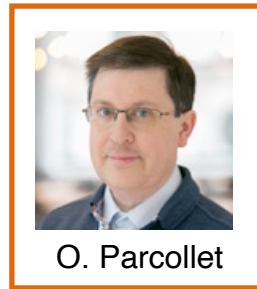
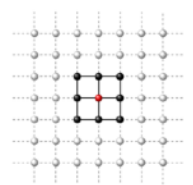


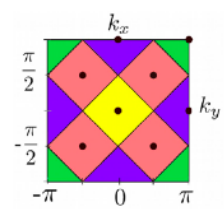


A Software Platform for Quantum Embedding





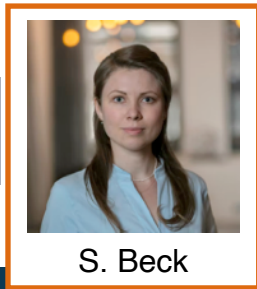
O. Parcollet



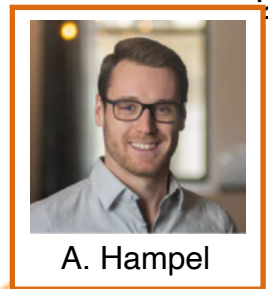
DMFT & Cluster Extensions



WANNIER90



S. Beck



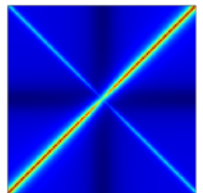
A. Hampel

DFT + DMFT

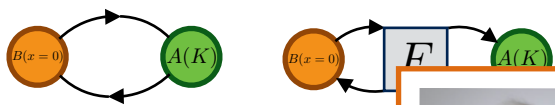
dft tools
solid dmft



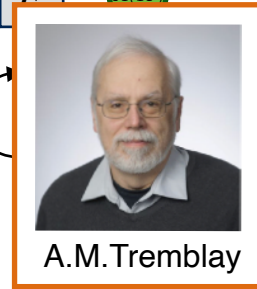
TRIQS



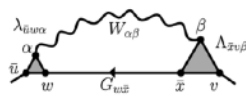
Vertex Methods



$$F \approx \Gamma_{\text{imp}} + \Gamma_{\text{imp}}$$



A.M. Tremblay



Impurity Solvers

ED

CTQMC

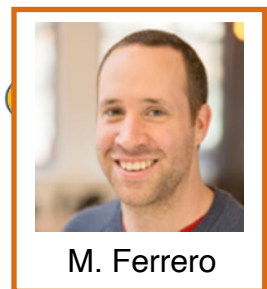
NRG

DMRG

DiagMC

PT

Non-Equilibrium



M. Ferrero

TRIQS Library

- TRIQS - A **T**oolbox for **R**esearch on **I**nteracting **Q**uantum **S**ystems
 - TRIQS Library — Fundamental Building Blocks
 - Applications based on the TRIQS Library



triqs.github.io

Releases 14

 Version 3.2.1 Latest
on Mar 21

TRIQS Library

- TRIQS - A **T**oolbox for **R**esearch on **I**nteracting **Q**uantum **S**ystems
 - TRIQS Library — Fundamental Building Blocks
 - Applications based on the TRIQS Library





triqs.github.io

Releases 14

 Version 3.2.1 Latest
on Mar 21

TRIQS Library

- TRIQS - A **T**oolbox for **R**esearch on **I**nteracting **Q**uantum **S**ystems
 - TRIQS Library — Fundamental Building Blocks
 - Applications based on the TRIQS Library
- Open source (GPLv3 and Apache 2).
- High-level Interface in Python 3 
- Low-level Backend in Modern C++ 





triqs.github.io

Releases 14

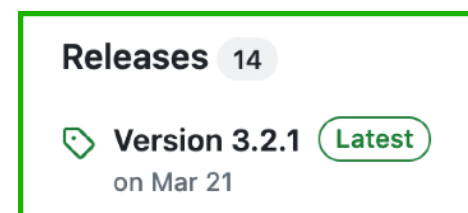
 Version 3.2.1 Latest
on Mar 21

TRIQS Library

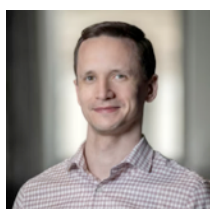
- TRIQS - A Toolbox for **R**esearch on **I**nteracting **Q**uantum **S**ystems
 - TRIQS Library — Fundamental Building Blocks
 - Applications based on the TRIQS Library
- Open source (GPLv3 and Apache 2).
- High-level Interface in Python 3 
- Low-level Backend in Modern C++ 



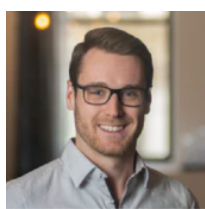
triqs.github.io



O. Parcollet



M. Ferrero



A. Hampel



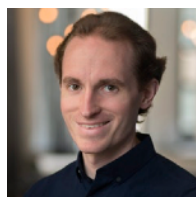
I. Krivenko



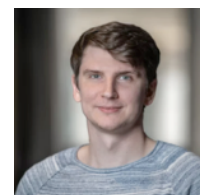
T. Ayrat



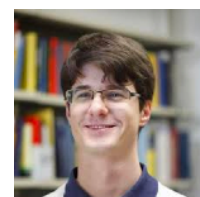
H. Strand



D. Simon



D. Kiese



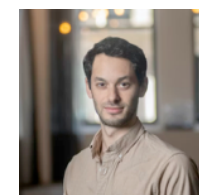
M. Zingl



P. Dumitrescu



A. Moutenet



J. Kaye



S. Beck

TRIQS — Software Stack



TRIQS




Version 3.1.1

Latest



- Green Functions
- Many-Body Operators
- Lattice Tools
- Exact Diagonalization
- Monte Carlo Tools
- Statistical Analysis Tools


TRIQS — Software Stack

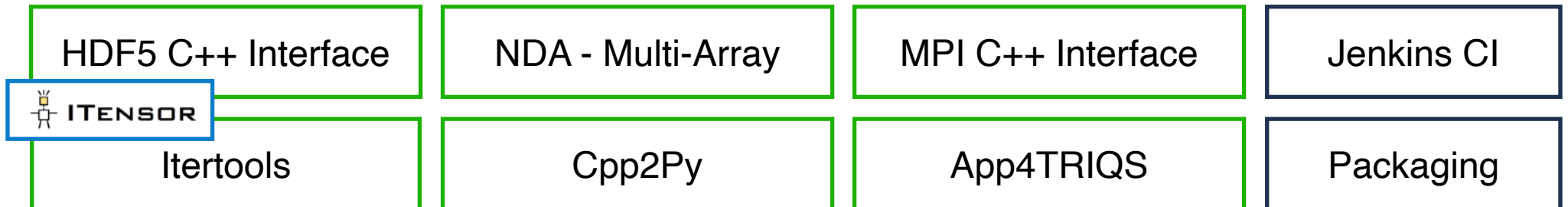


TRIQS

- Green Functions
- Exact Diagonalization
- Many-Body Operators
- Monte Carlo Tools
- Lattice Tools
- Statistical Analysis Tools

 Version 3.1.1 Latest



TRIQS — Software Stack

Solvers

- CT-Hyb
- CT-Seg
- CT-Int
- Inchworm
- ForkTPS
- Keldy Quasi-MC
- HubbardI
- Hartree Fock

Electronic Structure

- DFTTools
- solid_dmft
- FermiSee

Vertex

- TPRF
- TRILEX
- SBE

Tools

- MaxEnt
- Nevanlinna
- SolverBenchmarks

Interfaces

- NRG LJublana
- OmegaMaxEnt
- W2Dynamics
- Pomerol



TRIQS



Version 3.1.1

Latest

- Green Functions
- Many-Body Operators
- Lattice Tools
- Exact Diagonalization
- Monte Carlo Tools
- Statistical Analysis Tools



HDF5 C++ Interface

NDA - Multi-Array

MPI C++ Interface

Jenkins CI



lertools

Cpp2Py

App4TRIQS

Packaging

TRIQS Applications — CT-Hyb QMC

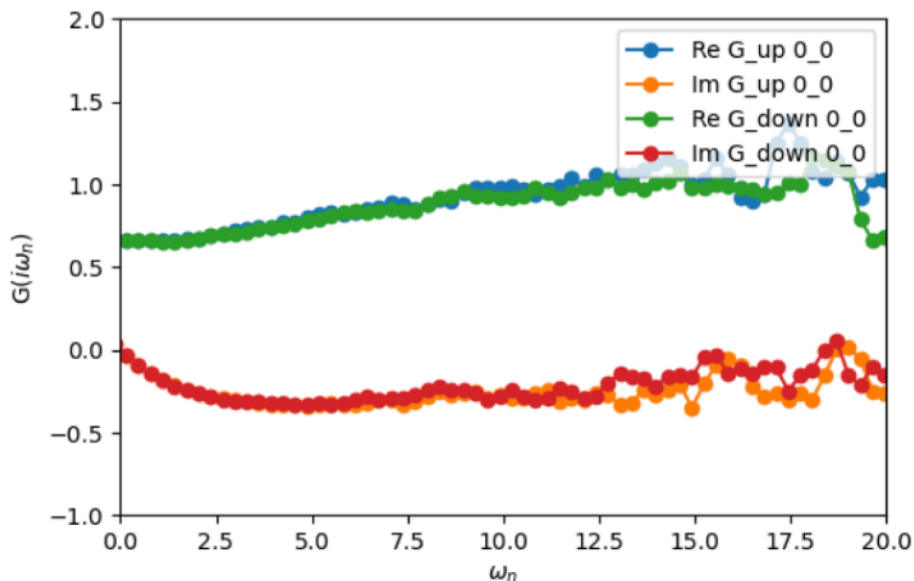
triqs.github.io/cthyb

- Quantum Impurity Solver
- Hybridization Expansion
- Generic Multi-band/orbital Interactions
- Complex Interactions $\sum_{ijkl} \sum_{\sigma\sigma'} U_{ijkl} c_{\sigma i}^\dagger c_{\sigma' j}^\dagger c_{\sigma' k} c_{\sigma l}$

What can we measure?

$$\langle \mathcal{T} c_{\sigma i}(\tau) c_{\sigma j}^\dagger \rangle$$

$$\langle \mathcal{T} c_{\sigma i}^\dagger(i\omega) c_{\sigma j}(i\omega') c_{\sigma' k}^\dagger(i\omega'') c_{\sigma' l}(0) \rangle$$

$$\langle \mathcal{T} A(\tau) B(0) \rangle$$


P. Seth



I. Krivenko



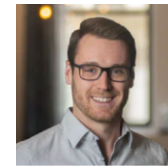
M. Ferrero



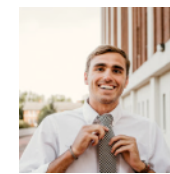
H. Strand



O. Parcollet



A. Hampel



H. LaBollita

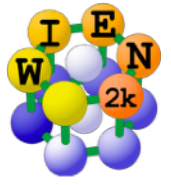


TRIQS Applications — Connection to Electronic Structure

- DFT Tools — Toolbox for Ab-Initio Calculations of Correlated Materials

triqs.github.io/dft_tools

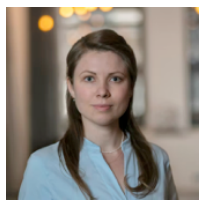
M. Aichhorn et al. CPC '16 ~ 140 Citations



WANNIER90



A. Hampel



S. Beck



M. Aichhorn



L. Pourovskii



V. Vildosola



O. Peil



M. Zingl



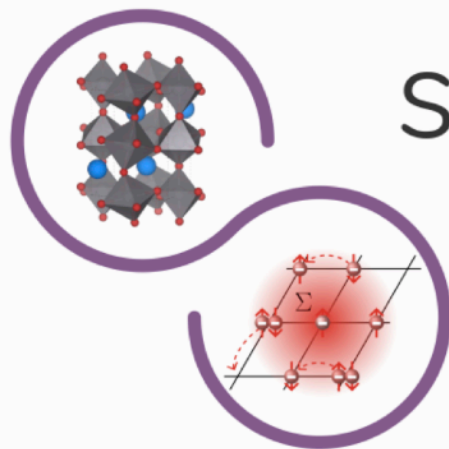
M. Ferrero



G. Kraberger



J. Karp




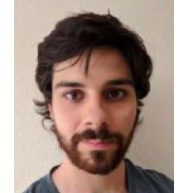
solid_dmft


A versatile python wrapper to perform DFT + DMFT calculations utilizing the TRIQS software library.


triqs.github.io/solid_dmft/

M. Merkel et al. JoSS '22


 A. Hampel


 A. Carta


 S. Beck


 M. Merkel

TRIQS Applications — Vertex Calculations

- TPRF — The Two-particle Response Function Tool Box

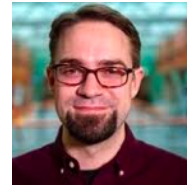
triqs.github.io/tprf



H. Strand



Y. in't Veld



M. Rösner



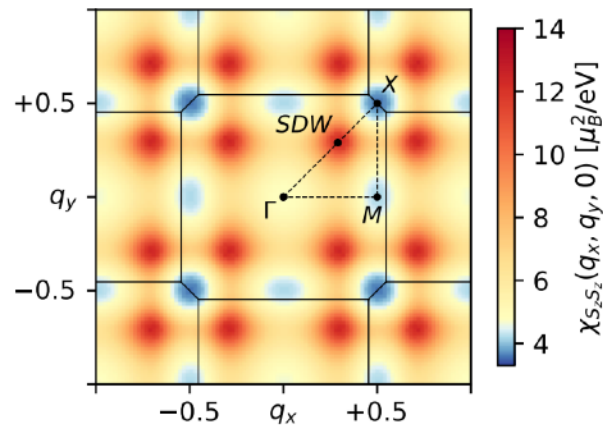
S. Kaeser



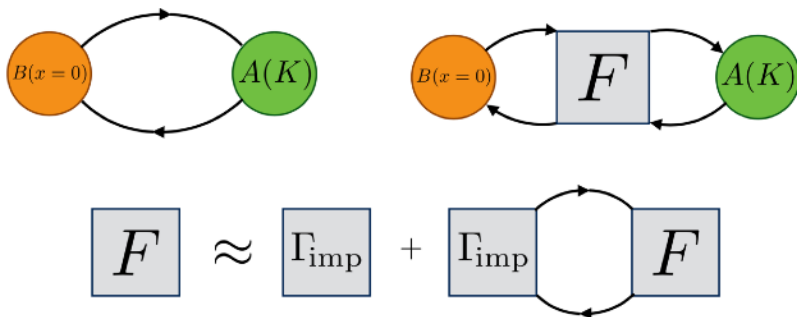
P. Hansmann



E. van Loon



H. Strand et al. PRB '19



- Lindhard Susceptibilities
- Random-phase Approximation
- GW Approximation
- Generalized Susceptibilities
- Bethe-Salpeter Equation Solver
- Vertex-Corrected Lattice Susceptibilities

TRIQS — Packaging

triqs.github.io/triqs/latest/install.html

- Anaconda `conda install -c conda-forge triqs`



Versions 3.3
Soon!

- Debian Packages for Ubuntu 22.04 and 24.04

`apt-get install triqs`



- Binder Notebook triqs.github.io/notebook



- Docker Image `docker pull flatironinstitute/triqs`
`docker run -p 8888:8888 flatironinstitute/triqs`



- Singularity `singularity pull docker://flatironinstitute/triqs`
`singularity exec triqs.sif python myscript.py`



- EasyBuild `eb -r --software-name=TRIQS`



TRIQS — Slack Channel

triqs.github.io/slack

The screenshot shows a Slack interface for the channel #sherbrooke-school-2024. The left sidebar contains navigation options: Home, DMs, Activity, Later, and More. Under 'Channels', the channel #sherbrooke-school-2024 is highlighted with an orange border. The main content area shows the channel name, a search bar, and a 'Canvas' button. Below the channel name, there are options to 'Add a bookmark', 'Add description', 'Add coworkers', and 'Send emails to channel'. A message from Nils Wentzell, dated 3:01 PM, states 'joined #sherbrooke-school-2024.'. The message input field at the bottom contains the text 'Message #sherbrooke-school-2024' and various formatting and action icons.

TRIQS

Search TRIQS

sherbrooke-school-2024

+ Add a bookmark

sherbrooke-school-2024

You created this channel today. This is the very beginning of the # sherbrooke-school-2024 channel.

Add description Add coworkers Send emails to channel

Today

Nils Wentzell 3:01 PM
joined #sherbrooke-school-2024.

Message #sherbrooke-school-2024

TRIQS Documentation

triqs.github.io

🏠 TRIQS
3.2.0

Search docs

- 📄 Welcome
- 📄 Installation
- 📄 Documentation
 - 📄 Manual
 - 📄 C++ API
 - 📄 Python API
 - 📄 triqs.atom_diag
 - 📄 triqs.dos
 - 📄 triqs.fit
 - 📄 triqs.gf
 - 📄 triqs.lattice
 - 📄 triqs.operators
 - 📄 triqs.plot
 - 📄 triqs.random_generator
 - 📄 triqs.stat
 - 📄 triqs.sumk
 - 📄 triqs.utility
- 📄 Applications based on TRIQS
- 📄 User guide
- 📄 Contributing

🏠 » Documentation » triqs.gf » triqs.gf.meshes » triqs.gf.meshes.MeshImFreq

triqs.gf.meshes.MeshImFreq

class triqs.gf.meshes.MeshImFreq

Mesh of Matsubara frequencies

Parameters:

- **beta** (*float*) – Inverse temperature
- **S** (*str*) – Statistic, 'Fermion' or 'Boson'
- **n_iw** (*int [default=1025]*) – Number of positive Matsubara frequencies

Methods

<code>__init__</code> (*args, **kwargs)	Initialize self.
<code>copy</code>	Signature : () -> MeshImFreq Make a copy (clone) of self
<code>copy_from</code>	Signature : (MeshImFreq other) -> None Assignment
<code>first_index</code>	Signature : () -> int
<code>index_to_linear</code>	Signature : (int i) -> int index -> linear index
<code>last_index</code>	Signature : () -> int
<code>positive_only</code>	Signature : () -> bool
<code>set_tail_fit_parameters</code>	Signature : (float tail_fraction, int n_tail_max = 30, std::optional<int> expansion_order = {}) -
<code>values</code>	Signature : () -> PyObject * A numpy array of all the values of the mesh points

TRIQS Documentation

triqs.github.io

TRIQS
3.2.0

Search docs

- Welcome
- Installation
- Documentation
 - Manual
 - C++ API
 - Python API
 - triqs.atom_diag
 - triqs.dos
 - triqs.fit
 - triqs.gf
 - triqs.lattice
 - triqs.operators
 - triqs.plot
 - triqs.random_generator
 - triqs.stat
 - triqs.sumk
 - triqs.utility
- Applications based on TRIQS
- User guide
- Contributing

» Documentation » triqs.gf » triqs.gf.meshes » triqs.gf.meshes.MeshImFreq

triqs.gf.meshes.MeshImFreq

`class triqs.gf.meshes.MeshImFreq`

Mesh of Matsubara frequencies

Parameters:

- `beta` (*float*) – Inverse temperature
- `S` (*str*) – Statistic, 'Fermion' or 'Boson'
- `n_iw` (*int* [*default=1025*]) – Number of positive Matsubara frequencies

Methods

<code>__init__</code> (*args, **kwargs)	Initialize self.
<code>copy</code>	Signature : () -> MeshImFreq Make a copy (clone) of self
<code>copy_from</code>	Signature : (MeshImFreq other) -> None Assignment
<code>first_index</code>	Signature : () -> int
<code>index_to_linear</code>	Signature : (int i) -> int index -> linear index
<code>last_index</code>	Signature : () -> int
<code>positive_only</code>	Signature : () -> bool
<code>set_tail_fit_parameters</code>	Signature : (float tail_fraction, int n_tail_max = 30, std::optional<int> expansion_order = {}) -
<code>values</code>	Signature : () -> PyObject * A numpy array of all the values of the mesh points

TRIQS — Getting Started

jupyter.quantum2024.ccs.usherbrooke.ca

Sign in

Username:

Password:

Server Options

Reservation <input type="text" value="None"/>	Partition <input type="text"/>
Account <input type="text" value="def-lect00"/>	Time (hours) <input type="text" value="5.0"/>
Number of cores <input type="text" value="1"/>	Memory (MB) <input type="text" value="7664"/>

Enable core oversubscription? Recommended for interactive usage

GPU configuration

User interface

TRIQS — Getting Started

jupyter.quantum2024.ccs.usherbrooke.ca

The image shows the JupyterLab Launcher interface. The top menu bar includes File, Edit, View, Run, Kernel, Tabs, Settings, and Help. The left sidebar shows a file browser with a search bar and a table of files:

Name	Last Modified
/	
tutorials	20 hours ago

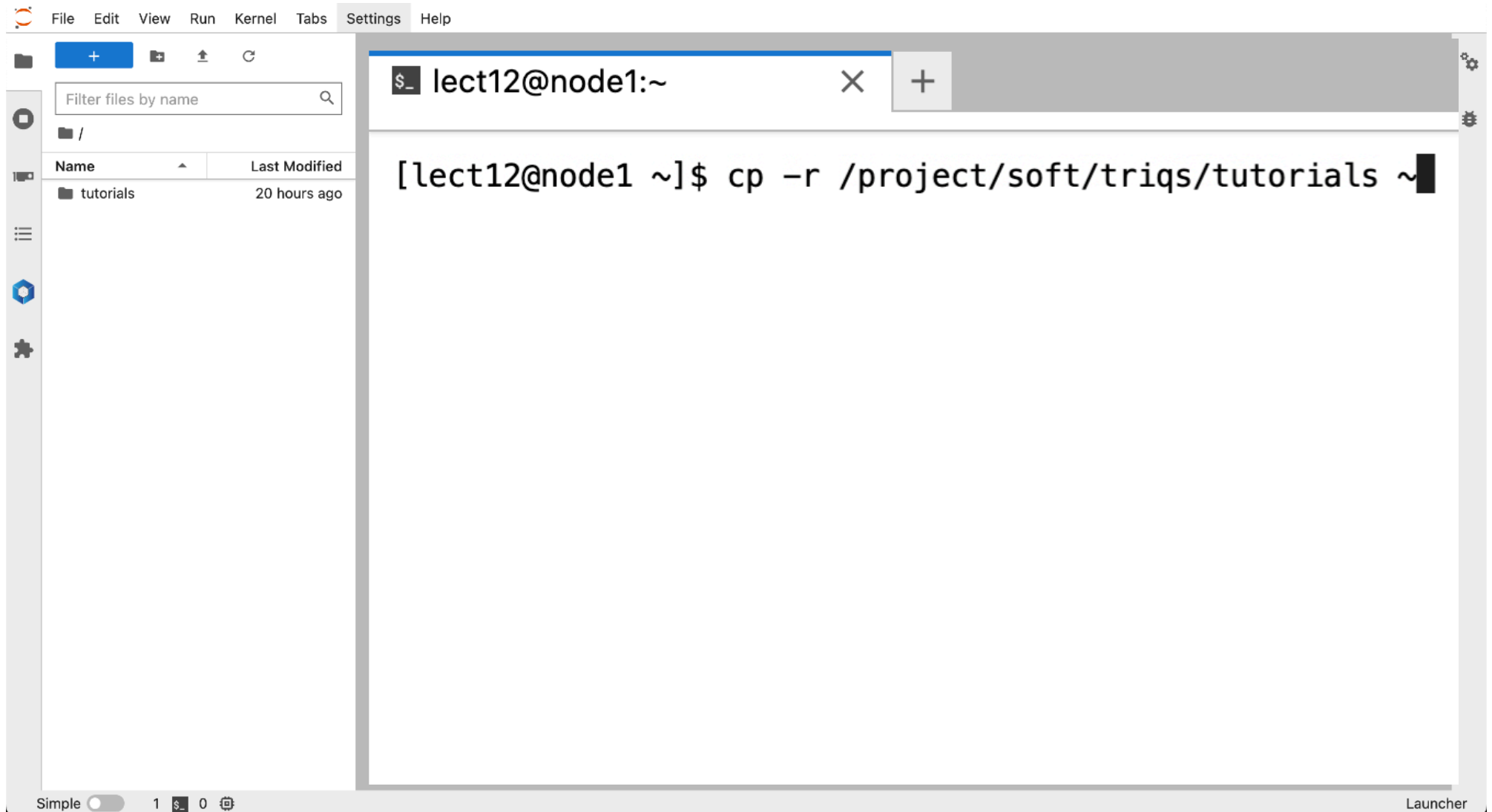
The main area is titled "Launcher" and contains three sections:

- Notebook**: Contains four icons for starting a notebook:
 - Python 3 (ipykernel)
 - Desktop [↗]
 - pyqcm
 - TRIQS 3.3 share
- Console**: Contains three icons for starting a console:
 - Python 3 (ipykernel)
 - pyqcm
 - TRIQS 3.3 share
- Other**: Contains five icons for other actions:
 - Terminal (highlighted with an orange border)
 - Text File
 - Markdown File
 - Python File
 - Show Contextual Help

The bottom status bar shows "Simple" with a toggle, "1" kernel, "\$_" prompt, and "0" files. The bottom right corner says "Launcher".

TRIQS — Getting Started

jupyter.quantum2024.ccs.usherbrooke.ca



The screenshot displays the JupyterLab interface. At the top, a menu bar includes 'File', 'Edit', 'View', 'Run', 'Kernel', 'Tabs', 'Settings', and 'Help'. On the left, a file browser shows a search bar 'Filter files by name' and a directory listing with columns 'Name' and 'Last Modified'. The listing shows a folder named 'tutorials' modified '20 hours ago'. The main area is a terminal window titled 'lect12@node1:~' with a '+' button to add more tabs. The terminal shows the command `[lect12@node1 ~]$ cp -r /project/soft/triqs/tutorials ~` being executed. At the bottom left, there is a 'Simple' toggle switch, a '1' tab indicator, and a '\$' icon. At the bottom right, the word 'Launcher' is visible.

TRIQS — Getting Started

jupyter.quantum2024.ccs.usherbrooke.ca

The screenshot displays the JupyterLab Launcher interface. On the left, a file browser shows a directory structure under "/ tutorials /" with files like "AbinitioDMFT", "Basics", "C++", "ModelDMFT", "TwoParticleResp...", and "README.md", all last modified 16 minutes ago. The main area is titled "Launcher" and is divided into three sections: "Notebook", "Console", and "Other".

Notebook section includes:

- Python 3 (ipykernel)
- Desktop [?]
- pyqcm
- TRIQS 3.3 share

Console section includes:

- Python 3 (ipykernel)
- pyqcm
- TRIQS 3.3 share

Other section includes:

- Terminal
- Text File
- Markdown File
- Python File
- Show Contextual Help

The bottom status bar shows "Simple" mode, a kernel icon, and the text "0 \$ _ 0". The word "Launcher" is visible in the bottom right corner.

TRIQS — Getting Started

jupyter.quantum2024.ccs.usherbrooke.ca

File Edit View Run Kernel Tabs Settings Help

Filter files by name

/ tutorials / Basics /

Name	Last Modified
solutions	23 minutes ago
00a-Introducing_th...	23 minutes ago
00b-Matplotlib_Exa...	23 minutes ago
01-Greens_function...	23 minutes ago
02-Archiving_your_...	23 minutes ago
03-Operators.ipynb	23 minutes ago
04-Multivariable_Gr...	23 minutes ago
my_archive.h5	40 minutes ago
sample.dat	2 hours ago

Launcher 01-Greens_functions.ipynb

TRIQS 3.3 share

TRIQS Green's functions

It is now time to start using some of the tools provided by TRIQS.

Much of the functionality in TRIQS, while implemented in C++ for optimal performance, is exposed through a Python interface to make it easier to use. From a practical point of view this means that you can think of TRIQS as a python library, just like numpy or matplotlib.

One of the central objects of a many-body calculation is a Green's function. Green's functions in TRIQS are functions defined on a mesh \mathcal{M} of points that hold values in some domain \mathcal{D} , for example $\mathbb{C}^{2 \times 2}$

$$G : \mathcal{M} \rightarrow \mathcal{D}$$

A few common Green's function meshes in TRIQS include:

- `MeshReFreq` - Real-frequencies equally spaced in $[\omega_{min}, \omega_{max}]$
- `MeshImFreq` - Matsubara Frequencies
- `MeshImTime` - Imaginary time points equally spaced in $[0, \beta]$
- `MeshReTime` - Real-time points (not covered in this tutorial)

Let's see how we can **construct a Mesh and print its values**.

```
[ ]: # Import the Mesh type we want to use
from triqs.gf import MeshImTime

# The documentation tells us which parameters we need to pass for the mesh construction
?MeshImTime

[ ]: # Provide the inverse temperature, Statistic, and number of points
tau_mesh = MeshImTime(beta=5, statistic='Fermion', n_tau=11)
```

Simple 0 \$ 1 TRIQS 3.3 share | Idle Mode: Command Ln 1, Col 1 01-Greens_functions.ipynb

TRIQS Tutorials Summary

- Notebook Server

jupyter.quantum2024.ccs.usherbrooke.ca

- TRIQS Tutorials

```
cp -r /project/soft/triqs/tutorials ~
```

- TRIQS Slack Workspace

triqs.github.io/slack

- TRIQS Documentation

triqs.github.io