

MATERIALS SCIENCE CODES ON INNOVATIVE HPC ARCHITECTURES

PRACE-MAX TRAINING

CINECA – CASALECCHIO DI RENO (BO, ITALY)

5-7 DECEMBER 2016

TUTORIAL: THE SIESTA CODE

OSWALDO DIÉGUEZ

DEPARTMENT OF MATERIALS SCIENCE AND ENGINEERING

TEL AVIV UNIVERSITY

7/12/2016

SIESTA: A DENSITY-FUNCTIONAL THEORY CODE

MAIN DIFFERENCE BETWEEN SIESTA AND OTHER DFT CODES
(SUCH AS QUANTUM ESPRESSO):

**SIESTA WAS DESIGNED TO SCALE LINEARLY WITH THE NUMBER OF
ATOMS IN THE SIMULATION CELL**

IMPLICATIONS:

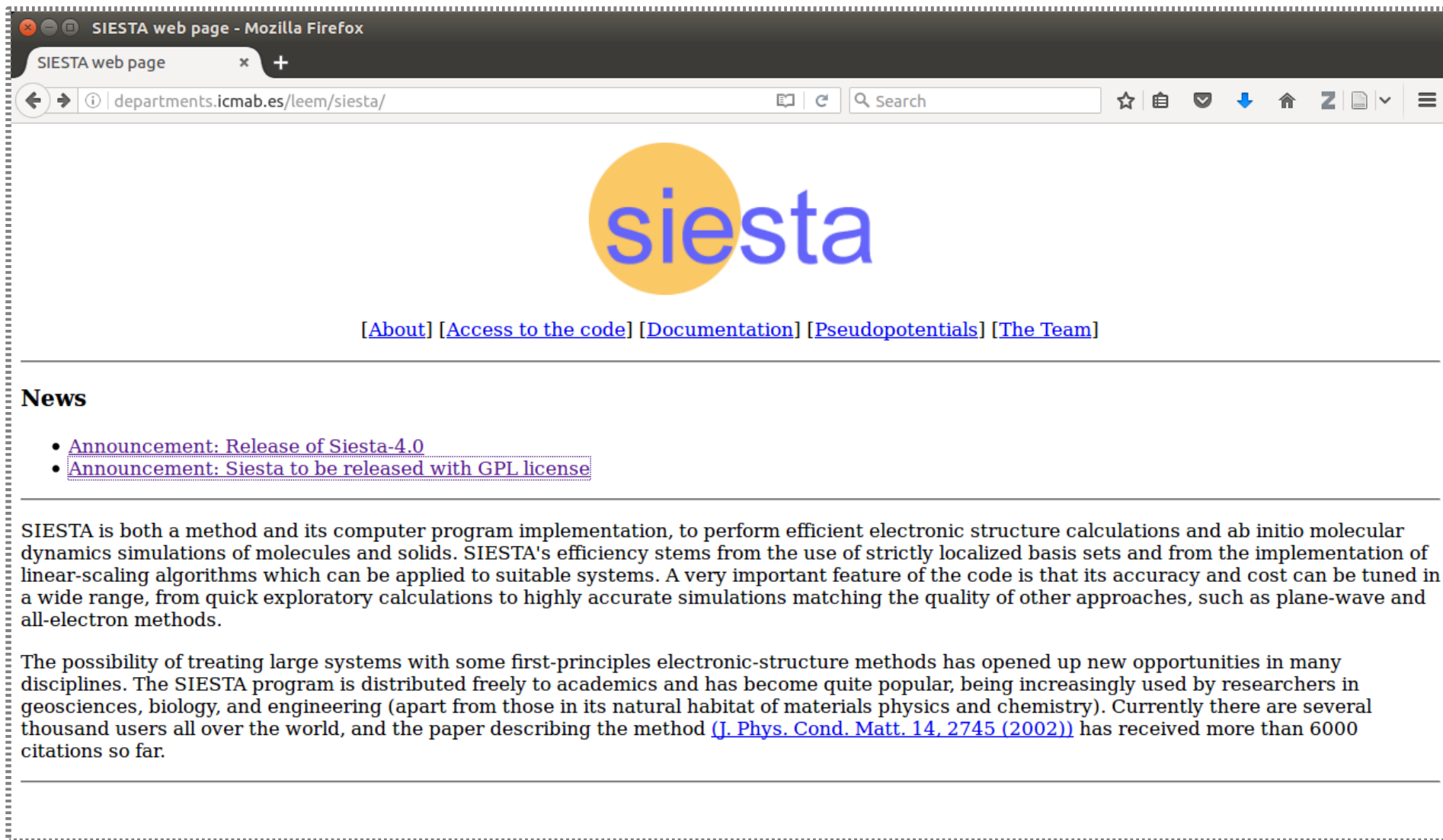
- **SIESTA USES NUMERICAL LOCALIZED ATOMIC ORBITALS AS BASIS SET:**
THIS MEANS THAT THE DESIGN OF THE BASIS SET REQUIRES MORE TESTING
THAN IN THE CASE OF PLANE-WAVE CODES
- ***PERSONAL STATEMENT*:** SIESTA RUNS FASTER THAN ANY PLANE-
WAVE CODE THAT I HAVE USED FOR THE UNITS CELLS THAT I HAVE USED
(BETWEEN 20 AND 120 ATOMS; THE LARGER THE NUMBER OF ATOMS, THE
LARGER THE SPEED DIFFERENCE)

TODAY

- HOW TO GET, INSTALL, AND RUN SIESTA
- TUTORIALS (BY JAVIER JUNQUERA)
- A WORD ABOUT PARALLELISM (BY A NON-EXPERT)

HOW TO GET SIESTA

SIESTA IS NOW UNDER THE GNU GENERAL PUBLIC LICENSE



The screenshot shows a Mozilla Firefox browser window with the address bar displaying "departments.icmab.es/leem/siesta/". The main content area features the SIESTA logo, which consists of a yellow circle partially overlapping the word "siesta" in blue lowercase letters. Below the logo are several blue hyperlinks: [About], [Access to the code], [Documentation], [Pseudopotentials], and [The Team].

News


- [Announcement: Release of Siesta-4.0](#)
- [Announcement: Siesta to be released with GPL license](#)

SIESTA is both a method and its computer program implementation, to perform efficient electronic structure calculations and ab initio molecular dynamics simulations of molecules and solids. SIESTA's efficiency stems from the use of strictly localized basis sets and from the implementation of linear-scaling algorithms which can be applied to suitable systems. A very important feature of the code is that its accuracy and cost can be tuned in a wide range, from quick exploratory calculations to highly accurate simulations matching the quality of other approaches, such as plane-wave and all-electron methods.

The possibility of treating large systems with some first-principles electronic-structure methods has opened up new opportunities in many disciplines. The SIESTA program is distributed freely to academics and has become quite popular, being increasingly used by researchers in geosciences, biology, and engineering (apart from those in its natural habitat of materials physics and chemistry). Currently there are several thousand users all over the world, and the paper describing the method ([J. Phys. Cond. Matt. 14, 2745 \(2002\)](#)) has received more than 6000 citations so far.

HOW TO GET SIESTA

SIESTA IS NOW UNDER THE GNU GENERAL PUBLIC LICENSE




SIESTA web page - Mozilla Firefox

SIESTA web page x +

departmentos.icmab.es/leem/siesta/

Search



[\[About\]](#) [\[Access to the code\]](#) [\[Documentation\]](#) [\[Pseudopotentials\]](#) [\[The Team\]](#)

News

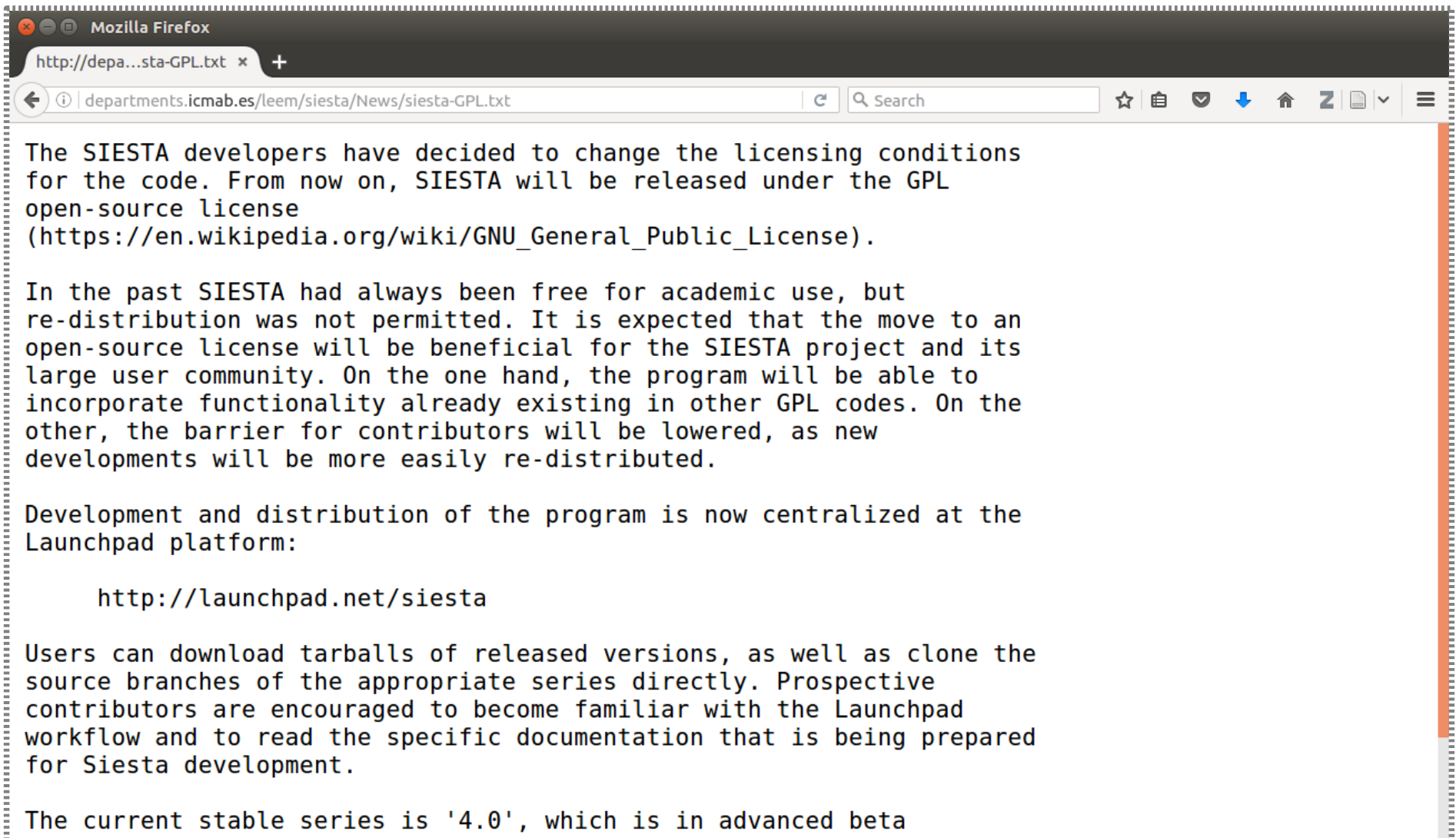
- [Announcement: Release of Siesta-4.0](#)
- [Announcement: Siesta to be released with GPL license](#)

SIESTA is both a method and its computer program implementation, to perform efficient electronic structure calculations and ab initio molecular dynamics simulations of molecules and solids. SIESTA's efficiency stems from the use of strictly localized basis sets and from the implementation of linear-scaling algorithms which can be applied to suitable systems. A very important feature of the code is that its accuracy and cost can be tuned in a wide range, from quick exploratory calculations to highly accurate simulations matching the quality of other approaches, such as plane-wave and all-electron methods.

The possibility of treating large systems with some first-principles electronic-structure methods has opened up new opportunities in many disciplines. The SIESTA program is distributed freely to academics and has become quite popular, being increasingly used by researchers in geosciences, biology, and engineering (apart from those in its natural habitat of materials physics and chemistry). Currently there are several thousand users all over the world, and the paper describing the method ([J. Phys. Cond. Matt. 14, 2745 \(2002\)](#)) has received more than 6000 citations so far.

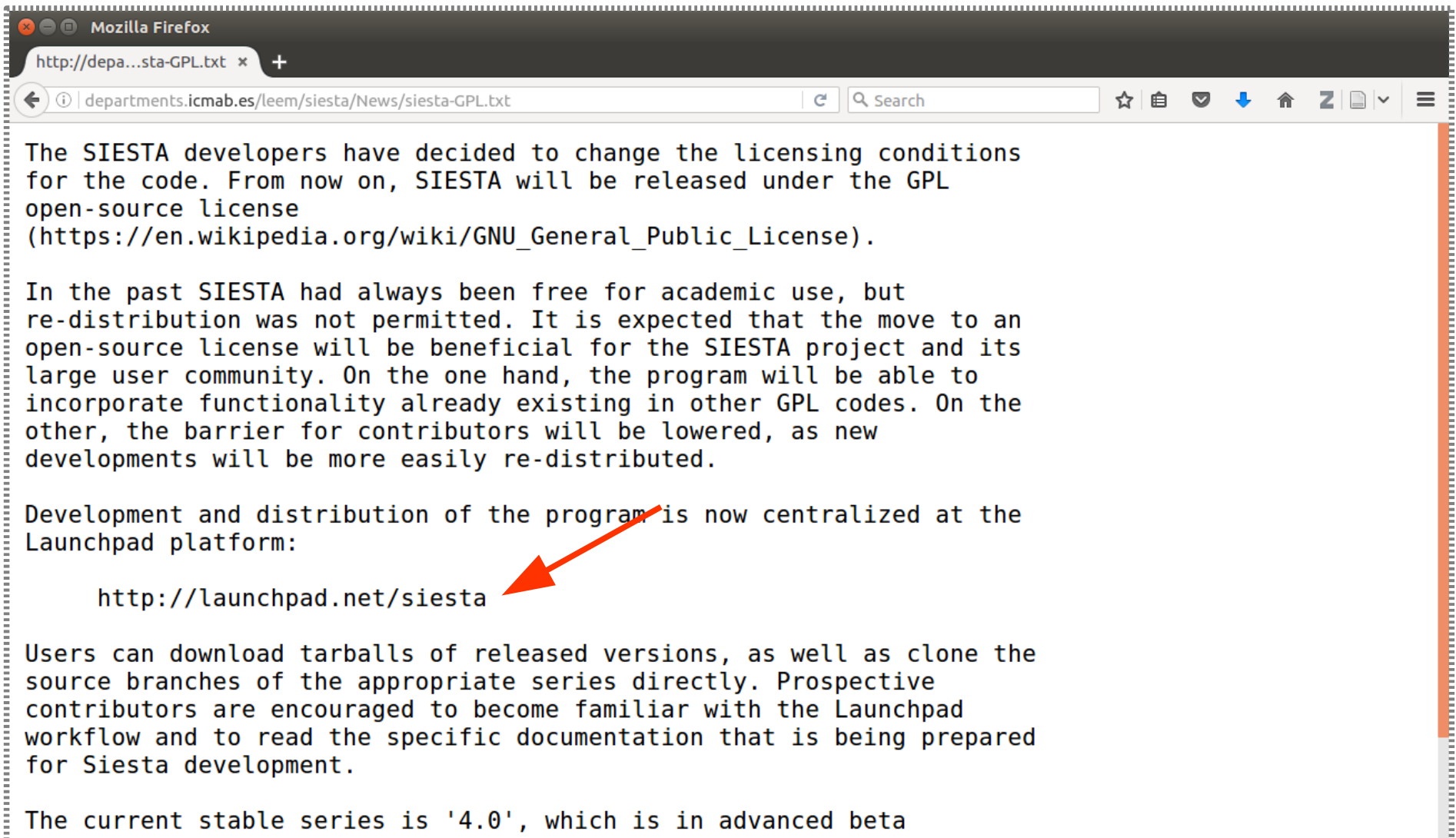
HOW TO GET SIESTA

SIESTA IS NOW UNDER THE GNU GENERAL PUBLIC LICENSE



HOW TO GET SIESTA

SIESTA IS NOW UNDER THE GNU GENERAL PUBLIC LICENSE



The screenshot shows a Mozilla Firefox browser window with the address bar displaying `http://depa...sta-GPL.txt` and `departments.icmab.es/leem/siesta/News/siesta-GPL.txt`. The page content is as follows:

The SIESTA developers have decided to change the licensing conditions for the code. From now on, SIESTA will be released under the GPL open-source license (https://en.wikipedia.org/wiki/GNU_General_Public_License).

In the past SIESTA had always been free for academic use, but re-distribution was not permitted. It is expected that the move to an open-source license will be beneficial for the SIESTA project and its large user community. On the one hand, the program will be able to incorporate functionality already existing in other GPL codes. On the other, the barrier for contributors will be lowered, as new developments will be more easily re-distributed.

Development and distribution of the program is now centralized at the Launchpad platform:

`http://launchpad.net/siesta`

Users can download tarballs of released versions, as well as clone the source branches of the appropriate series directly. Prospective contributors are encouraged to become familiar with the Launchpad workflow and to read the specific documentation that is being prepared for Siesta development.

The current stable series is '4.0', which is in advanced beta

HOW TO GET SIESTA

SIESTA IS NOW UNDER THE GNU GENERAL PUBLIC LICENSE

The screenshot shows a web browser window with the URL <https://launchpad.net/siesta/>. The page features the Siesta logo and navigation tabs for Overview, Code, Bugs, Blueprints, Translations, and Answers. The Overview tab is active, displaying the project's description: "A first-principles materials simulation code using DFT". It also lists the maintainer as Alberto Garcia and provides links for "Get Involved" (Report a bug, Ask a question, Register a blueprint, Help translate) and "Downloads" (Latest version is 4.1-b2, siesta-4.1-b2.tar.gz, siesta.pdf, tbtrans.pdf). The "Project information" section shows the maintainer as Siesta Maintainers and the driver as Alberto Garcia. The "Series and milestones" section shows the current series as "trunk" with a "View full history" link.

Siesta in Launchpad - Mozilla Firefox

Siesta in Launchpad

Canonical Group Ltd (GB) | <https://launchpad.net/siesta/>

Log in / Register

siesta Siesta

Overview Code Bugs Blueprints Translations Answers

Registered 2016-01-21 by [Alberto Garcia](#)

A first-principles materials simulation code using DFT

SIESTA is both a method and its computer program implementation, to perform efficient electronic structure calculations and ab initio molecular dynamics simulations of molecules and solids. SIESTA's efficiency stems from the use of strictly localized basis sets and from the implementation of linear-scaling algorithms which can be applied to suitable systems. A very important feature of the code is that its accuracy and cost can be tuned in a wide range, from quick exploratory calculations to highly accurate simulations matching the quality of other approaches, such as plane-wave and all-electron methods.

Stable series, appropriate for most users: 4.0
Development series, subject to instabilities as new features are added: trunk

[Home page](#)

Project information

Maintainer: [Siesta Maintainers](#) Driver: [Alberto Garcia](#)

Series and milestones

[View full history](#)

trunk

Get Involved

- [Report a bug](#)
- [Ask a question](#)
- [Register a blueprint](#)
- [Help translate](#)

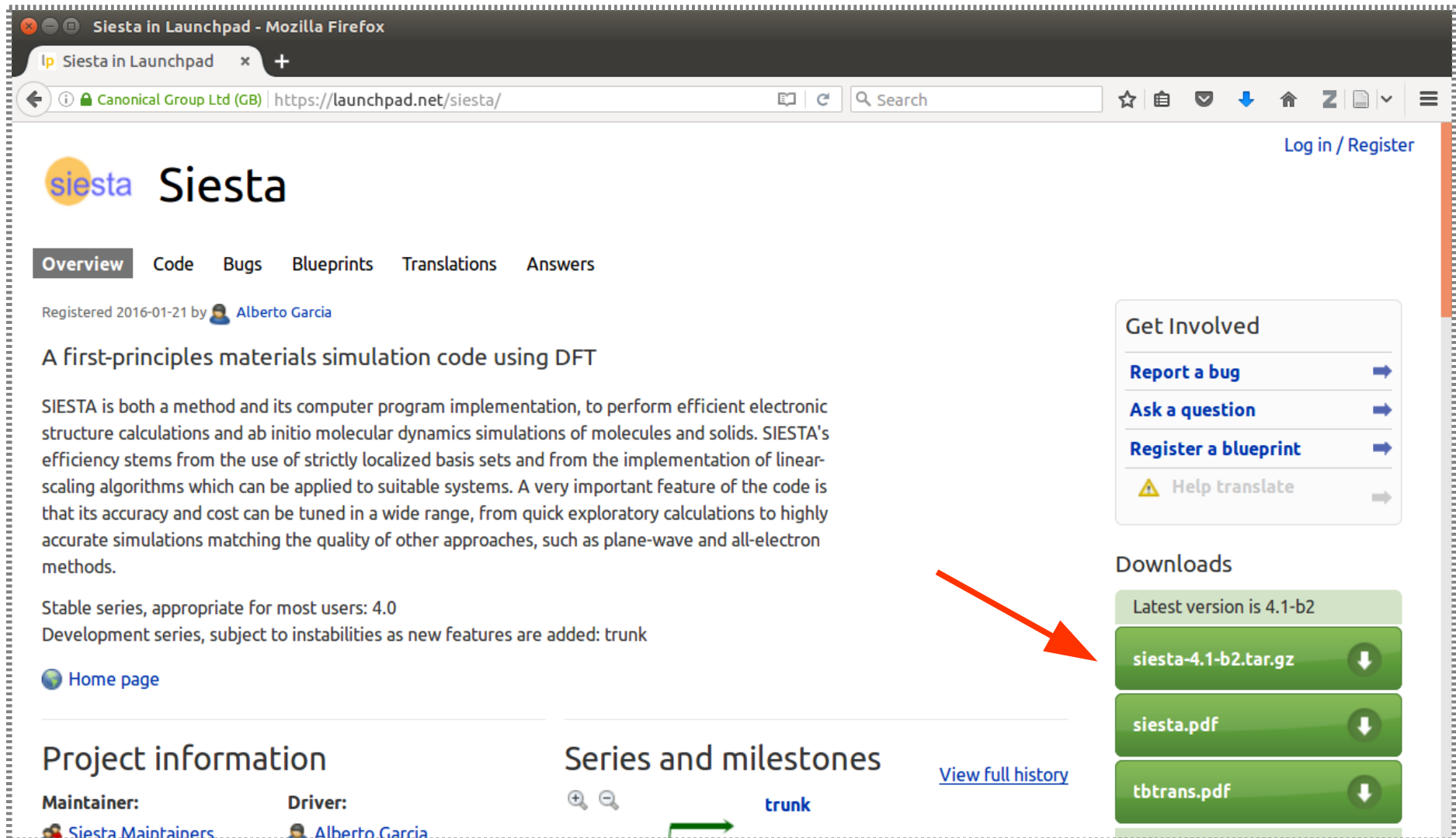
Downloads

Latest version is 4.1-b2

- [siesta-4.1-b2.tar.gz](#)
- [siesta.pdf](#)
- [tbtrans.pdf](#)


HOW TO GET SIESTA

SIESTA IS NOW UNDER THE GNU GENERAL PUBLIC LICENSE



The screenshot shows the Launchpad page for the Siesta project. The browser address bar indicates the URL is <https://launchpad.net/siesta/>. The page features a navigation menu with tabs for Overview, Code, Bugs, Blueprints, Translations, and Answers. The Overview tab is selected, displaying the project's description: "A first-principles materials simulation code using DFT". It also lists the maintainer as Alberto Garcia and provides information about the stable series (4.0) and development series (trunk). A red arrow points to the "Downloads" section, which lists the latest version (4.1-b2) and provides download links for `siesta-4.1-b2.tar.gz`, `siesta.pdf`, and `tbtrans.pdf`. The "Project information" section shows the maintainer as "Siesta Maintainers" and the driver as "Alberto Garcia".


Overview Code Bugs Blueprints Translations Answers

Registered 2016-01-21 by  Alberto Garcia



A first-principles materials simulation code using DFT

SIESTA is both a method and its computer program implementation, to perform efficient electronic structure calculations and ab initio molecular dynamics simulations of molecules and solids. SIESTA's efficiency stems from the use of strictly localized basis sets and from the implementation of linear-scaling algorithms which can be applied to suitable systems. A very important feature of the code is that its accuracy and cost can be tuned in a wide range, from quick exploratory calculations to highly accurate simulations matching the quality of other approaches, such as plane-wave and all-electron methods.



Stable series, appropriate for most users: 4.0
Development series, subject to instabilities as new features are added: trunk

 [Home page](#)


Project information

Maintainer:  Siesta Maintainers
Driver:  Alberto Garcia

Series and milestones

  **trunk** [View full history](#)

Get Involved

- [Report a bug](#) →
- [Ask a question](#) →
- [Register a blueprint](#) →
-  [Help translate](#) →

Downloads

Latest version is 4.1-b2

- [siesta-4.1-b2.tar.gz](#) ↓
- [siesta.pdf](#) ↓
- [tbtrans.pdf](#) ↓

HOW TO GET SIESTA

SIESTA IS NOW UNDER THE GNU GENERAL PUBLIC LICENSE

The screenshot shows the Launchpad page for the Siesta project. The browser address bar indicates the URL is <https://launchpad.net/siesta/>. The page features a navigation menu with tabs for Overview, Code, Bugs, Blueprints, Translations, and Answers. The main content area includes a description of Siesta as a first-principles materials simulation code using DFT, its efficiency, and its stability. A 'Downloads' section on the right lists the latest version (4.1-b2) and provides download links for `siesta-4.1-b2.tar.gz`, `siesta.pdf`, and `tbtrans.pdf`. Two red arrows point to the `siesta-4.1-b2.tar.gz` and `siesta.pdf` download buttons. The 'Project information' section at the bottom shows the maintainer as 'Siesta Maintainers' and the driver as 'Alberto Garcia', with the current series set to 'trunk'.

Siesta in Launchpad - Mozilla Firefox

ip Siesta in Launchpad x +

Canonical Group Ltd (GB) | <https://launchpad.net/siesta/> Search

Log in / Register

siesta Siesta

Overview Code Bugs Blueprints Translations Answers

Registered 2016-01-21 by [Alberto Garcia](#)

A first-principles materials simulation code using DFT

SIESTA is both a method and its computer program implementation, to perform efficient electronic structure calculations and ab initio molecular dynamics simulations of molecules and solids. SIESTA's efficiency stems from the use of strictly localized basis sets and from the implementation of linear-scaling algorithms which can be applied to suitable systems. A very important feature of the code is that its accuracy and cost can be tuned in a wide range, from quick exploratory calculations to highly accurate simulations matching the quality of other approaches, such as plane-wave and all-electron methods.

Stable series, appropriate for most users: 4.0
Development series, subject to instabilities as new features are added: trunk

[Home page](#)

Project information

Maintainer: [Siesta Maintainers](#) Driver: [Alberto Garcia](#)

Series and milestones

[View full history](#)

trunk

Get Involved

- [Report a bug](#)
- [Ask a question](#)
- [Register a blueprint](#)
- [Help translate](#)

Downloads

Latest version is 4.1-b2

- [siesta-4.1-b2.tar.gz](#)
- [siesta.pdf](#)
- [tbtrans.pdf](#)

How To GET SIESTA

FILES THAT YOU SHOULD NOW HAVE IN YOUR COMPUTER:

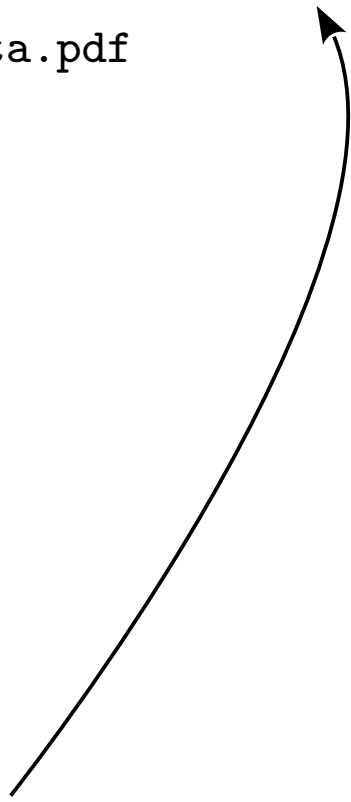
```
-rwxrwxrwx 1 dieguez dieguez 13132430 Dec  6 09:13 siesta-4.1-b2.tar  
-rwxrwxrwx 1 dieguez dieguez   989293 Dec  6 09:14 siesta.pdf
```

HOW TO GET SIESTA

FILES THAT YOU SHOULD NOW HAVE IN YOUR COMPUTER:

```
-rwxrwxrwx 1 dieguez dieguez 13132430 Dec  6 09:13 siesta-4.1-b2.tar  
-rwxrwxrwx 1 dieguez dieguez   989293 Dec  6 09:14 siesta.pdf
```

TAR BALL CONTAINING
ALL THE SIESTA FILES
(SOURCE CODE,
DOCUMENTATION,
EXAMPLES, ETC)



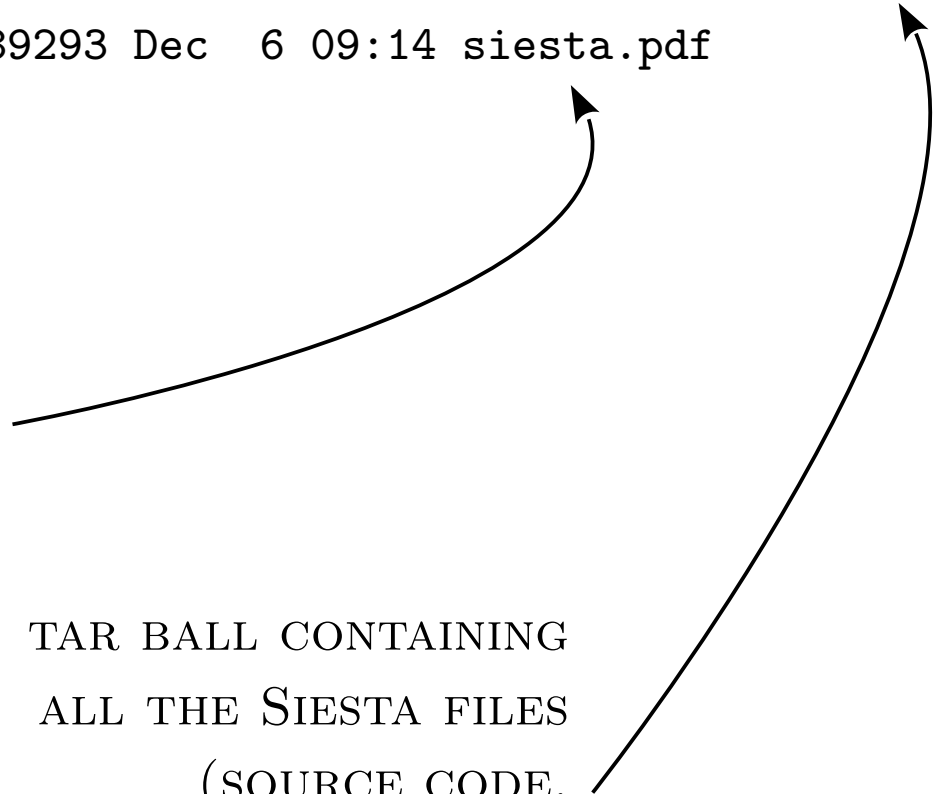
How To GET SIESTA

FILES THAT YOU SHOULD NOW HAVE IN YOUR COMPUTER:

```
-rwxrwxrwx 1 dieguez dieguez 13132430 Dec  6 09:13 siesta-4.1-b2.tar  
-rwxrwxrwx 1 dieguez dieguez   989293 Dec  6 09:14 siesta.pdf
```

SIESTA MANUAL
(OPEN IT...)

TAR BALL CONTAINING
ALL THE SIESTA FILES
(SOURCE CODE,
DOCUMENTATION,
EXAMPLES, ETC)



HOW TO GET SIESTA

THE SIESTA MANUAL

USER'S GUIDE

SIESTA 4.1-b2

August 31, 2016

<http://www.uam.es/siesta>

Contributors to SIESTA

SIESTA is Copyright © 1996-2016 by The Siesta Group:

| | |
|-----------------------|--|
| Emilio Artacho | <i>CIC-Neuquén and University of Cambridge</i> |
| José María Cols | <i>Barcelona Supercomputing Center</i> |
| Julian D. Gale | <i>Curtin University of Technology, Perth</i> |
| Alberto García | <i>Institut de Ciència de Materials, CSIC, Barcelona</i> |
| Javier Junquera | <i>Universidad de Cantabria, Santander</i> |
| Richard M. Martin | <i>University of Illinois at Urbana-Champaign</i> |
| Pablo Ordejon | <i>Centre de Investigació en Nanociència i Nanotecnologia, (CSIC-ICN), Barcelona</i> |
| Daniel Sánchez-Portal | <i>Unidad de Física de Materiales, Centro Mito CSIC-UPV/EHU, San Sebastián</i> |
| José M. Soler | <i>Universidad Autónoma de Madrid</i> |

The SIESTA project was initiated by Pablo Ordejon (then at the Univ. de Oviedo), and José M. Soler and Emilio Artacho (Univ. Autónoma de Madrid, UAM). The development team was then joined by Alberto García (then at Univ. del País Vasco, Bilbao), Daniel Sánchez-Portal (UAM), and Javier Junquera (Univ. de Oviedo and later UAM), and sometime later by Julian Gale (then at Imperial College, London). In 2007 José M. Cols (Barcelona Supercomputing Center, BSC) became a core developer.

The current TransSIESTA module within SIESTA is developed by Nick R. Papko (then at Technical University of Denmark) and Mads Brandbyge. The original TransSIESTA module was developed by Pablo Ordejon and José L. Moros (then at ICMAB-CSIC), and Mads Brandbyge, Kurt Stokbro, and Jeremy Taylor (Technical Univ. of Denmark).

Other contributors (we apologise for any omissions):

Eduardo Anglada, Thomas Archer, Luis C. Balbuena, Xavier Blase, Ramon Casandró, Michele Corradi, Raúl de la Cruz, Gabriel Fabozius, Mariví Fernández-Serna, Jaime Ferrer, Chai-Chun Fu, Sandra García, Víctor M. García-Suarez, Georg Hubs, Rogell Grima, Rainer Hoff, Jorgo Kohnenoff, Richard Korytar, In-Ho Lee, Lin Lin, Nicolas Louate, Miguel Luján, Eduardo Machado, Mador Machado, José Luis Martínez, Volodymyr Maslyuk, Juana Moreno, Frederico Dutilleul Nomes, Michael Oliveira, Nick Rührner Papko, Magnus Paulson, Oscar Paz, Andriy Postnikov, Tistiana Sandon, Andrew Walker, Andrew Walkingshaw, Toby White, Francois Williams, Chao Yang.

O.F. Sankey, D.J. Niklewski and D.A. Drabold made the FIREBALL code available to P. Ordejon. Although we no longer use the routines in that code, it was essential in the initial development of SIESTA, which still uses many of the algorithms developed by them.

HOW TO GET SIESTA

THE SIESTA MANUAL

SIESTA (Spanish Initiative for Electronic Simulations with Thousands of Atoms) is both a method and its computer program implementation, to perform electronic structure calculations and *ab initio* molecular dynamics simulations of molecules and solids. Its main characteristics are:

- It uses the standard Kohn-Sham selfconsistent density functional method in the local density (LDA-LSD) and generalized gradient (GGA) approximations, as well as in a non local functional that includes van der Waals interactions (VDW-DF).
- It uses norm-conserving pseudopotentials in their fully nonlocal (Kleinman-Bylander) form.
- It uses atomic orbitals as a basis set, allowing unlimited multiple-zeta and angular momenta, polarization and off-site orbitals. The radial shape of every orbital is numerical and any shape can be used and provided by the user, with the only condition that it has to be of finite support, i.e., it has to be strictly zero beyond a user-provided distance from the corresponding nucleus. Finite-support basis sets are the key for calculating the Hamiltonian and overlap matrices in $O(N)$ operations.
- Projects the electron wavefunctions and density onto a real-space grid in order to calculate the Hartree and exchange-correlation potentials and their matrix elements.
- Besides the standard Rayleigh-Ritz eigenstate method, it allows the use of localized linear combinations of the occupied orbitals (valence-bond or Wannier-like functions), making the computer time and memory scale linearly with the number of atoms. Simulations with several hundred atoms are feasible with modest workstations.
- It is written in Fortran 95 and memory is allocated dynamically.
- It may be compiled for serial or parallel execution (under MPI).

HOW TO GET SIESTA

THE SIESTA MANUAL

PAGES 8-9

And also (though not all options are compatible):

- Geometry relaxation, fixed or variable cell.
- Constant-temperature molecular dynamics (Nose thermostat).
- Variable cell dynamics (Parrinello-Rahman).
- Spin polarized calculations (collinear or not).
- k-sampling of the Brillouin zone.
- Local and orbital-projected density of states.
- COOP and COHP curves for chemical bonding analysis.
- Dielectric polarization.
- Vibrations (phonons).
- Band structure.
- Ballistic electron transport under non-equilibrium (through TRANSIESTA)

It routinely provides:

- Total and partial energies.
- Atomic forces.
- Stress tensor.
- Electric dipole moment.
- Atomic, orbital and bond populations (Mulliken).
- Electron density.

HOW TO INSTALL SIESTA: **marconi**

TRANSFER THE SIESTA TAR BALL TO **marconi**:

- THIS IS TYPICALLY DONE USING A SCP UTILITY
- FOR EXAMPLE, FROM MY LINUX MACHINE, I DID:

```
scp /home/diequez/Downloads/siesta-4.1-b2.tar.gz  
a08tra74@login.marconi.cineca.it:
```

HOW TO INSTALL SIESTA: **marconi**

TRANSFER THE SIESTA TAR BALL TO **marconi**:

- THIS IS TYPICALLY DONE USING A SCP UTILITY
- FOR EXAMPLE, FROM MY LINUX MACHINE, I DID:

```
scp /home/dieguez/Downloads/siesta-4.1-b2.tar.gz  
a08tra74@login.marconi.cineca.it:
```

LOG IN TO **marconi** AND CHECK THAT THE TAR BALL IS THERE:

```
[a08tra74@r000u06l01 ~]$ ls  
siesta-4.1-b2.tar.gz
```

HOW TO INSTALL SIESTA: **marconi**

CREATE NEW DIRECTORY, MOVE THERE THE TAR BALL, AND
UNCOMPRESS IT:

```
[a08tra74@r000u06l01 ~]$ mkdir SIESTA
```

HOW TO INSTALL SIESTA: **marconi**

CREATE NEW DIRECTORY, MOVE THERE THE TAR BALL, AND
UNCOMPRESS IT:

```
[a08tra74@r000u06l01 ~]$ mkdir SIESTA
```

```
[a08tra74@r000u06l01 ~]$ mv siesta-4.1-b2.tar.gz SIESTA
```

HOW TO INSTALL SIESTA: **marconi**

CREATE NEW DIRECTORY, MOVE THERE THE TAR BALL, AND UNCOMPRESS IT:

```
[a08tra74@r000u06l01 ~]$ mkdir SIESTA
```

```
[a08tra74@r000u06l01 ~]$ mv siesta-4.1-b2.tar.gz SIESTA
```

```
[a08tra74@r000u06l01 ~]$ cd SIESTA
```

HOW TO INSTALL SIESTA: **marconi**

CREATE NEW DIRECTORY, MOVE THERE THE TAR BALL, AND UNCOMPRESS IT:

```
[a08tra74@r000u06l01 ~]$ mkdir SIESTA
```

```
[a08tra74@r000u06l01 ~]$ mv siesta-4.1-b2.tar.gz SIESTA
```

```
[a08tra74@r000u06l01 ~]$ cd SIESTA
```

```
[a08tra74@r000u06l01 SIESTA]$ gunzip siesta-4.1-b2.tar.gz
```

HOW TO INSTALL SIESTA: marconi

CREATE NEW DIRECTORY, MOVE THERE THE TAR BALL, AND UNCOMPRESS IT:

```
[a08tra74@r000u06l01 ~]$ mkdir SIESTA
```

```
[a08tra74@r000u06l01 ~]$ mv siesta-4.1-b2.tar.gz SIESTA
```

```
[a08tra74@r000u06l01 ~]$ cd SIESTA
```

```
[a08tra74@r000u06l01 SIESTA]$ gunzip siesta-4.1-b2.tar.gz
```

```
[a08tra74@r000u06l01 SIESTA]$ tar -xvf siesta-4.1-b2.tar
```

HOW TO INSTALL SIESTA: marconi

CREATE NEW DIRECTORY, MOVE THERE THE TAR BALL, AND UNCOMPRESS IT:

```
[a08tra74@r000u06l01 ~]$ mkdir SIESTA
[a08tra74@r000u06l01 ~]$ mv siesta-4.1-b2.tar.gz SIESTA
[a08tra74@r000u06l01 ~]$ cd SIESTA
[a08tra74@r000u06l01 SIESTA]$ gunzip siesta-4.1-b2.tar.gz
[a08tra74@r000u06l01 SIESTA]$ tar -xvf siesta-4.1-b2.tar
[a08tra74@r000u06l01 SIESTA]$ ls -l
total 46400
drwxr-xr-x 10 a08tra74 corsi      4096 Nov 28 10:35 siesta-4.1-b2
-rw-r--r--  1 a08tra74 corsi 47513600 Dec  6 12:09 siesta-4.1-b2.tar
```


HOW TO INSTALL SIESTA: **marconi**

WHAT IS IN THE NEWLY CREATED DIRECTORY?

```
[a08tra74@r000u06l01 SIESTA]$ cd siesta-4.1-b2/
```

HOW TO INSTALL SIESTA: **marconi**

WHAT IS IN THE NEWLY CREATED DIRECTORY?

```
[a08tra74@r000u06l01 SIESTA]$ cd siesta-4.1-b2/
```

```
[a08tra74@r000u06l01 siesta-4.1-b2]$ ls
```

| | | | | |
|------------------|------------|-------------------|-----------|--------------|
| AUTHORS | Docs | Pseudo | Src | version.info |
| CHANGES | Examples | README | Tests | |
| CHANGES_DETAILED | NOTICE.txt | README_TRANSIESTA | Tutorials | |
| COPYING | Obj | RELEASE_NOTES | Util | |

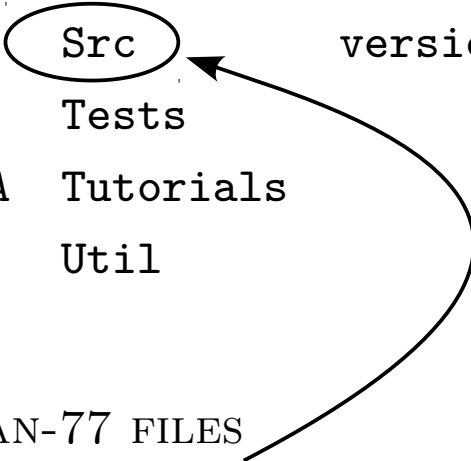
HOW TO INSTALL SIESTA: **marconi**

WHAT IS IN THE NEWLY CREATED DIRECTORY?

```
[a08tra74@r000u06l01 SIESTA]$ cd siesta-4.1-b2/
```

```
[a08tra74@r000u06l01 siesta-4.1-b2]$ ls
```

| | | | | |
|------------------|------------|-------------------|------------|--------------|
| AUTHORS | Docs | Pseudo | Src | version.info |
| CHANGES | Examples | README | Tests | |
| CHANGES_DETAILED | NOTICE.txt | README_TRANSIESTA | Tutorials | |
| COPYING | Obj | RELEASE_NOTES | Util | |



VERSION IN MID-2014: 269 FORTRAN-77 FILES
207 FORTRAN-90 FILES
15 SHELL SCRIPTS
23 M4 FILES
3 MATLAB FILES
1 C FILE
185,263 LINES OF CODE

HOW TO INSTALL SIESTA: **marconi**

HOW TO COMPILE A SERIAL EXECUTABLE?

(SEE SECTION 2 OF THE SIESTA MANUAL)

```
[a08tra74@r000u06l01 siesta-4.1-b2]$ cd Obj/
```

HOW TO INSTALL SIESTA: **marconi**

HOW TO COMPILE A SERIAL EXECUTABLE?

(SEE SECTION 2 OF THE SIESTA MANUAL)

```
[a08tra74@r000u06l01 siesta-4.1-b2]$ cd Obj/  
[a08tra74@r000u06l01 Obj]$ sh ../Src/obj_setup.sh  
*** Compilation setup done.  
*** Remember to copy an arch.make file into the directory.  
*** These files are template arch.make files:  
***     gfortran.make (for gfortran compiler)  
***     intel.make (for intel compiler)  
***     DOCUMENTED-TEMPLATE.make (requires customization)
```

HOW TO INSTALL SIESTA: **marconi**

HOW TO COMPILE A SERIAL EXECUTABLE?

(SEE SECTION 2 OF THE SIESTA MANUAL)

```
[a08tra74@r000u06l01 siesta-4.1-b2]$ cd Obj/  
[a08tra74@r000u06l01 Obj]$ sh ../Src/obj_setup.sh  
*** Compilation setup done.  
*** Remember to copy an arch.make file into the directory.  
*** These files are template arch.make files:  
***     gfortran.make (for gfortran compiler)  
***     intel.make (for intel compiler)  
***     DOCUMENTED-TEMPLATE.make (requires customization)  
[a08tra74@r000u06l01 Obj]$ cp gfortran.make arch.make
```

HOW TO INSTALL SIESTA: **marconi**

HOW TO COMPILE A SERIAL EXECUTABLE?

(SEE SECTION 2 OF THE SIESTA MANUAL)

```
[a08tra74@r000u06l01 siesta-4.1-b2]$ cd Obj/  
[a08tra74@r000u06l01 Obj]$ sh ../Src/obj_setup.sh  
*** Compilation setup done.  
*** Remember to copy an arch.make file into the directory.  
*** These files are template arch.make files:  
***     gfortran.make (for gfortran compiler)  
***     intel.make (for intel compiler)  
***     DOCUMENTED-TEMPLATE.make (requires customization)  
[a08tra74@r000u06l01 Obj]$ cp gfortran.make arch.make  
[a08tra74@r000u06l01 Obj]$ make  
[...lots of output messages about the compilation...]
```

HOW TO INSTALL SIESTA: marconi

HOW TO COMPILE A SERIAL EXECUTABLE?

(SEE SECTION 2 OF THE SIESTA MANUAL)

```
[a08tra74@r000u06l01 siesta-4.1-b2]$ cd Obj/
[a08tra74@r000u06l01 Obj]$ sh ../Src/obj_setup.sh
*** Compilation setup done.
*** Remember to copy an arch.make file into the directory.
*** These files are template arch.make files:
***     gfortran.make (for gfortran compiler)
***     intel.make (for intel compiler)
***     DOCUMENTED-TEMPLATE.make (requires customization)
[a08tra74@r000u06l01 Obj]$ cp gfortran.make arch.make
[a08tra74@r000u06l01 Obj]$ make
[...lots of output messages about the compilation...]
[a08tra74@r000u06l01 Obj]$ ls -l siesta
-rwxr-xr-x 1 a08tra74 corsi 7298284 Dec  6 15:06 siesta
```


HOW TO INSTALL SIESTA: **marconi**

HOW TO COMPILE ANOTHER SERIAL EXECUTABLE?

```
[a08tra74@r000u06l01 Obj]$ cd ..
```

HOW TO INSTALL SIESTA: **marconi**

HOW TO COMPILE ANOTHER SERIAL EXECUTABLE?

```
[a08tra74@r000u06l01 Obj]$ cd ..
```

```
[a08tra74@r000u06l01 siesta-4.1-b2]$ mkdir Obj_intel/
```

HOW TO INSTALL SIESTA: **marconi**

HOW TO COMPILE ANOTHER SERIAL EXECUTABLE?

```
[a08tra74@r000u06l01 Obj]$ cd ..
```

```
[a08tra74@r000u06l01 siesta-4.1-b2]$ mkdir Obj_intel/
```

```
[a08tra74@r000u06l01 siesta-4.1-b2]$ cd Obj_intel/
```

HOW TO INSTALL SIESTA: marconi

HOW TO COMPILE ANOTHER SERIAL EXECUTABLE?

```
[a08tra74@r000u06l01 Obj]$ cd ..  
[a08tra74@r000u06l01 siesta-4.1-b2]$ mkdir Obj_intel/  
[a08tra74@r000u06l01 siesta-4.1-b2]$ cd Obj_intel/  
[a08tra74@r000u06l01 Obj_intel]$ sh ../Src/obj_setup.sh  
*** Compilation setup done.  
*** Remember to copy an arch.make file into the directory.  
*** These files are template arch.make files:  
***     gfortran.make (for gfortran compiler)  
***     intel.make (for intel compiler)  
***     DOCUMENTED-TEMPLATE.make (requires customization)
```

HOW TO INSTALL SIESTA: **marconi**

HOW TO COMPILE ANOTHER SERIAL EXECUTABLE?

```
[a08tra74@r000u06l01 Obj]$ cd ..  
[a08tra74@r000u06l01 siesta-4.1-b2]$ mkdir Obj_intel/  
[a08tra74@r000u06l01 siesta-4.1-b2]$ cd Obj_intel/  
[a08tra74@r000u06l01 Obj_intel]$ sh ../Src/obj_setup.sh  
*** Compilation setup done.  
*** Remember to copy an arch.make file into the directory.  
*** These files are template arch.make files:  
***     gfortran.make (for gfortran compiler)  
***     intel.make (for intel compiler)  
***     DOCUMENTED-TEMPLATE.make (requires customization)  
[a08tra74@r000u06l01 Obj_intel]$ cp ../Obj/intel.make arch.make
```

HOW TO INSTALL SIESTA: **marconi**

HOW TO COMPILE ANOTHER SERIAL EXECUTABLE?

```
[a08tra74@r000u06l01 Obj]$ cd ..  
[a08tra74@r000u06l01 siesta-4.1-b2]$ mkdir Obj_intel/  
[a08tra74@r000u06l01 siesta-4.1-b2]$ cd Obj_intel/  
[a08tra74@r000u06l01 Obj_intel]$ sh ../Src/obj_setup.sh  
*** Compilation setup done.  
*** Remember to copy an arch.make file into the directory.  
*** These files are template arch.make files:  
***     gfortran.make (for gfortran compiler)  
***     intel.make (for intel compiler)  
***     DOCUMENTED-TEMPLATE.make (requires customization)  
[a08tra74@r000u06l01 Obj_intel]$ cp ../Obj/intel.make arch.make  
[a08tra74@r000u06l01 Obj_intel]$ module load intel
```

HOW TO INSTALL SIESTA: **marconi**

HOW TO COMPILE ANOTHER SERIAL EXECUTABLE?

```
[a08tra74@r000u06l01 Obj]$ cd ..  
[a08tra74@r000u06l01 siesta-4.1-b2]$ mkdir Obj_intel/  
[a08tra74@r000u06l01 siesta-4.1-b2]$ cd Obj_intel/  
[a08tra74@r000u06l01 Obj_intel]$ sh ../Src/obj_setup.sh  
*** Compilation setup done.  
*** Remember to copy an arch.make file into the directory.  
*** These files are template arch.make files:  
***     gfortran.make (for gfortran compiler)  
***     intel.make (for intel compiler)  
***     DOCUMENTED-TEMPLATE.make (requires customization)  
[a08tra74@r000u06l01 Obj_intel]$ cp ../Obj/intel.make arch.make  
[a08tra74@r000u06l01 Obj_intel]$ module load intel  
[a08tra74@r000u06l01 Obj_intel]$ make  
[...lots of output messages about the compilation...]
```

HOW TO INSTALL SIESTA: **marconi**

HOW TO COMPILE ANOTHER SERIAL EXECUTABLE?

```
[a08tra74@r000u06l01 Obj]$ cd ..
[a08tra74@r000u06l01 siesta-4.1-b2]$ mkdir Obj_intel/
[a08tra74@r000u06l01 siesta-4.1-b2]$ cd Obj_intel/
[a08tra74@r000u06l01 Obj_intel]$ sh ../Src/obj_setup.sh
*** Compilation setup done.
*** Remember to copy an arch.make file into the directory.
*** These files are template arch.make files:
***     gfortran.make (for gfortran compiler)
***     intel.make (for intel compiler)
***     DOCUMENTED-TEMPLATE.make (requires customization)
[a08tra74@r000u06l01 Obj_intel]$ cp ../Obj/intel.make arch.make
[a08tra74@r000u06l01 Obj_intel]$ module load intel
[a08tra74@r000u06l01 Obj_intel]$ make
[...lots of output messages about the compilation...]
[a08tra74@r000u06l01 Obj_intel]$ ls -l siesta
-rwxr-xr-x 1 a08tra74 corsi 12122531 Dec  6 16:33 siesta
```


HOW TO INSTALL SIESTA: **marconi**

HOW TO COMPILE A PARALLEL EXECUTABLE?

```
[a08tra74@r000u06l01 Obj_intel]$ cd ..  
[a08tra74@r000u06l01 siesta-4.1-b2]$ mkdir Obj_parallel/  
[a08tra74@r000u06l01 siesta-4.1-b2]$ cd Obj_parallel/  
[a08tra74@r000u06l01 Obj_parallel]$ sh ../Src/obj_setup.sh  
*** Compilation setup done.  
*** Remember to copy an arch.make file into the directory.  
*** These files are template arch.make files:  
***     gfortran.make (for gfortran compiler)  
***     intel.make (for intel compiler)  
***     DOCUMENTED-TEMPLATE.make (requires customization)  
[a08tra74@r000u06l01 Obj_parallel]$ cp ../Obj/DOCUMENTED-TEMPLATE.make  
arch.make
```

...AND NOW EDIT THE **arch.make** TO SUIT **marconi**...

HOW TO RUN SIESTA

FIRST, PREPARE THE INPUT FILES...

```
[a08tra74@r000u06l01 Obj_intel]$ cd ..
```

```
[a08tra74@r000u06l01 siesta-4.1-b2]$ cd ..
```

```
[a08tra74@r000u06l01 SIESTA]$ mkdir h2o/
```

```
[a08tra74@r000u06l01 SIESTA]$ cd h2o/
```

```
[a08tra74@r000u06l01 h2o]$ cp ../siesta-4.1-b2/Examples/H2O/h2o.fdf .
```

HOW TO RUN SIESTA

```
[a08tra74@r000u06l01 h2o]$ cat h2o.fdf
```

```
SystemName          Water molecule
```

```
SystemLabel         h2o
```

```
NumberOfAtoms       3
```

```
NumberOfSpecies     2
```

```
%block ChemicalSpeciesLabel
```

```
 1  8  0      # Species index, atomic number, species label
```

```
 2  1  H
```

```
%endblock ChemicalSpeciesLabel
```

```
AtomicCoordinatesFormat  Ang
```

```
%block AtomicCoordinatesAndAtomicSpecies
```

```
 0.000  0.000  0.000  1
```

```
 0.757  0.586  0.000  2
```

```
-0.757  0.586  0.000  2
```

```
%endblock AtomicCoordinatesAndAtomicSpecies
```

HOW TO RUN SIESTA

```
[a08tra74@r000u06l01 h2o]$ cp ../siesta-4.1-b2/Examples/Vps/O.psf .  
[a08tra74@r000u06l01 h2o]$ cp ../siesta-4.1-b2/Examples/Vps/H.psf .  
[a08tra74@r000u06l01 h2o]$ ../siesta-4.1-b2/Obj/siesta < h2o.fdf
```

TUTORIALS

- TUTORIAL 1 (BY JAVIER JUNQUERA):

H₂O MOLECULE: CONVERGING THE SIZE OF THE UNIT CELL FOR A POLAR MOLECULE

- TUTORIAL 2 (BY JAVIER JUNQUERA):

LATTICE CONSTANT, BULK MODULUS, AND EQUILIBRIUM ENERGY OF SOLIDS: BULK SI

- TUTORIAL 3 (BY JAVIER JUNQUERA):

ANALYSIS OF THE CHARGE DENSITY IN REAL SPACE OF A SOLID: BULK SI

- TUTORIAL 4 (BY JAVIER JUNQUERA):

BAND STRUCTURE OF AN IONIC SOLID: THE CASE OF MgO

A WORD ABOUT PARALLELISM IN SIESTA

FROM THE SIESTA MANUAL:

2.3 Parallel

To achieve a parallel build of SIESTA one should first determine which type of parallelism one requires. It is advised to use MPI for calculations with moderate number of cores. If one requires eXa-scale parallelism SIESTA provides hybrid parallelism using both MPI and OpenMP.

[...]

If using only 1 thread per MPI core it is advised to compile SIESTA without OpenMP. As such it may be advantageous to compile SIESTA in 3 variants; OpenMP-only (small systems), MPI-only (medium to large systems) and MPI+OpenMP (large+ systems).

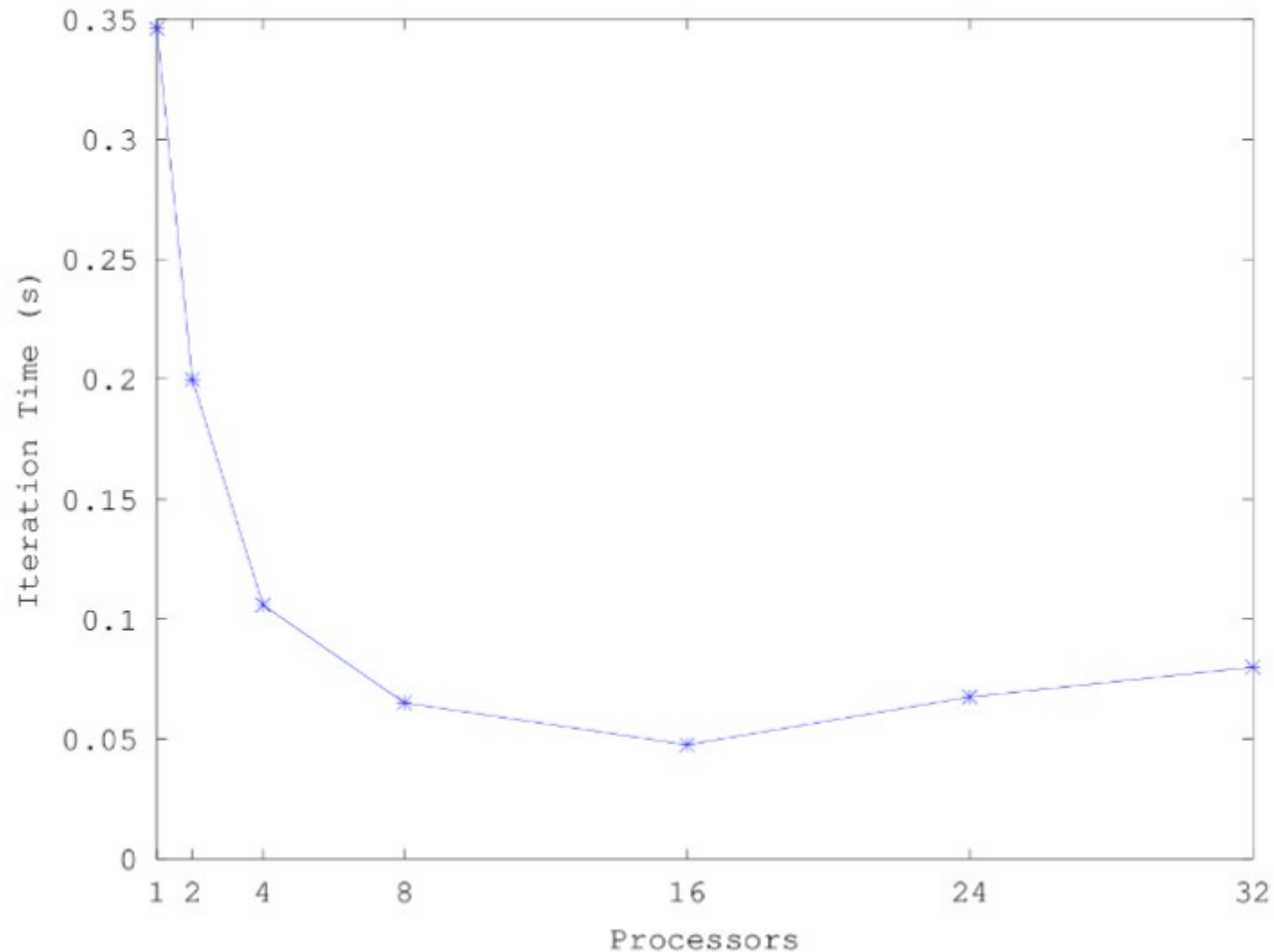
Diag.ParallelOverK false

(logical)

For the diagonalisation there is a choice in strategy about whether to parallelise over the K points or over the orbitals. K point diagonalisation is close to perfectly parallel but is only useful where the number of K points is much larger than the number of processors and therefore orbital parallelisation is generally preferred. The exception is for metals where the unit cell is small, but the number of K points to be sampled is very large. In this last case it is recommended that this option be used.

A WORD ABOUT PARALLELISM IN SIESTA

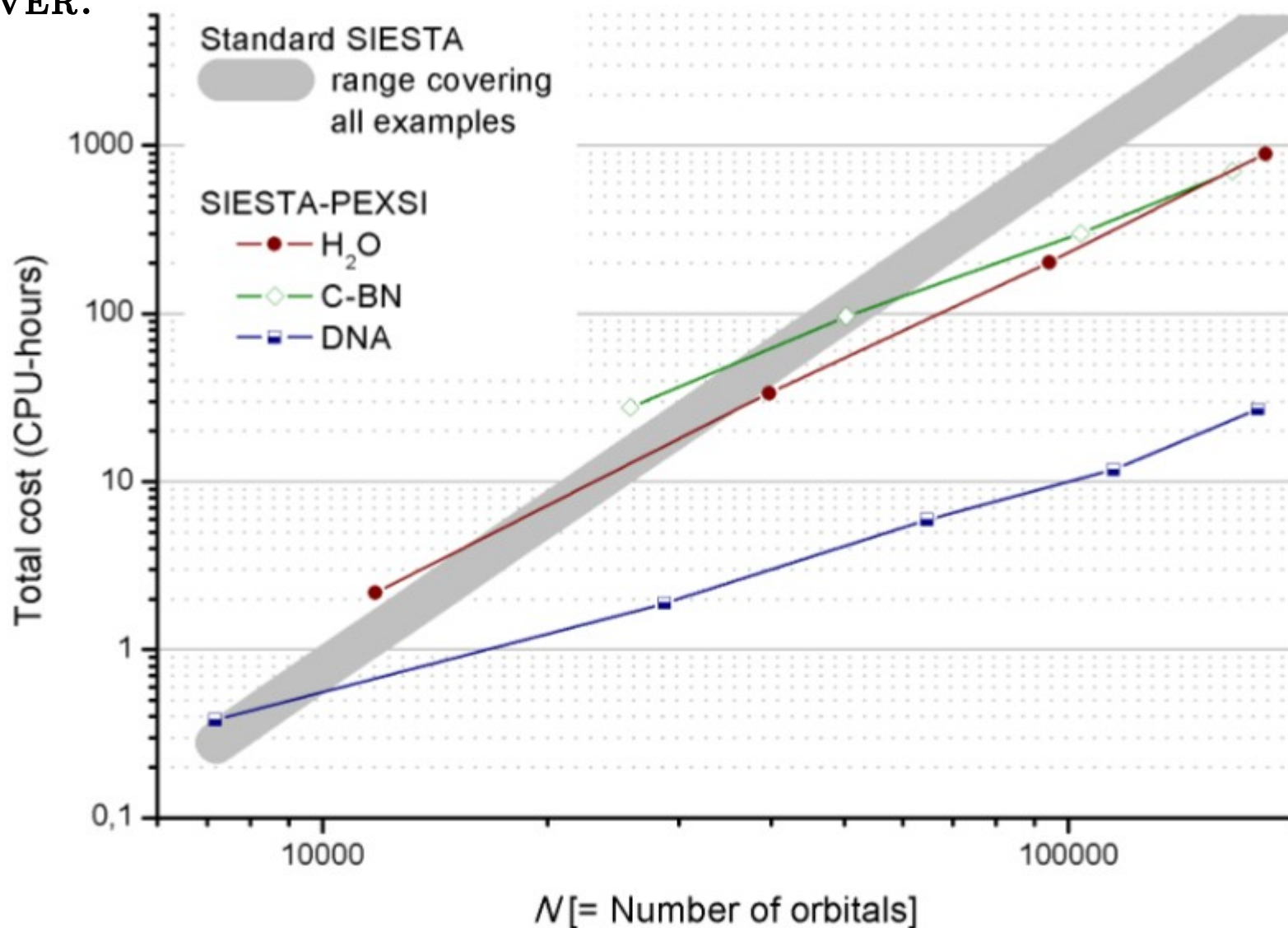
FROM PERSONAL EXPERIENCE:



INFORMATION FROM: SIESTA TUTORIAL IN ZARAGOZA (2014)

A WORD ABOUT PARALLELISM IN SIESTA

PEXSI SOLVER:



A WORD ABOUT PARALLELISM IN SIESTA

IOP Publishing

Journal of Physics: Condensed Matter

J. Phys.: Condens. Matter **26** (2014) 305503 (15pp)

doi:[10.1088/0953-8984/26/30/305503](https://doi.org/10.1088/0953-8984/26/30/305503)

SIESTA-PEXSI: massively parallel method for efficient and accurate *ab initio* materials simulation without matrix diagonalization

Lin Lin¹, Alberto García², Georg Huhs³ and Chao Yang¹

Abstract

We describe a scheme for efficient large-scale electronic-structure calculations based on the combination of the pole expansion and selected inversion (PEXSI) technique with the SIESTA method, which uses numerical atomic orbitals within the Kohn–Sham density functional theory (KSDFT) framework. The PEXSI technique can efficiently utilize the sparsity pattern of the Hamiltonian and overlap matrices generated in SIESTA, and for large systems it has a much lower computational complexity than that associated with the matrix diagonalization procedure. The PEXSI technique can be used to evaluate the electron density, free energy, atomic forces, density of states and local density of states without computing any eigenvalue or eigenvector of the Kohn–Sham Hamiltonian. It can achieve accuracy fully comparable to that obtained from a matrix diagonalization procedure for general systems, including metallic systems at low temperature. The PEXSI method is also highly scalable. With the recently developed massively parallel PEXSI technique, we can make efficient use of more than 10000 processors on high performance machines. We demonstrate the performance and accuracy of the SIESTA-PEXSI method using several examples of large scale electronic structure calculations, including 1D, 2D and bulk problems with insulating, semi-metallic, and metallic character.

MORE ABOUT SIESTA

- TUTORIALS: [WEBPAGES OF JAVIER JUNQUERA](#)
- DOCUMENTATION (TUTORIALS, MANUALS, VIDEOS): [OLD SIESTA PAGE](#)
- ABOUT THE METHOD: [TECHNICAL PUBLICATIONS](#)
- GETTING SIESTA AND COLLABORATING: [LAUNCHPAD](#)
- ASKING QUESTIONS: [SIESTA MAILING LIST](#)

dieguez@tau.ac.il