

Linux サーバ版 GAMESS インストールマニュアル

2025 年 6 月 15 日

本マニュアルの目的

本マニュアルでは、単独ユーザが独占的に Linux サーバ（Rocky Linux 9.4）を使用して GAMESS 2024R2 のジョブを並列実行するための環境構築方法と Winmostar のリモートジョブ投入機能から Linux サーバへジョブを投入する方法を示しています。計算環境は全てユーザのホームディレクトリ配下で行うことを想定しています。複数ユーザが使用する共用サーバの環境を構築する方法、複数ノードを利用する環境を構築する方法などは本マニュアルに含まれませんので、ハードウェアベンダにお問い合わせください。

また、本マニュアルではサーバホスト名は「remote_server」、ユーザ名は「winmostar_user」であると仮定します。

1. コンパイラの確認・インストール

- ① Linux サーバにログインし、以下のコマンドで make, cmake がインストールされているか確認する。

- ② `$ which make`

- ③ `$ which cmake`

インストールされていない場合は dnf などのパッケージ管理ソフトウェアを用いてインストールする。Rocky 9.4 で dnf を使う場合は以下のように実行する。

- ④ `$ sudo dnf install make`

- ⑤ `$ sudo dnf install cmake`

- ⑥ 以下のコマンドで gfortran、g++ がインストールされているか確認する。

```
$ which gfortran
```

```
$ which g++
```

インストールされていない場合は dnf などのパッケージ管理ソフトウェアを用いてインストールする。Rocky 9.4 で dnf を使う場合は以下のように実行する。

```
$ sudo dnf install gcc
```

```
$ sudo dnf install gcc-c++
```

```
$ sudo dnf install gcc-gfortran
```

- ⑦ 以下のコマンドで tcsh がインストールされているか確認する。

```
$ which tcsh
```

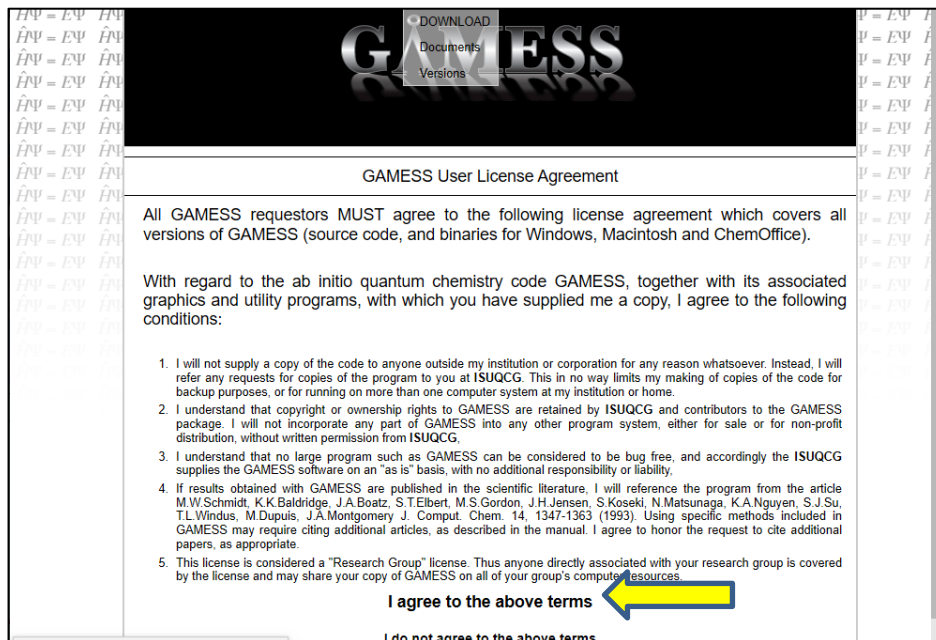
インストールされていない場合は dnf などのパッケージ管理ソフトウェアを用いてインストールする。Rocky 9.4 で dnf を使う場合は以下のように実行する。

```
$ sudo dnf install tcsh
```

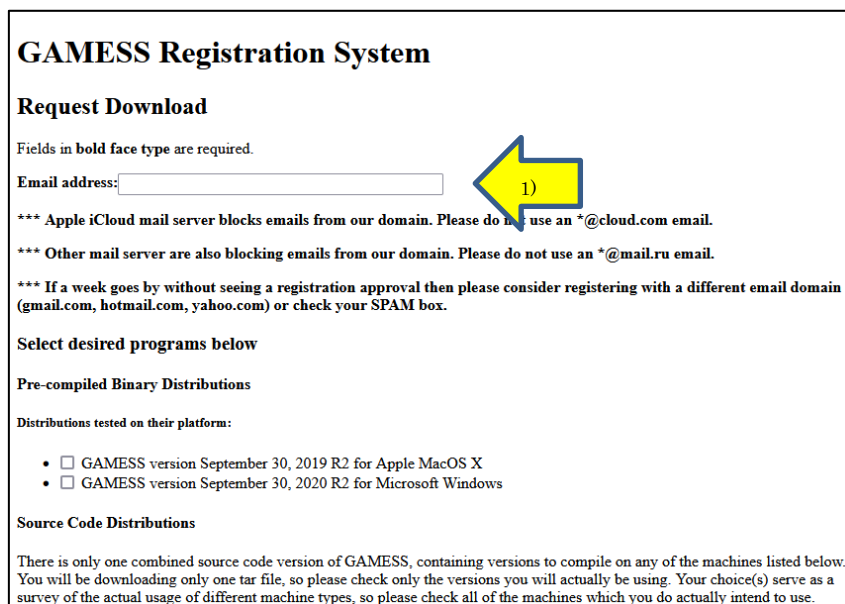
2. GAMESS のインストール

① GAMESS のサイト内の[GAMESS User License Agreement]

[https://www.msg.chem.iastate.edu/gamess/License Agreement.html](https://www.msg.chem.iastate.edu/gamess/License%20Agreement.html) にブラウザを用いてアクセスする。同意できる場合は**"I agree to the above terms"**をクリックする。サーバのメンテナンスにより数日間アクセスできないことがあるので、その場合はアクセスできるようになるまで待つ。



次に、**Email address** にメールアドレスを入力し、**GAMESS version July 15, 2024 R2 for DoD HPCMP TI-25** にチェックを入れ（64 bit Linux の場合）、**Submit Request** をクリックする。



Distributions well-tested on their platform:

- ☒ GAMESS version July 15, 2024 R2 for DoD HPCMP TI-25
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for 64 bit ARM under Linux using armflang compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for 64 bit ARM under Linux using GNU compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for 64 bit ARM under Linux using NVIDIA HPC SDK
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for 64 bit ARM under Linux using unlisted compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for 64 bit x86_64 under Linux using AMD compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for 64 bit x86_64 under Linux using GNU compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for 64 bit x86_64 under Linux using Intel compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for 64 bit x86_64 under Linux using Intel OneAPI compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for 64 bit x86_64 under Linux using NVIDIA HPC SDK
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for 64 bit x86_64 under Linux using PGI compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for 64 bit x86_64 under Linux using unlisted compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for Apple MacOS X using GNU compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for Apple MacOS X using Intel compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for Apple MacOS X using PGI compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for Apple MacOS X using unlisted compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for HPE Cray
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for IBM Power processors under Linux
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for Microsoft Windows using Intel compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for Microsoft Windows using Intel OneAPI compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for Microsoft Windows using PGI compilers
- ☐ GAMESS version July 15, 2024 R2 Patch 1 for Microsoft Windows using unlisted compilers

[Submit Request](#)

初回は登録フォームが表示される。太字の欄をすべて入力し、再度 **Submit Request** をクリックする。もしエラーが出て登録フォームが表示されない場合、他のブラウザで試してください。

Microsoft Edge の場合は、Internet Explorer モードでアクセスし直してください。

GAMESS Registration System

User Registration Required

It seems that this is your first time downloading GAMESS software. Please fill in the following form so we have an idea of who's using GAMESS.

Fields in **bold face type** are required.

Email address:	<input type="text"/>
Email address again:	<input type="text"/>
First name:	<input type="text"/>
Last name:	<input type="text"/>
Organization:	<input type="text"/>
Country:	<input type="text" value="Select a Country"/>
Use for GAMESS:	<input type="text" value="Choose a usage type"/>

Select desired programs below

以下のような受付完了の画面が表示される。

GAMESS Registration System

Request Accepted

Your download request has been accepted. E-mail has been sent to your address containing instructions for downloading the requested software. If these email(s) do not arrive in the next few days, contact gamess@si.msg.chem.iastate.edu Note that the email will have a from address of gamess@si.msg.chem.iastate.edu and a reply to address of gamess@si.msg.chem.iastate.edu. If you do not receive the email reply within the next hour please check your spam folder or otherwise verify that the email reply has not been filtered by your email software.

- ② 以下のようなメールが届くのを待つ（数日程度掛かる可能性がある）。届いたら、メールに記載されているパスワードをメモした上でダウンロードリンクをクリックする。

You can download the GAMESS source with your browser by going to:
<http://www.msg.chem.iastate.edu/GAMESS/download/dist.source.shtml>
Instructions are provided on that web page for downloading and unpacking the archive.

ダウンロードリンク

The username is:

source

You will need the password listed below:

Please note that the password will be changed regularly (usually weekly on Sunday). You should therefore arrange to complete the downloading of the source in a timely manner, before the password is changed. The password this week is:

パスワード

リンク先で `games-current.tar.gz` をクリックする。ユーザ名とパスワードの入力を要求されるので、以下のように入力する。

ユーザ名: source

パスワード: (メールに記述されたもの)

You should have already filled out the GAMESS user registration and received an email reply with the current password for downloading the source code. If you have not received the current password click [here](#) to begin the registration process.

Downloading the latest version of GAMESS:

You can download the GAMESS source with your browser by clicking on the link below. Once you click on the link you will be prompted for the username and password, the username is always:

source

and the password is in the email reply you have received. Note that the password is changed weekly.

The download is for the current GAMESS release: [games-current.tar.gz](#)
MD5 (games-current.tar.gz) = 4490b5a5969dc05c470b4a2e756253b3

If your browser attempts to view this file instead of saving to disk, try holding down the Shift key when you click the link.

If you have problems with your download, please make sure that you have noticed that the user name is always 'source', and that the case-sensitive password changes every week. Most download problems are a result of one or both of these. In case you are trying to use an expired password, please repeat your registration to learn the current password, by clicking [here](#).

DoD HPCMP TI-23: Sept 30, 2021 R2 Public Release:

The download is for the Sept 30, 2021 R2 GAMESS release: [games.Sep302021R2.tar.gz](#)
MD5 (games.Sep302021R2.tar.gz) = 4b5bf661e5efc5c7d79cd6dd8541598f

- ⑧ `games.Jul152024R2.tar.gz` を SCP, FTP など Linux サーバに転送する。ここではホームディレクトリ直下に置いたと仮定する。
- ⑨ Linux サーバにログインし、以下のコマンドで圧縮ファイルを解凍する。
- ```
$ tar xzvf games.Jul152024R2.tar.gz
```
- ⑩ 使用するコンパイラのバージョンを調べてメモする。本書の手順に従いコンパイラに `gfortran` を利用する場合は以下のコマンドを実行する。
- ```
$ gfortran -v
```

⑪ 以下のコマンドで GAMESS のビルド設定を開始する。

```
$ cd ~/gamess
```

```
$ ./config
```

以下の画面が表示されたら[Return]キーを押す。

```
[winmostar_user@remote_server]$ ./config
This script asks a few questions, depending on your computer system,
to set up compiler names, libraries, message passing libraries,
and so forth.
You can quit at any time by pressing control-C, and then <return>.
```

Please open a second window by logging into your target machine, in case this script asks you to 'type' a command to learn something about your system software situation. All such extra questions will use the word 'type' to indicate it is a command for the other window.

After the new window is open, please hit <return> to go on.

以下の画面が表示されたら「linux64」と入力する。

```
GAMESS can compile on the following 32 bit or 64 bit machines:
hpe-cray-ex  - HPE Cray's EX architecture (e.g., narwhal, spock)
hpe-cray-cs  - HPE Cray's CS architecture (e.g., tulip)
cray-xt      - Cray's XT massively parallel system, running CNL
cray-xc      - Cray's XC massively parallel system (e.g., DoE Theta, DoD Onyx)
fj-a64fx     - Fujitsu PRIMEHPC with A64FX cpus and Fujitsu Compilers (e.g., FX1000, FX700)
ibm64        - IBM, Power8 chip or newer, running AIX or Linux (e.g., DoE Summit, DoD Hokulea)
linux32      - Linux (any 32 bit distribution), for x86
linux64      - Linux (any 64 bit distribution), for x86_64, ia64, or arm64 chips,
                using gfortran, ifort, or perhaps PGI compilers.
mac64        - Apple Mac, any chip, running OS X 10.5 or newer
win64        - Windows 64-bit (Windows 10)
singularity  - GAMESS Singularity container image
```

type 'uname -a' to partially clarify your computer's flavor.

please enter your target machine name: linux64

以下の画面が表示されたら[Return]キーを押す。

```
Where is the GAMESS software on your system?
A typical response might be /u1/mike/gamess,
most probably the correct answer is /home/ winmostar_user /gamess
```

```
GAMESS directory? [/home/winmostar_user/gamess]
```

以下の画面が表示されたら[Return]キーを押す。

```
Setting up GAMESS compile and link for GMS_TARGET=linux64
GAMESS software is located at GMS_PATH=/home/winmostar_user/gamess
```

Please provide the name of the build location.
This may be the same location as the GAMESS directory.

```
GAMESS build directory? [/home/winmostar_user/games]
```

以下の画面が表示されたら[Return]キーを押す。

Please provide a version number for the GAMESS executable.
This will be used as the middle part of the binary's name,
for example: gamess.00.x

Version? [00]

以下の画面が表示されたら[Return]キーを押す。

Are you building for a specific HPC system?

Options:

nautilus - DoD Penguin system at Navy DSCR w/ AMD EPYC (Milan), NVIDIA
A100 GPGPUs, and HDR-200 Infiniband
grace-hopper - NVIDIA Grace-Hopper Superchip System
<ENTER> - none

HPC system target for 64-bit Linux system:

以下の画面が表示されたら「gfortran」と入力する。

Linux FORTRAN compiler selection

=====

aocc	- AMD Fortran compiler	
armflag	- ARM Fortran compiler	
gfortran	- GNU Fortran compiler	(minimum support v. GCC-5)
ifort	- Intel Fortran compiler	(minimum support v. 2018.3)
oneapi-ifort	- Intel OneAPI Classic Fortran compiler	(minimum support v. 2021.3.0)
oneapi-ixf	- Intel OneAPI Beta Fortran compiler	(minimum support v. 2021.3.0)
pgfortran	- NVIDIA PGI Fortran compiler	(minimum support v. 19.10
Community)		
nvfortran	- NVIDIA HPC SDK Fortran compiler	(minimum support v. 21.7-0)

Please enter your choice of FORTRAN:: [gfortran](#)

以下の画面が表示されたら 2. ⑤で調べたコンパイラのバージョンを入力する (gfortran の場合)。エラーが表示される場合は少し古いバージョンを指定する。

gfortran is very robust, so this is a wise choice.

Please type 'gfortran -dumpversion' or else 'gfortran -v' to
detect the version number of your gfortran.
This reply should be a string with at least two decimal points,
such as 4.9.4 or 6.3.0.

The reply may be labeled as a 'gcc' version,
but it is really your gfortran version.

Please enter only the first decimal place, such as 8.2 or 11.2 [11.2](#)

以下の画面が表示されたら[Return]キーを押す。

Please enter only the first decimal place, such as 8.2 or 11.2: 11.2

Good, the newest gfortran version number 11.2 can compile REAL*16 data type.
Please report any numerical issues you encounter.

Hit <ENTER> to continue to the math library setup.

以下の画面が表示されたら「none」と入力する¹。その後、[Return]キーを押す。

```
Linux distributions do not include a standard math library.

There are several reasonable add-on library choices,
    MKL from Intel           for 32 or 64 bit Linux (very fast)
    ACML from AMD           for 32 or 64 bit Linux (free)
    LibFLAME from AMD       for 64 bit Linux (free)
    ATLAS from www.rpmfind.net for 32 or 64 bit Linux (free)
    PGI BLAS from Portland Group for 32 or 64 bit Linux
    ArmPL from ARM          for 64 bit Linux
and one very unreasonable option, namely 'none', which will use
some slow FORTRAN routines supplied with GAMESS.  Choosing 'none'
will run MP2 jobs 2x slower, or CCSD(T) jobs 5x slower.

Some typical places (but not the only ones) to find math libraries are
Type 'echo $MKLROOT'           to look for MKL
Type 'ls -d /opt/acml*'        to look for ACML
Type 'ls -d /usr/local/acml*'  to look for ACML
Type 'ls /usr/lib64/atlas'     to look for Atlas
Type 'ls /opt/pgi/linux86-64/*lib/*' to look for libblas.a from PGI
Type 'ls /opt/pgi/osx86-64/*lib/*'  to look for libblas.a from PGI
Type 'echo $ARMPL_DIR'        to look for ArmPL

Enter your math library choice from one of the options below:

'acml', 'atlas', 'libflame', 'mkl', 'openblas', 'pgiblas', 'armpl', 'none'

: none
```

以下の画面が表示されたら[Return]キーを押す。

```
GAMESS internal BLAS routines in blas.src has been replaced
with Netlib BLAS and LAPACK libraries.

The Netlib BLAS and LAPACK libraries will now be downloaded.

Please run 'config' first, to set up GAMESS compiling information

The download failed. You can try again later by calling
the following script from inside the GAMESS directory:

tools/lapack/download-lapack.csh

The Netlib BLAS and LAPACK libraries will be compiled by running:

make -j lapack

CMake is the preferred approach to building the Netlib BLAS and
LAPACK libraries.

Alternatelively you can build these libraries without CMake by
following the instructions available at:

/home/winmostar_user/gamess/tools/lapack/README.md
-----

Hit <ENTER> after you have read this message.
```

以下の画面が表示されたら「sockets」と入力する。

¹ GAMESS2022 バージョンでは math library に none を選択すると cmake 3.2 以降が必要となります。

If you have a slow network, like Gigabit Ethernet (GE), or
if you have so few nodes you won't run extensively in parallel, or
if you have no MPI library installed, or
if you want a fail-safe compile/link and easy execution,
 choose 'sockets'
to use good old reliable standard TCP/IP networking.

If you have an expensive but fast network like Infiniband (IB), and
if you have an MPI library correctly installed,
 choose 'mpi'.

If you wish to use a combination of TCP/IP networking for small
messages and MPI for large messages in a 'mixed' fashion,
 choose 'mixed'.

communication library ('serial','sockets' or 'mpi' or 'mixed')? [sockets](#)

以後、表示される各種の質問について全て「no」と入力する。(LibXC から RISM-SCF-cSED
まで) そして、下記メッセージが出力されればビルドの設定は完了し `install.info` に情報
が書き込まれる。

Your configuration for GAMESS compilation is now in
 /home/winmostar_user/games/install.info

- ⑫ 以下のコマンドを実行し **LAPACK** をビルドする。

```
$ ./tools/lapack/download-lapack.csh
```

```
$ make lapack
```

- ⑬ 以下のコマンドを実行し **GAMESS** をビルドする。make が正常終了すると、実行ファイルの `gamess.00.x` が生成される。生成されていない場合は `make.log` を確認しエラーの内容を確認する。

```
$ make 2>&1 | tee make.log
```

- ⑭ `gamess` ディレクトリ以下の `rungms` の内容を以下のように修正する。**GAMESS** をビルドしたディレクトリの名前を `gamess` から変更した場合は `rungms` 内の `set GMSPATH` の行の値も追加で修正する（本書の手順に従った場合、`GMSPATH` の変更は不要）。

```
set SCR=~ /gamess/restart
set USERSCR=~ /gamess/restart
```

↓

```
set SCR=$HOME/scr
set USERSCR=$HOME/scr
```

- ⑮ 以下のコマンドを実行しユーザのホームディレクトリに `scr` ディレクトリを作成する。

```
$ mkdir $HOME/scr
```

- ⑯ 動作確認のため以下のコマンドでテスト計算を実行する。

- ⑰ `$ rm ~/scr/*`

```
$ ./runall 00
```

終了後、テスト結果を確認するために次のコマンドを実行する。

```
$ ./tests/standard/checktst
```

最後に "All 49 test results are correct!" と表示されていれば全て正常に計算されたことを意味する。

Checking the results of your sample GAMESS calculations,
the output files (exam??.log) will be taken from .

All jobs terminated normally, now checking detailed numerical results

exam01: Eerr=0.0e+00 Gerr=0.0e+00.	Passed.
exam02: Eerr=0.0e+00 Gerr=0.0e+00 Serr=0.0e+00 Lerr=1.8e-03+6.6e-05.	Passed.
(省略)	
exam47: Serr=0.0e+00 Perr=6.0e-10.	Passed.
exam48: E0err=0.0e+00 E1err=0.0e+00 Gerr=0.0e+00.	Passed.
exam49: Eerr=0.0e+00 Rerr=4.0e-10 Terr=2.5e-07.	Passed.

All 48 test results are correct!

テストに失敗して exam01.log に

```
DDI Process 0: shmget returned an error.
```

```
Error EINVAL: Attempting to create 160525768 bytes of shared memory.
```

```
Check system limits on the size of SysV shared memory segments.
```

と書かれている場合は、sudo で/etc/sysctl.conf に

```
kernel.shmmax = (物理メモリサイズを byte 単位で記入)
```

```
kernel.shmall = (kernel.shmmax÷4096 の値を記入)
```

と追記または変更し `$ sudo sysctl -p` と実行する。

3. Linux サーバ上での最終確認

- ① Winmostar から接続するときの状況を再現するため、Linux サーバにログインしなおす。
- ② GAMESS の動作を確認するため、GAMESS の起動に必要な環境を設定するコマンドを入力しメモに控えておく。本書の手順に従った場合は、以下のコマンドを入力する。

```
$ export PATH=$HOME/games:$PATH
```

- ③ 以下のコマンドを実行してインストールした GAMESS のパスが表示されることを確認する。確認できない場合は②の設定を見直す。

```
$ which runrms
```

4. Winmostar からの設定および動作確認

ユーザマニュアルの [7.2. リモートジョブの設定手順](#)に従って設定し動作確認を行う。

テンプレートスクリプトを編集する際には、「# Insert commands here」 から「# Do not modify the followings」の間に 3-②でメモに控えた内容を追記する。本書の手順に従った場合は以下ようになる。

```
echo "*****"
echo "****      Set user-defined variables      ****"
echo "*****"
set -v
# Insert commands here

export PATH=$HOME/games:$PATH

# Do not modify the followings
```

以上