

# SIESTA Deployment Options

SIESTA School, 2021

*Vladimir Dikan* [vdikan@icmab.es](mailto:vdikan@icmab.es)

[Setting up the working environment](#)[Basics of Siesta](#)

## Intermediate and Advanced Topics

[Molecular Dynamics tutorials](#)[Time-Dependent Density-Functional Theory](#)[Spin-Orbit coupling](#)[Polarization calculations with the Berry-phase approach](#)[DFT+U calculations](#)[Wannier functions](#)[Advanced analysis of the electronic structure](#)[Advanced topics in phonons](#)[Calculation of optical properties](#)[Computing magnetic interaction parameters with TB2J](#)[Simulation of STM images](#)

## Siesta Deployment Options

[1 Ready-to-Use Options](#)[2 Source Code Compilation](#)[3 Scripted Installations](#)[4 Contact Information](#)[How-to guides](#)[Technical reference](#)[Background information](#)

# Siesta Deployment Options

Author: Vladimir Dikan

[Edit on GitLab](#)

- Ready-to-Use Options
- Source Code Compilation
- Scripted Installations

<https://docs.siesta-project.org>

As you might know (or will realize during the school), Siesta has quite a number of capabilities and operation modes. And as many other HPC codes, it relies on quite a number of dependencies and build options, that sometimes give their users hard times with configuration of their research environments.

Below is a review of some aspects of compilation and deployment of Siesta. Some pre-configured options are discussed, followed by an overview of Siesta's general Makefile template and

# Ready-to-Use Options



21.06.04

 Search the docs ...

## Quantum Mobile

[Releases](#)

## USERS

[Launching Quantum Mobile](#)[Using Quantum Mobile](#)[VirtualBox FAQ](#)[Troubleshooting](#)

## DEVELOPERS

[Customise Quantum Mobile](#)[Build a Desktop VM](#)[Build a Cloud VM](#)[Build a Docker container](#)[Create a new ansible role](#)

## MAINTAINERS

[Developing Quantum Mobile](#)[Preparing releases](#)

# Quantum Mobile

## What is Quantum Mobile

Quantum Mobile is a Virtual Machine for computational materials science.

Quantum Mobile provides a uniform environment for quantum mechanical materials simulations. Simulation codes are set up and ready to be used either directly or through the AiiDA python framework for automated workflows and provenance tracking.

# QuantumMobile VM

Open source throughout

Based on Ubuntu Linux

Pre-built images

Available for Linux, MacOS or Windows computers, using VirtualBox. Or deploy on cloud services like OpenStack or Amazon Elastic Compute Cloud using [ansible](#).

### Simulation codes pre-installed

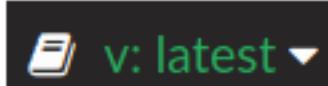
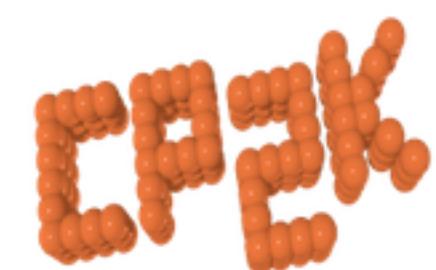
[Abinit](#), [BigDFT](#), [CP2K](#), [Fleur](#), [Quantum ESPRESSO](#), [Siesta](#), [Wannier90](#), [Yambo](#), together with [AiiDA](#), [JupyterLab](#), and the [AiiDALab](#) Jupyter environment.

### Tools pre-installed

atomistic (xcrysden, jmol, cif2cell, ase, pymatgen, seekpath, spglib, pycifrw), visualization (grace, gnuplot, matplotlib, bokeh, jupyter), simulation environment (slurm, OpenMPI, FFT/BLAS/LAPACK, gcc, gfortran, singularity).

### Modular setup

with individually tested [ansible roles](#). Build your own flavour tailored to your use case.



TAG

**master**

Last pushed a month ago by vdikan

DIGEST

[eee62cd76e63](#)

OS/ARCH

linux/amd64

COMPRESSED SIZE ⓘ

185.66 MB

[docker pull vdikan/siesta-dist:master](#) 

TAG

**psml**

Last pushed a month ago by vdikan

DIGEST

[a7e09b3984f7](#)

OS/ARCH

linux/amd64

[docker pull vdikan/siesta-dist:psml](#) 

COMPRESSED SIZE ⓘ

186.83 MB

# Containers: Docker

<https://hub.docker.com/>

TAG

**elsi**

Last pushed a month ago by vdikan

DIGEST

[6dcbf229a52b](#)

OS/ARCH

linux/amd64

[docker pull vdikan/siesta-dist:elsi](#) 

COMPRESSED SIZE ⓘ

193.45 MB

```
docker pull vdikan/siesta-dist:master
docker run --interactive --tty -w /app \
-v "$(pwd):/app" vdikan/siesta-dist:master
```

TAG

**4.1-b4**

Last pushed a month ago by vdikan

DIGEST

[b2b17b466a1d](#)

OS/ARCH

linux/amd64

COMPRESSED SIZE ⓘ

170.26 MB

[docker pull vdikan/siesta-dist:4.1-b4](#) 



# Cloud Library

Cloud Library is the official image registry provided by [Sylabs.io](#). Users can share Singularity images through the Cloud Library, as well as pull/push SIF™ images through Singularity CLI. Email any feature requests or feedback to [support@sylabs.io](mailto:support@sylabs.io).

## Containers: Singularity

<https://cloud.sylabs.io/library>

From an existing Docker image:

```
singularity build siesta-dist.sif \
    docker://vdikan/siesta-dist:master
```

```
singularity run siesta-dist.sif
```

> You created 0 Remote Builds

> You created 1 Keys

Signed With: 2568c99d884fe031abefe05b777b95800adee8b6

### MOST DOWNLOADED

library/default/alpine	430411
sylabs-bot/smoke-test-collection	221028
/testimage0	
sylabs-bot/smoke-test-collection	221003
/testimage1	
library/default/busybox	212710
sylabs/tests/not-default	106845
sylabs/tests/unsigned	63408
library/default/ubuntu	26092
sylabs/tests/signed	22842
sylabsed/examples/lolcow	12679

### library/default/centos

### MOST STARRED

library/default/ubuntu	28
library/default/alpine	19

# Source Code Compilation



siesta

Project information

Repository

Issues

Merge requests

Requirements

CI/CD

Deployments

Environments

Releases

Monitor

Packages &amp; Registries

Analytics

Wiki

Snippets

siesta-project &gt; siesta &gt; Releases

# Staging the build directory

<https://gitlab.com/siesta-project/siesta/-/releases/v4.1.5>

```
# 1. obtain src archive
wget https://gitlab.com/siesta-project/siesta/-/archive/v4.1.5/siesta-v4.1.5.tar.gz
tar -xzf siesta-v4.1.5.tar.gz
cd siesta-v4.1.5/
# 2. create distinct build directory for siesta
cd Obj/
sh ./Src/obj_setup.sh
# 3. bootstrap arch.make inside build directory
cp ARCH-EXPERIMENTAL/master-raw.make ./arch.make
$EDITOR arch.make # configure siesta build
# 4. compile siesta
make
# 5. optionally, build the utilities
cd Util && ./build_all.sh
```

This is a production release of the 4.1 series.

See the [Guide to Siesta versions](#) for more information.

Please read the [Release Notes](#) shipped with the sources. They may also be found [here](#).

```
# Make sure you have the appropriate library symbols
# (Either explicitly here, or through shell variables, perhaps
# set by a module system)
# Define also compiler names and flags
#-----
#XMLF90_ROOT=$(shell spack location -i xmlf90)
#PSML_ROOT=$(shell spack location -i libpsml)
#GRIDXC_ROOT=$(shell spack location -i libgridxc@thtr)
#LIBXC_ROOT=$(shell spack location -i libxc)
#ELSI_ROOT=
#ELPA_ROOT=
#ELPA_INCLUDE_DIRECTORY=
#FLOOK_ROOT=
#-----
#NETCDF_ROOT=$(shell spack location -i netcdf-c)
#NETCDF_FORTRAN_ROOT=$(shell spack location -i netcdf-fortran)
#HDF5_LIBS=-L/apps/HDF5/1.8.20/GCC/OPENMPI/lib -lhdff5 -lbdf5 -lcurl -lz
#SCALAPACK_LIBS=-L$(shell spack location -i netlib-scalapack)/lib -lscalapack
#LAPACK_LIBS = -L$(shell spack location -i openblas)/lib -lopenblas
#FFTW_ROOT=/apps/FFTW/3.3.7-GCC-OpenMPI
# Needed for PEXSI (ELSI) support
#LIBS_CPLUS=-stdc++ -lstdc++ -lstdc++abi
#-----
# FC_PARALLEL=$(shell spack location -i OpenMPI)/lib/mpi.f90
#
#FC_SERIAL=gfortran
#FPP = $(FC_SERIAL) -E -P -x c
#FFLAGS = -g -O2 -fPIC -ftree-vectorize
#FFLAGS_DEBUG= -g -O0 -fPIC -ftree-vectorize
#RANLIB=echo
#
# Alternatively, prepare a fortran.mk file with compiler definitions,
```

# arch.make configuration

## Inside the Obj/arch.make file:

- define a set of external dependencies
- provide linking paths and symbols for dependencies
- configure compilers



siesta-project &gt; Libraries

# Siesta Dependencies

<https://gitlab.com/siesta-project/libraries>

Libraries originating in the Siesta project.

## Common

MPI, BLAS, LAPACK

## SIESTA-specific

libfdf

ELSI + ext. solvers

xmlf90

NetCDF

libPSML

libXC

libGridXC

flook

flos

libfdf 11 months ago

xmlf90 1 year ago

libPSML 1 year ago

libGridXC 1 year ago

flos 2 weeks ago

xmlf90 11 months ago

# Scripted Installations



S

▶

▶

▶

▶

▶

▶

▶

▶

▶

▶

▶

▶

Xu He &gt; siesta-install-scripts &gt; Repository

ubuntu

siesta-install-scripts

History

Web IDE

Download

Clone ▾

# Siesta-Install-Scripts

[https://gitlab.com/mailhexu/siesta-install-scripts/-/tree  
/ubuntu](https://gitlab.com/mailhexu/siesta-install-scripts/-/tree/ubuntu)

```
git clone https://gitlab.com/mailhexu/siesta-install-scripts.git
cd siesta-install-scripts/
# 1. address README-s in subdirectories
# 2. inspect ./Tarballs/download.sh
./Tarballs/download.sh
cp Config/gnu/* Tarballs/ # copy Makefile for GNU
cp Config/siesta.common.arch.make Tarballs/
# 3. inspect and adapt Makefiles
# 4. inspect and edit ./do_all.sh
./do_all.sh
```

README

Collection of scripts to build Siesta (MaX version 1)



# Spack

A flexible package manager supporting various compilers, platforms, and compilers.

[GitHub](#)[Twitter](#)[Slack](#)[Docs](#)[Discussion](#)

## Obtain Spack with SIESTA suit packages:

```
git clone -b siesta-develop https://github.com/vdikan/spack.git
```

After configuration, install SIESTA with (in principle) one-line spec command:

```
spack install siesta@master+utils^openmpi +cxx +cxx_exceptions
```

Installing scientific software easy. Spack isn't tied to a particular language; you can build a software stack in [Python](#) or [R](#), link to libraries written in C, C++, or

Fortran, and easily [swap compilers](#) or target [specific microarchitectures](#). Learn more [here](#).

Recent Posts

**Intermediate and Advanced Topics**

Molecular Dynamics tutorials

Time-Dependent Density-Functional Theory

Spin-Orbit coupling

Polarization calculations with the Berry-phase approach

DFT+U calculations

Wannier functions

Advanced analysis of the electronic structure

Advanced topics in phonons

Calculation of optical properties

Computing magnetic interaction parameters with TB2J

Simulation of STM images

**Siesta Deployment Options**

1 Ready-to-Use Options

2 Source Code Compilation

3 Scripted Installations

4 Contact Information

**How-to guides**

Technical reference

Background information

# Thank You!

Author: Vladimir Dikan

## Table of Contents

- SIESTA Deployment Options

- <https://docs.siesta-project.org/projects/siesta/>
- <https://siesta-project.org/siesta/>
- **SIESTA Manual**
- **Matter Modeling StackExchange: siesta**

## Contact Me:

As you might know (or will realize during the school), SIESTA has quite a number of capabilities and operation modes. And as any other LPO codes it relies on quite a number of dependencies and build options, that sometimes give their users hard times with configuration of their research environments.

- **Event Official Channel**
- **Email: [vdikan@icmab.es](mailto:vdikan@icmab.es)**

Below is a review of some aspects of compilation and deployment of SIESTA. Some pre-configured options are discussed, followed by an overview of SIESTA's general Makefile template and