COSLITE_ACCESS	1	23-mar-2122
11 5280@ic-lic02	cdslmd	Integrated_Metrics_Center	3	2	11 5280@ic-lic02	Concept_HDL_studio	1	27-sep-2119
12 5280@ic-lic02	cdslmd	PCB_design_studio	1	0	12 5280@ic-lic02	Extended_Verilog_Lib	1	27-sep-2119
13 5280@ic-lic02	cdslmd	Palladium_XP_HDL_Checking	1	0	13 5280@ic-lic02	Integrated_Metrics_Center	1	14-jul-2121
14 5280@ic-lic02	cdslmd	Palladium_Z1_Compiler	2	0	14 5280@ic-lic02	Integrated_Metrics_Center	1	23-mar-2122

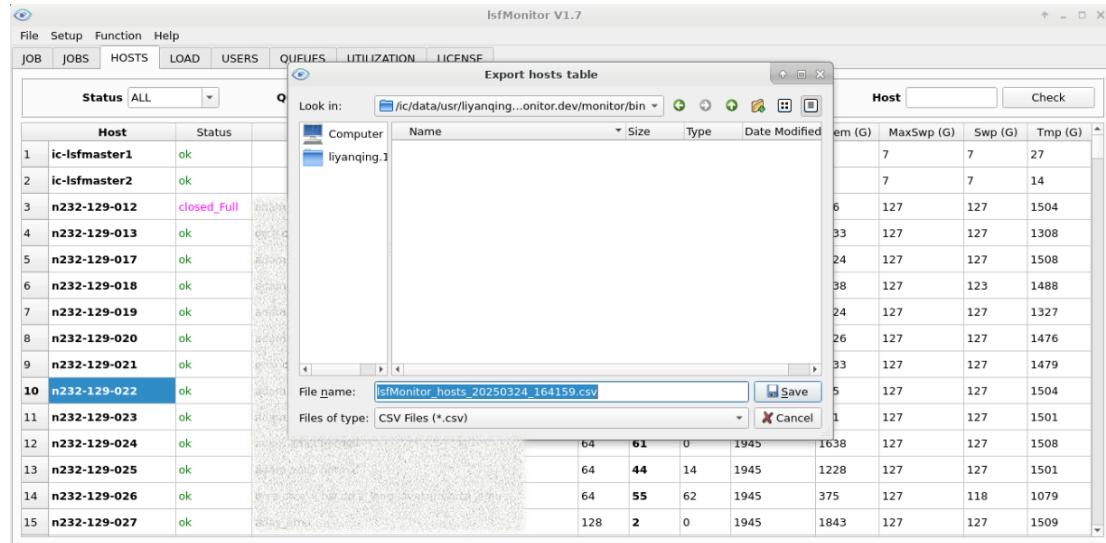
说明：

- 左侧 Feature Information 表格中“In_Use”列的内容，如果非零，左击可以弹出 license feature 的使用详情。
- 点击任意列标题，可以排序列内容。
- 其中 START_TIME 启动时间在 3 天以前的，日期会标红。
- 右侧 Expires Information 表格中“Expires”列的内容，如果已过期，显示为灰色字体；如果两周内过期，显示为红色字体；如果未过期，显示为黑色字体。

4.2.12 其它功能介绍

4.2.12.1 保存表格

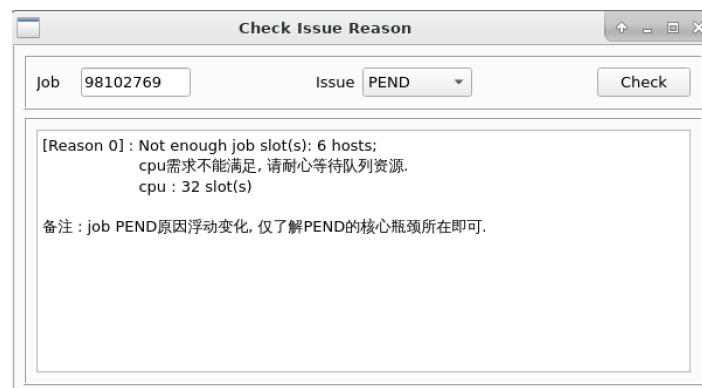
通过菜单栏 File -> Export * table, 可以将 bmonitor 中的任意表格保存为 csv 文件。



4.2.12.2 检查 job PEND 的原因

以下两种方式可以触发“Check Issue Reason”工具来查找指定 job PEND 的原因：

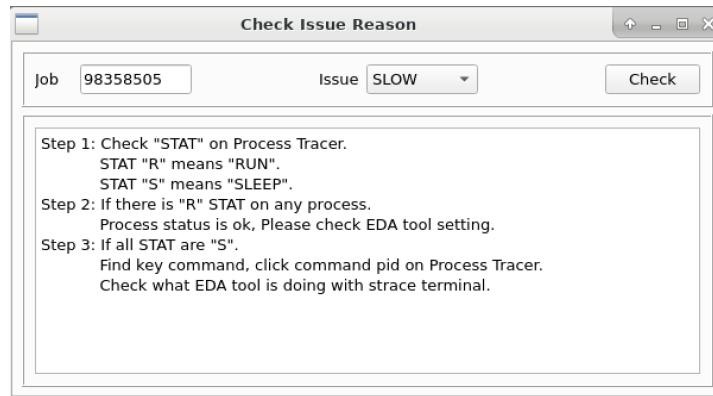
- 菜单栏 Function -> Check Pend reason。
- 点击 JOBS 页 Status 列的 PEND 单元格。



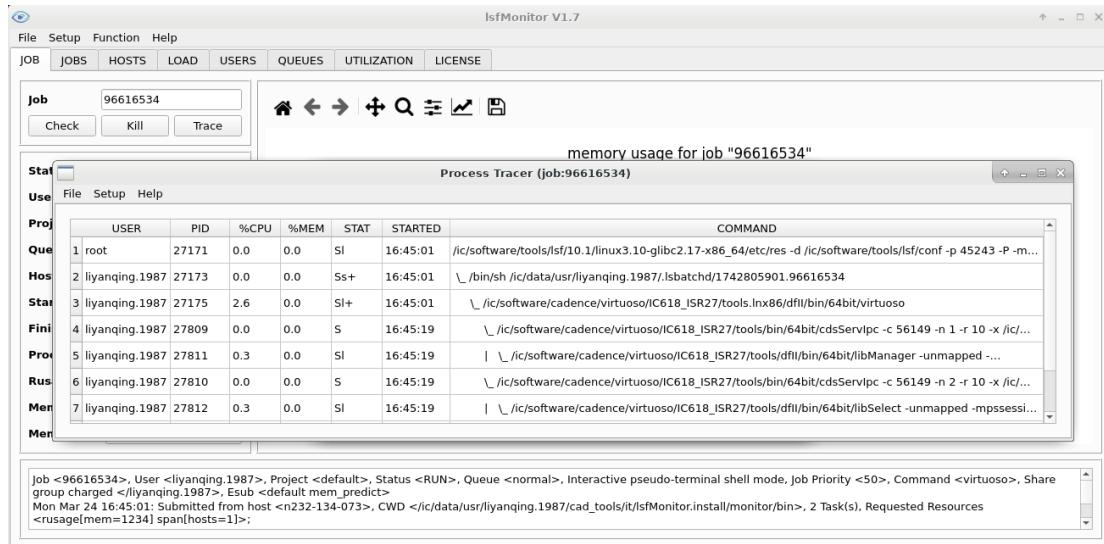
4.2.12.3 检查 job 运行缓慢的原因

以下三种方式可以触发“Check Issue Reason”工具来查找指定 job 运行缓慢的原因：

- 菜单栏 Function -> Check Slow reason。
- JOB 页 job 的 Status 为 RUN 时，点击“Trace”按钮。
- 点击 JOBS 页 Status 列的 RUN 单元格。

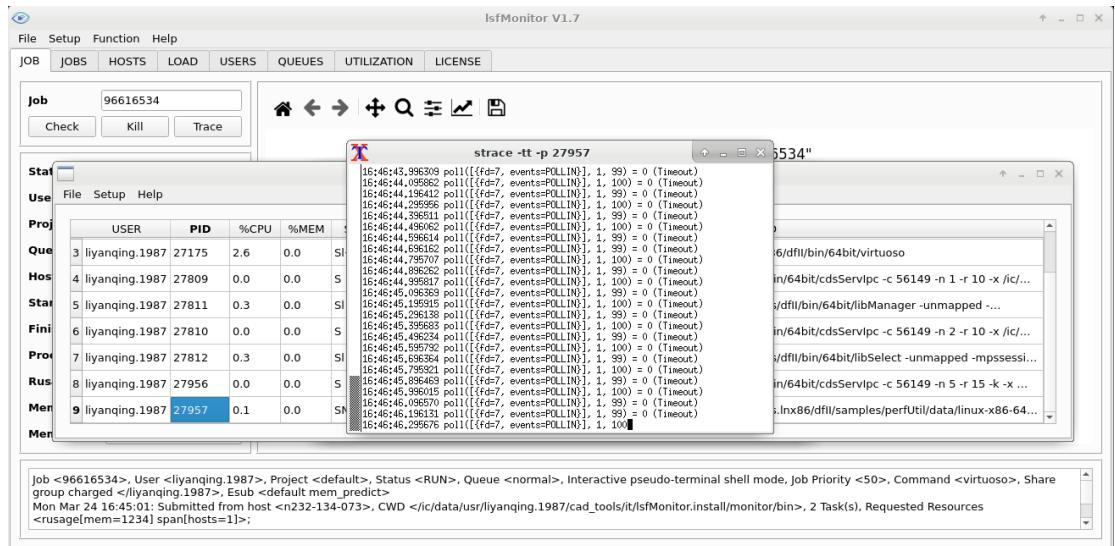


同时会弹出当前 job 的进程树信息。(一般普通用户只能 trace 自己的 job 进程树)



正常情况下应该至少有一个核心进程保持“R”的运行状态，如果所有的进程均为“S”的睡眠状态则可能意味着异常等待。

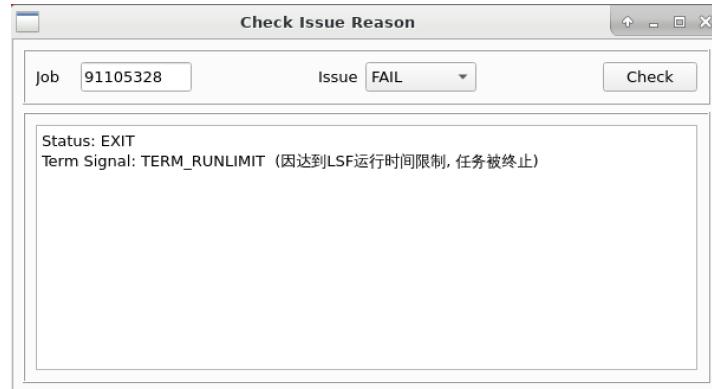
点击 PID 号，可以查看指定进程的系统交互信息，以方便判断进程运行缓慢的原因。



4.2.12.4 检查 job 非正常退出的原因

以下两种方式可以触发“Check Issue Reason”工具来查找指定 job 非正常退出的原因：

- 菜单栏 Function -> Check Pend reason。
- 点击 JOBS 页 Status 列的 EXIT 单元格。



4.2.12.5 查看 EDA license feature 的使用详情

在 LICENSE 页，如果左侧“Feature Information”的“In_Use”列的值不为零，那么点击“In_Use”值可以展示当前 feature 的使用信息。

The screenshot shows the lsfMonitor V1.7 interface. The main window has tabs for JOB, JOBS, HOSTS, LOAD, USERS, QUEUES, UTILIZATION, and LICENSE. The LICENSE tab is selected. Below the tabs are filters for Show (ALL), Server (ALL), Vendor (ALL), Feature (empty), and User (empty). A large central table displays "VCS-BASE-RUNTIME" usage on 27020@ic-lic09/snpslmd. The columns are: USER, SUBMIT_HOST, EXECUTE_HOST, NUM, VERSION, START_TIME, and JOB. Rows 34 through 44 show various check-in records. Row 13 shows a summary for cdsImd and row 14 for snpslmd. To the right of the main table is a smaller table titled "Num Expires" with 15 rows, each containing a value of 1 and a date from 01-Mar-2026 to 01-mar-2026.

USER	SUBMIT_HOST	EXECUTE_HOST	NUM	VERSION	START_TIME	JOB	
34	N/A	n232-132-017	1	v2023.03	Fri 3/21 16:33	*	
35	N/A	n232-134-024	1	v2023.03	Fri 3/21 16:36	94950055	
36	N/A	n232-134-022	1	v2023.03	Fri 3/21 16:40	94950054	
37	N/A	n232-132-200	1	v2023.03	Fri 3/21 16:43	94950052	
38	N/A	n232-132-017	1	v2023.03	Fri 3/21 16:47	*	
39	N/A	n232-132-030	1	v2023.03	Fri 3/21 16:47	94950051	
40	N/A	n249-075-159	1	v2023.03	Fri 3/21 19:33	*	
41	N/A	n249-075-159	1	v2023.03	Fri 3/21 19:33	*	
42	N/A	n249-075-154	1	v2023.03	Fri 3/21 22:11	95048626	
43	N/A	n249-075-155	1	v2023.03	Fri 3/21 23:09	95154720	
44	N/A	n249-075-132	1	v2023.03	Fri 3/21 23:21	95154725	
13	cdsImd	Innovus_CPU_Opt	250	249	13	27020@ic-lic01	DWC-PCIE-G5-PREM-A-V2-SRC
14	snpslmd	PrimeTime-SI	290	237	14	27020@ic-lic01	DWC-PCle-50-NS-A00-555LPE-X4

Num	Expires
1	01-Mar-2026

每一行为一个 check-in 记录，JOB 列记录着当前 check-in 行为对应的 job，如果 job 单元格为空说明 job 完成但 license 没释放，job 单元格为“*”说明找到当前 check-in 找到多个可能对应的 job，无法确切判断。

五、辅助工具

lsmMonitor 自带一些工具，大多数是其功能组件，也有部分是可以单独使用的工具。

```
[root@ic-admin2 lsfMonitor]# ls monitor/tools/
akill      check_issue_reason    lmstat      patch
process_tracer   seedb      show_license_feature_usage
akill.py  check_issue_reason.py message.py  patch.py
process_tracer.py seedb.py  show_license_feature_usage.py
```

akill: 独立工具，是 bkill 的增强型功能。

check_issue_reason: 组件，图形界面工具，查看 job PEND/FAIL/SLOW 的原因。

lmsta: 独立工具，用于检索 EDA license 信息。

message.py: 组件，图形界面工具，显示指定信息。

patch: 独立工具，用于更新工具安装包。

process_tracer: 组件，图形界面工具，追踪指定 process 或 jobid 的进程树。

seedb: 独立工具，查看 sqlite3 数据库内容，lsmMonitor 即使用 sqlite3 保存相关数据。

show_license_feature_usage: 组件，图形界面工具，查看 feature 的使用情况。

5.1 akill

akill 是 bkill 的增强型工具，位于安装目录下的 monitor/tools/akill，可以根据 jobid/job_name/command/submit_time/execute_host/queue/user 等维度来便捷地 kill job，其帮助信息如下。

```
[root@ic-admin2 lsfMonitor]# monitor/tools/akill -h
usage: akill.py [-h] [-j JOBID [JOBID ...]] [-J JOB_NAME
[JOB_NAME ...]] [-c COMMAND [COMMAND ...]] [-s SUBMIT_TIME
[SUBMIT_TIME ...]] [-m EXECUTE_HOST [EXECUTE_HOST ...]]
[-q QUEUE [QUEUE ...]] [-u USER [USER ...]]

optional arguments:
  -h, --help            show this help message and exit
  -j JOBID [JOBID ...], --jobid JOBID [JOBID ...]
                        kill specified job(s) based on jobid(s),
                        support fuzzy matching, also support jobid range like "10200-
                        10450".
```

```
-J JOB_NAME [JOB_NAME ...], --job_name JOB_NAME [JOB_NAME ...]
          kill specified job(s) based on
job_name(s), support fuzzy matching.
-c COMMAND [COMMAND ...], --command COMMAND [COMMAND ...]
          kill specified job(s) based on command(s),
support fuzzy matching.
-s SUBMIT_TIME [SUBMIT_TIME ...], --submit_time SUBMIT_TIME
[SUBMIT_TIME ...]
          kill specified job(s) based on
submit_time(s), support fuzzy matching.
-m EXECUTE_HOST [EXECUTE_HOST ...], --execute_host EXECUTE_HOST
[EXECUTE_HOST ...]
          kill specified job(s) based on execute
host(s).
-q QUEUE [QUEUE ...], --queue QUEUE [QUEUE ...]
          kill specified job(s) based on queue(s).
-u USER [USER ...], --user USER [USER ...]
          kill specified job(s) based on user(s).
```

akill 更详细的用法, 请参照 lsfMonitor 安装目录下的 docs/akill_user_manual.pdf。

5.2 patch

patch 是帮助 lsfMonitor 打补丁的工具, 其帮助信息如下。

```
[root@ic-admin2 lsfMonitor]# monitor/tools/patch -h
usage: patch.py [-h] [-p PATCH_PATH]

optional arguments:
  -h, --help            show this help message and exit
  -p PATCH_PATH, --patch_path PATCH_PATH
                        Specify patch path (new install package
path).
```

--patch_path: 指定补丁包（也就是新的安装包）路径。

一般而言, lsfMonitor 的版本变更, 主要是新增 python 脚本, 或者是现有 python 脚本内容变更, 针对这种变更的安装包, patch 可以自动将新的安装包变更更新到当前安装包。

下面是一个示例。

```
[root@ic-admin2 lsfMonitor]# monitor/tools/patch -p
/ic/data/usr/liyanqing.1987/test/lsfMonitor-master
Install Path : /ic/data/usr/liyanqing.1987/lsfMonitor
Patch path : /ic/data/usr/liyanqing.1987/lsfMonitor-master

*Warning*: current install path name is "lsfMonitor", but patch
path name is "lsfMonitor-master".
Do you want to continue? (y|n) y

> Copying python file "/ic/data/usr/liyanqing.1987/lsfMonitor-
master/monitor/bin/bsample.py" into
"/ic/data/usr/liyanqing.1987/test/lsfMonitor/monitor/bin/bsample.p
y".
> Copying python file "/ic/data/usr/liyanqing.1987/lsfMonitor-
master/monitor/common/common_license.py" into
"/ic/data/usr/liyanqing.1987/test/lsfMonitor/monitor/common/common
_license.py".
> Copying python file "/ic/data/usr/liyanqing.1987/lsfMonitor-
master/monitor/tools/seedb.py" into
"/ic/data/usr/liyanqing.1987/test/lsfMonitor/monitor/tools/seedb.p
y".
> Copying python file "/ic/data/usr/liyanqing.1987/lsfMonitor-
master/monitor/tools/patch.py" into
"/ic/data/usr/liyanqing.1987/test/lsfMonitor/monitor/tools/patch.p
y".
> Copying python file "/ic/data/usr/liyanqing.1987/lsfMonitor-
master/monitor/tools/tool_1.py" into
"/ic/data/usr/liyanqing.1987/test/lsfMonitor/monitor/tools/tool_1.
py".
```

5.3 seedb

seedb 是查看 sqlite3 文本数据库内容的工具，其帮助信息如下：

```
[root@ic-admin2 lsfMonitor]# monitor/tools/seedb -h
usage: seedb.py [-h] -d DATABASE [-t TABLES [TABLES ...]] [-k KEYS
[KEYS ...]] [-n NUMBER]

optional arguments:
-h, --help            show this help message and exit
```

```
-d DATABASE, --database DATABASE
                                Required argument, specify the database
file.

-t TABLES [TABLES ...], --tables TABLES [TABLES ...]
                                Specify the tables you want to review,
make sure the tables exist.

-k KEYS [KEYS ...], --keys KEYS [KEYS ...]
                                Specify the table keys you want to review,
make sure the table keys exist.

-n NUMBER, --number NUMBER
                                How many lines you want to see.
```

--database: 指定 sqlite3 数据库文件。

--tables: 指定想查看的数据库 tables。

--keys: 指定想查看的 table 关键词。

--number: 指定查看的行数，如果内容很多，可以截取有限的内容查看。

示例一，查看 load.db 数据库中的表。

```
[root@ic-admin2 lsfMonitor]# monitor/tools/seedb -d
/ic/software/cad_data/it/lsfMonitor/db/monitor/load.db
DB_FILE : /ic/software/cad_data/it/lsfMonitor/db/monitor/load.db
TABLES  :
=====
load_ic-hpc-mon02
load_ic-lsfmaster1
load_ic-lsfmaster2
...
=====
```

示例二，查看 load.db 数据库中指定的表 (load_n212-206-211) 的内容。

```
[root@ic-admin2 lsfMonitor]# monitor/tools/seedb -d
/ic/software/cad_data/it/lsfMonitor/db/monitor/load.db -t
load_n212-206-211
DB_FILE : /ic/software/cad_data/it/lsfMonitor/db/monitor/load.db
TABLE   : load_n212-206-211
=====
sample_second          sample_time          ut
```

tmp	swp	mem
---	---	---
---	---	---
1683984602	20230513_213002	0%
1671G	252.8G	773G
1683984902	20230513_213502	0%
1671G	252.8G	809G
...		
=====		

示例三，查看 load.db 数据库中指定的表 (load_n212-206-211) 的指定列 (mem) 的内容。

```
[root@ic-admin2 lsfMonitor]# monitor/tools/seedb -d
/ic/software/cad_data/it/lsfMonitor/db/monitor/load.db -t
load_n212-206-211 -k mem
DB_FILE : /ic/software/cad_data/it/lsfMonitor/db/monitor/load.db
TABLE   : load_n212-206-211
=====
mem
-----
773G
809G
845G
886G
903G
...
=====
```

示例四，查看 load.db 数据库中指定的表 (load_n212-206-211) 的指定列 (mem) 的内容，只看前三行。

```
[root@ic-admin2 lsfMonitor]# monitor/tools/seedb -d
/ic/software/cad_data/it/lsfMonitor/db/monitor/load.db -t
load_n212-206-211 -k mem -n 3
DB_FILE : /ic/software/cad_data/it/lsfMonitor/db/monitor/load.db
TABLE   : load_n212-206-211
=====
mem
-----
773G
```

```
809G  
845G  
=====
```

5.4 memPrediction

memPrediction 是 lsfMonitor 内置的一个 LSF 任务 memory 资源分析和自动预设工具，可以用来分析 LSF cluster 中用户 memory/cpu 使用状况，也可以通过 LSF esub 机制自动为未预设内存的 job 自动设置 rusage mem 的值。

lsfMonitor 安装时会自动安装 memPrediction，但是不会配置和启用，如果需要启用，具体使用方法请参照 lsfMonitor 安装目录下的 memPrediction 工具用户文档 [docs/memPrediction_user_manual.pdf](#)。

六、常见问题及解决

6.1 图形显示问题

问题描述：

安装后 bmonitor 不显示图形界面，或者图形界面显示不全、显示效果异常。

问题原因：

- * 使用的 python 版本并非 3.8.8 或者兼容版本。
- * python 库安装不全。

解决方案：

使用推荐的 python3.8.8 或者 anaconda3-2021.05，按照 3.2 章节的方法安装 requirements.txt 的 python 依赖库。

6.2 JOBS 页信息缺失

问题描述：

JOBS 页部分信息缺失。

问题原因：

- * 使用的非兼容版本的 openlava。
- * 使用的版本过老的 LSF（比如 9.1.2 或者更老的版本）。

解决方案：

使用较新版本的 LSF。

6.3 LICENSE 页信息缺失

问题描述：

LICENSE 页不显示有效的 license 信息。

问题原因：

- * bmonitor 启动的 terminal 没有配置环境变量 LM_LIECNSE_FILE。
- * monitor/conf/config.py 中 lmstat_bsub_command 变量配置错误。

解决方案：

* 确认 bmonitor 启动的 terminal 中已经配置好正确的环境变量 LM_LICENSE_FILE。

* 如果当前机器允许执行 EDA 工具（可运行 lmstat），那么将 monitor/conf/config.py 中 lmstat_bsub_command 变量配置为空，否则设置合适的 bsub 命令。

6.4 HOSTS 页和 LOAD 页中的 mem 值为什么不一致

问题描述：

对同一台 server，就当前时刻而言，在 HOSTS 页上看到的 Mem 值和在 LOAD 页中看到的“available mem”值不一致，HOSTS 中看到的值往往偏小。

问题原因：

HOSTS 页中的 Mem 值跟 LOAD 页中的 mem 值信息来源不一样，作用也不一样。

- HOSTS 页，mem 信息来源于“bhosts -l”，显示的是这台机器上实际可以被 reserve 的 mem，已经把被用的和被 reserve 的都排除在外了，所以值会偏小。（用途是判断机器还能否接受 job）
- LOAD 页，mem 的信息来源是“lsload”，显示的是这台机器上真实的剩余 mem 信息，不考虑 rusage 的 mem。（用途是判断机器真实的 mem 用量，判断是否会发生 OOM）

七、技术支持

本工具为开源工具，由开源社区维护，可以提供如下类型的技术支持：

- 部署和使用技术指导。
- 接收 bug 反馈并修复。
- 接收功能修改建议。(需审核和排期)

获取技术支持的方式包括：

- 通过 Contact 邮箱联系开发者。
- 添加作者微信 “liyanqing_1987”，注明“真实姓名/公司/lstMonitor”，由作者拉入技术支持群。



附录

附 1. 变更历史

日期	版本	变更描述	备注
2017	1.0	发布第一个版本 openlavaMonitor, 仅支持 openlava。	
2020	1.1	更名 lsfMonitor, 增加对 LSF 的支持。	
2022	1.2	增加 LICENSE 信息的采集和展示。	
2023.05	1.3	<ul style="list-style-type: none">增加 UTILIZATION 信息的采集和展示。增加 patch 工具。优化数据库数据格式。	数据库格式跟旧版本不兼容, 需重新安装。
2023.06	1.3.1	<ul style="list-style-type: none">优化 utilization 数据采样和取用方式, 采样时汇聚。优化 EDA license 信息采样操作, 通过 multi-process 方式并行大幅缩短采样时间。	
2023.06	1.3.2	<ul style="list-style-type: none">HOSTS 页增加按照 Status/MAX/MaxMem/Host 选择 hosts 的过滤功能。LICENSE 页增加按照 User 选择 license feature 的过滤功能。	
2023.09	1.3.3	<ul style="list-style-type: none">show_license_feature_usage 页面显示 feature 跟 job 的关联关系。增加工具 akill, 帮助更快捷地 kill job。Check 按钮自动触发 LSF/LICENSE 信息的重新采集, 保证刷新功能。	
2023.11	1.4	<ul style="list-style-type: none">所有的单选下拉框均改为复选下拉框, 支	queue.db 数据格式跟旧版本不

		<p>持多选。</p> <ul style="list-style-type: none"> • QUEUES 及 UTILIZATION 页面，支持选择更细力度的数据曲线。 • QUEUES 页面支持选择数据曲线日期范围。 • UTILIZATION 页面增加 queues 的 utilization 统计汇总信息。 	兼容，需要删掉旧的 db 文件。
2023.12	1.4.1	<ul style="list-style-type: none"> • 增加工具 logo 和菜单栏项目图标。 • 支持所有的表格导出。 • 曲线图显示效果优化。 	
2024.03	1.4.2	<ul style="list-style-type: none"> • 增加对多 LSF/openlava cluster 的支持。 	db_path 会自动做一次切换，无需人工干预。
2024.06	1.5	<ul style="list-style-type: none"> • 增加图形界面尺寸自适应。 • 增加文本自动补全功能。 • 在 JOB 和 JOBS 页增加 kill job 功能。 • 在 JOBS 页和 HOSTS 页增加右键菜单。 • 在 Check Fail Reason 功能中增加退出码和退出原因的明确对应。 • 将表格输出格式由 Excel 切换为 csv 文件。(避免 Excel 65536 行的限制) • 增加任务内存分析和预设工具 memPrediction。 	
2024.08	1.5.1	<ul style="list-style-type: none"> • bsample 增加 DONE/EXIT job 信息采样功能，保存为 json 文件。 • memPrediction 增加 bsample 采样 json 作为 training 数据源。 • 修复 job 信息采样中 MEM 信息获取的 bug。 	db_path 中的 job 目录切换为 job_mem 目录，job 目录改存 job 历史信息。

2024.09	1.6	<ul style="list-style-type: none">bsample 中修改 user 采样信息内容。bmonitor 新增 USER 页，展示用户 job 的汇总信息。bmonitor 增加 dark_mode，面板可以选择亮色系或者暗色系。支持 ByteDance 开源的 LSF 兼容调度器 volclava。	db_path 中新增 user 目录。
2025.03	1.7	<ul style="list-style-type: none">bsample 的 job 信息保存介质，从 json 转为 sqlite3 数据库，历史 job 检索速度加快数十倍。	

备注：小的 hotfix 不计入变更历史，bugfix 会实时 checkin 到 github 上。