SMRT: A new, modular snow microwave radiative transfer model

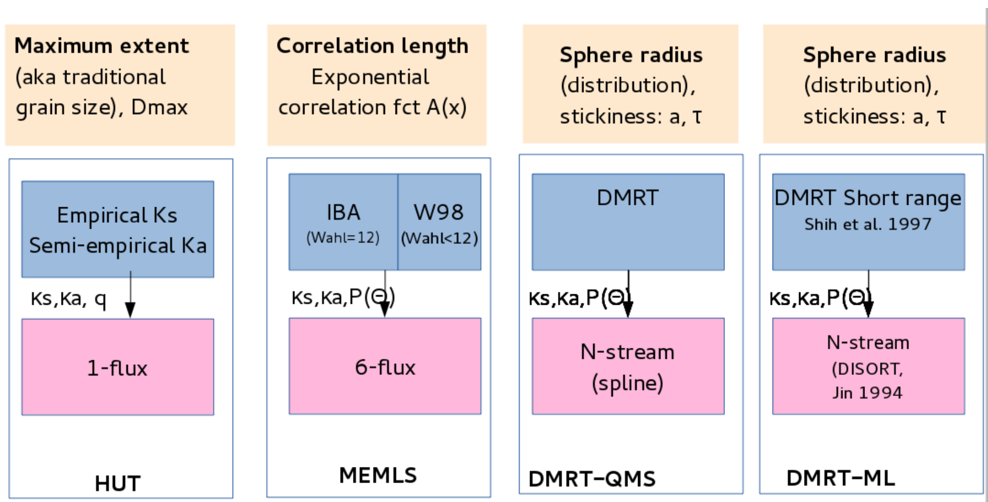
Outline

- Overview
- Implementation and structure
- Microstructure
- Model intercomparison via wrappers
- Github / website / help / documentation
- Summary

SMRT: Context

~~Are~~ existing snow microwave emission models aren't so different ~~?~~

Löwe and Picard (TC, 2015)
Pan et al. (TGRS, 2016)



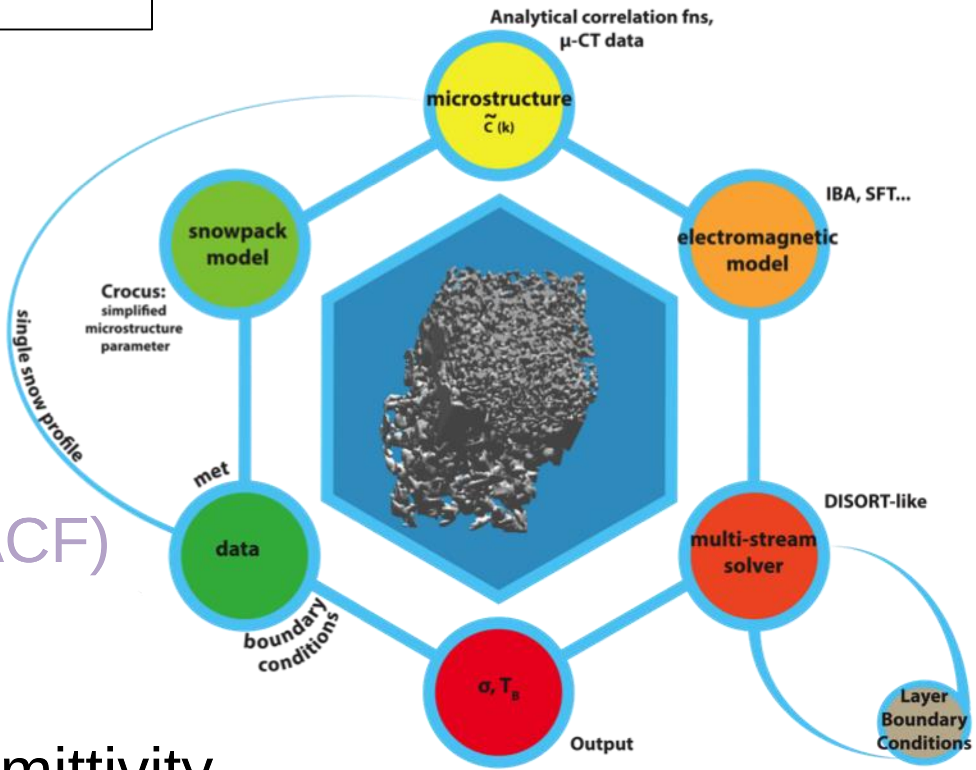
→ the different **electromagnetic theories**
→ the different **micro-structure representation** used by these models
→ the different **solution** of the radiative transfer equation

SMRT: Overview

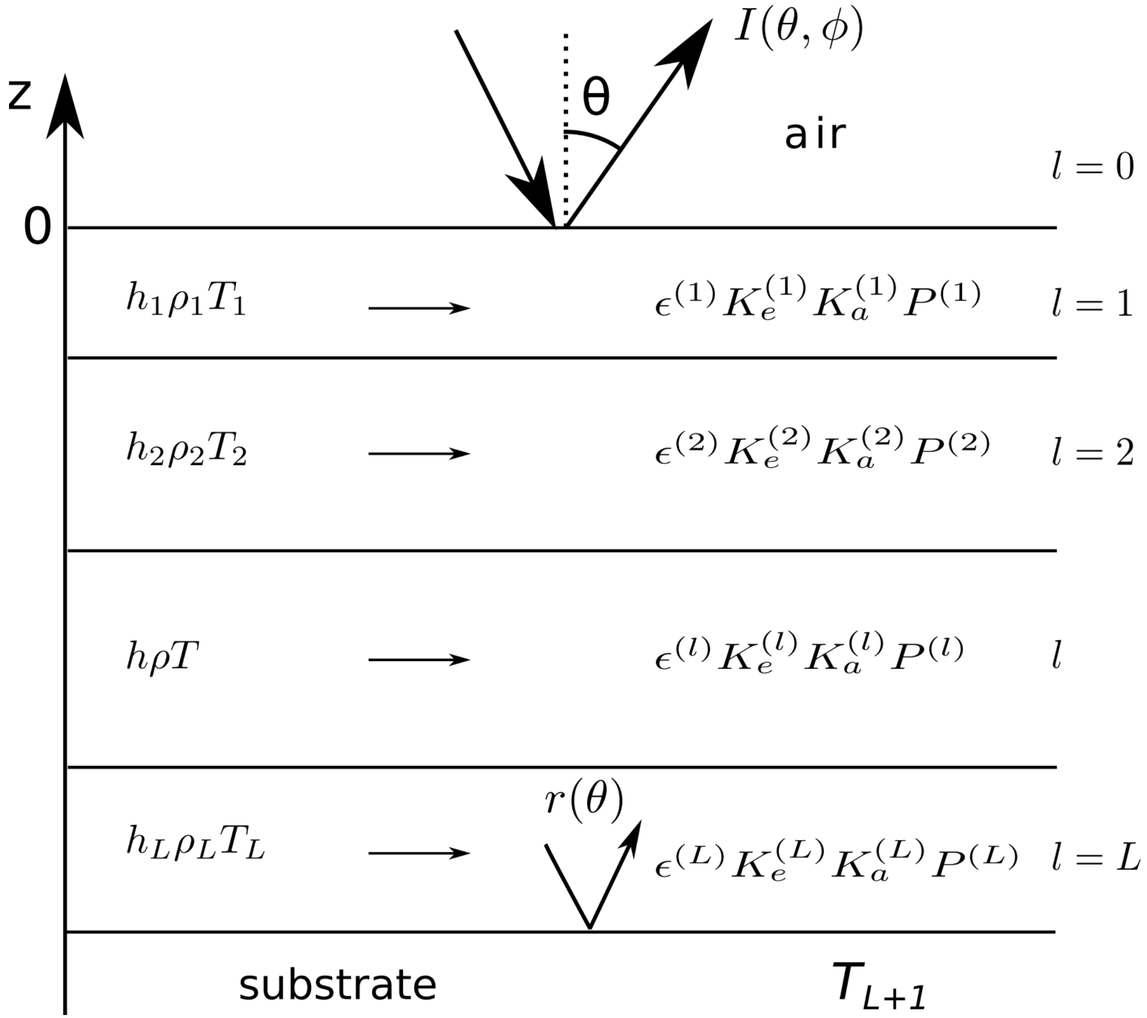
MICROSTRUCTURAL ORIGIN OF ELECTROMAGNETIC SIGNATURES IN MICROWAVE REMOTE SENSING OF SNOW

Main physics:

- Active / passive
- Multilayer
- Phase function:
  EM Theory
- Layer microstructure:
  autocorrelation functions (ACF)
- RT Solver
- Interface: Fresnel
- Substrate / atmosphere / permittivity



SMRT: Overview

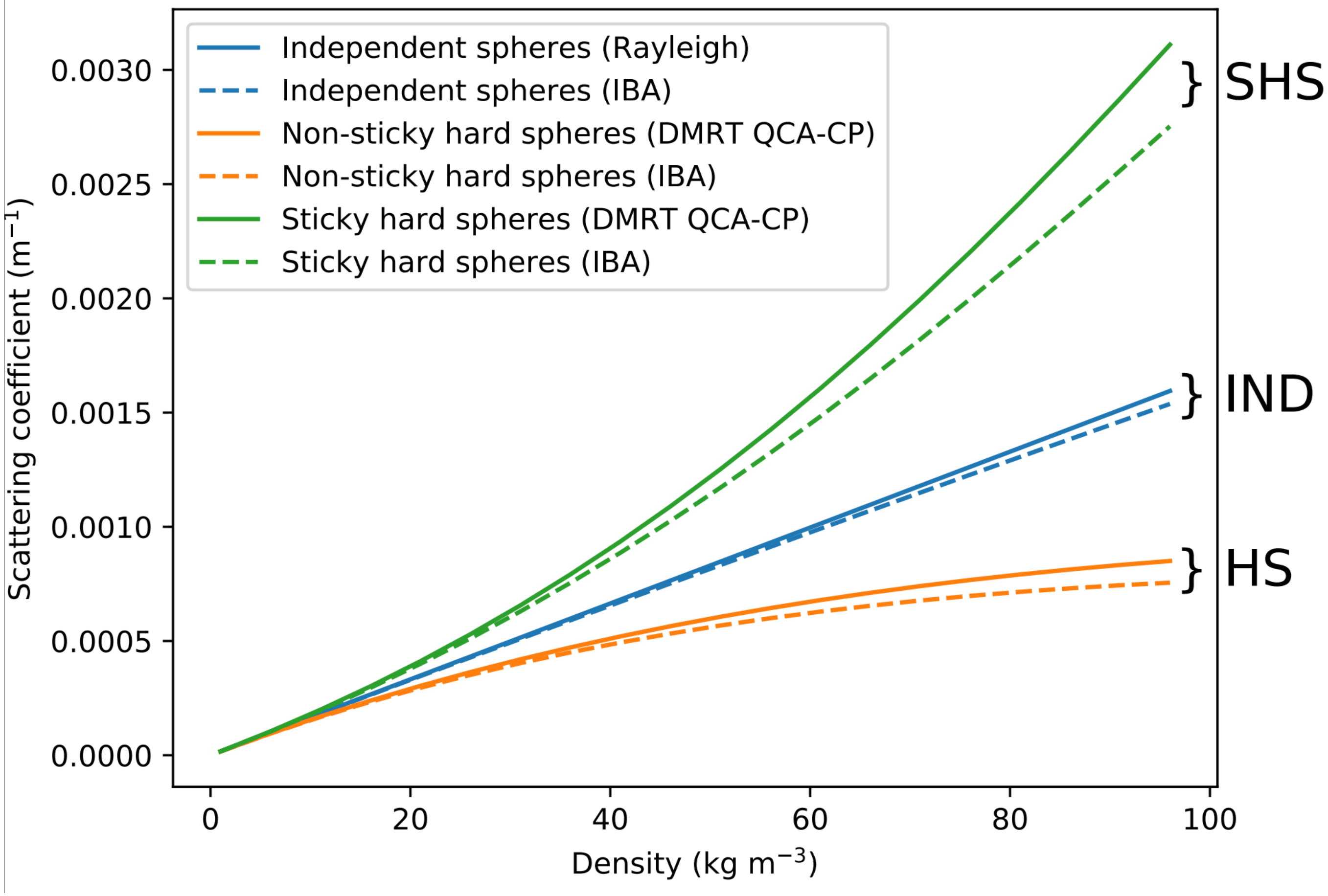


Picard et al., submitted to GMD

Switching modules….
- Microstructure
- EM model
- Interfaces
- Substrate
- Atmosphere
- RT Solver

SMRT: Overview

...makes model intercomparison easy!



Picard et al., submitted to GMD

Why python?

- open, object oriented (to realize “easy to use”)

Imported modules hide computational complexity

Example of a model run:

Inputs

Create snowpack

Choose sensor config

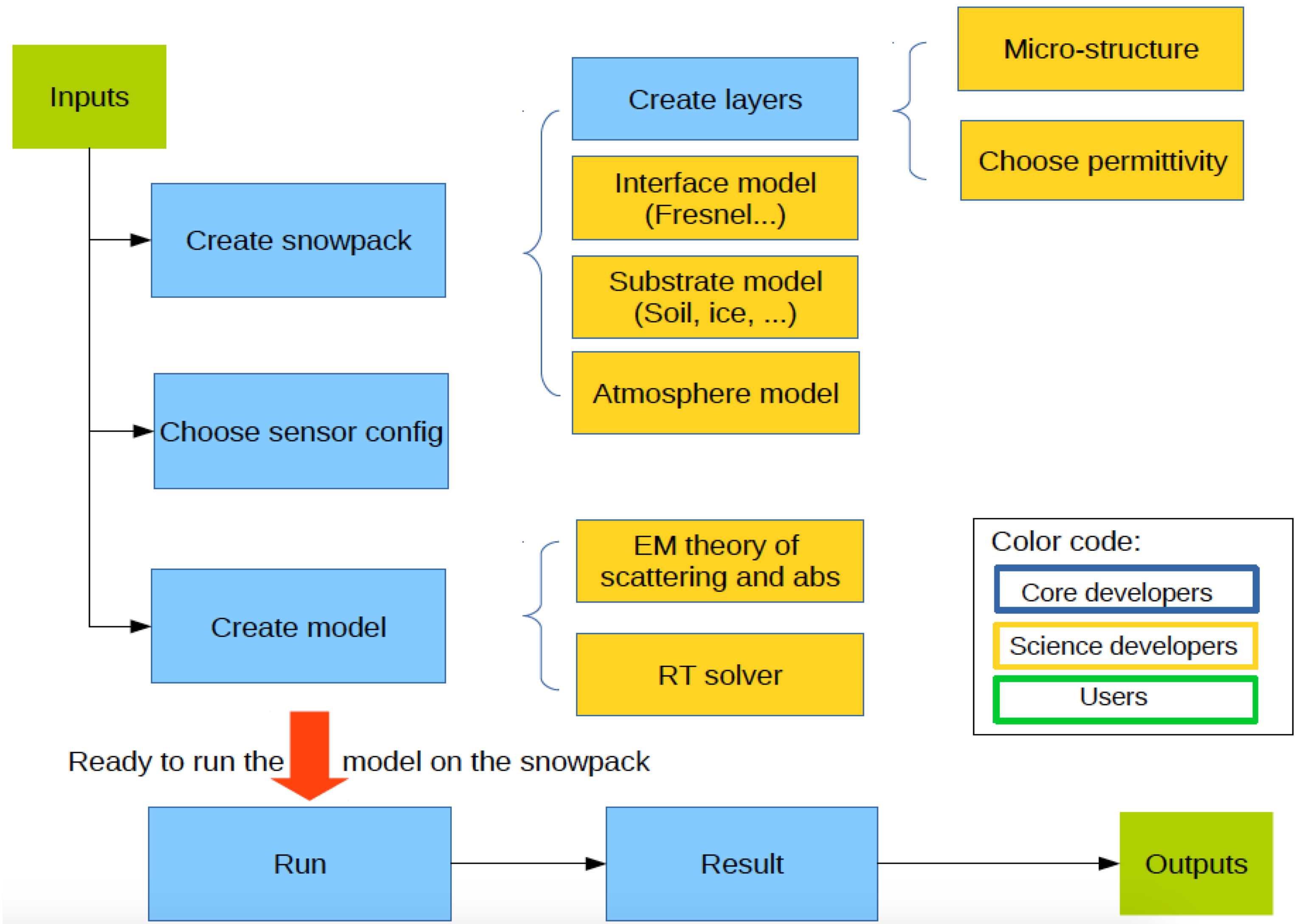
Create model

Run

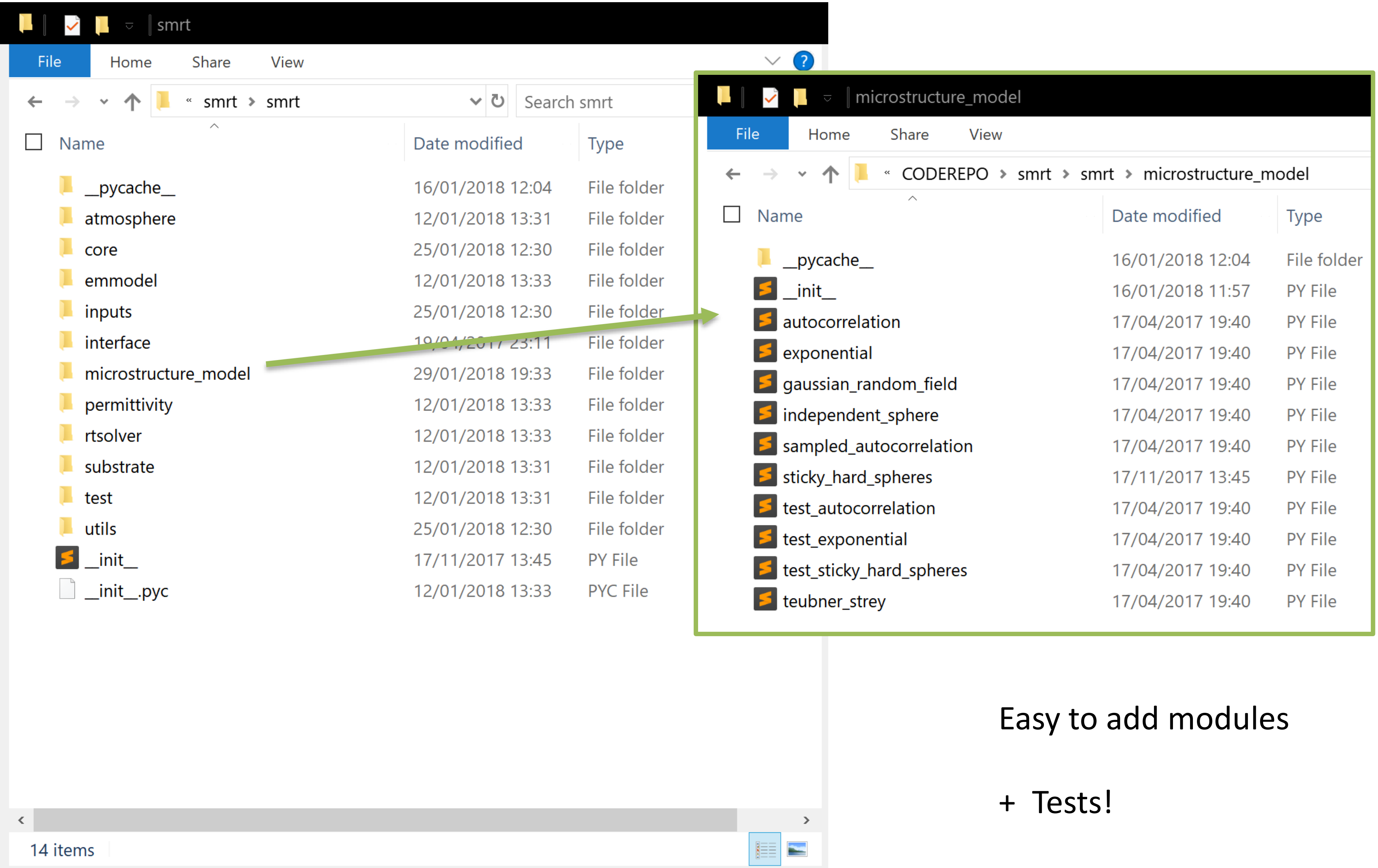
Outputs

```
from smrt import make_snowpack, make_model, sensor

# prepare inputs
thickness = [100]
corr_length = [5e-5]
temperature = [270]
density = [320]

# create the snowpack
snowpack = make_snowpack(thickness=thickness,
                         microstructure_model="exponential",
                         density=density,
                         temperature=temperature,
                         corr_length=corr_length)

# create the sensor
radiometer = sensor.amsre('37V')

# create the model
m = make_model("iba", "dort")

# run the model
result = m.run(radiometer, snowpack)

# outputs
print(result.TbV(), result.TbH())
```

SMRT: Structure



SMRT: Structure



Easy to add modules

+ Tests!

SMRT: Microstructure

Layer microstructure:

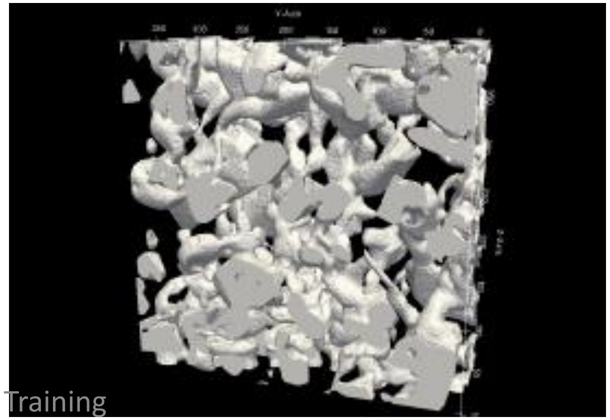
- represented by a two-point correlation *function*

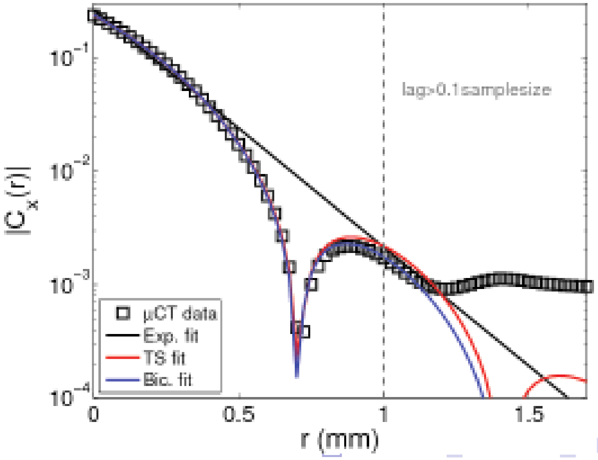
Implemented models: (and reasons for them)

- Exponential model (MEMLS)
- Sticky hard spheres (DMRT-ML, DMRT-QMS)
- Independent sphere (classic)
- Teubner–Strey (empirical evidence)
- (Level-cut) Gaussian random fields (full-field methods)

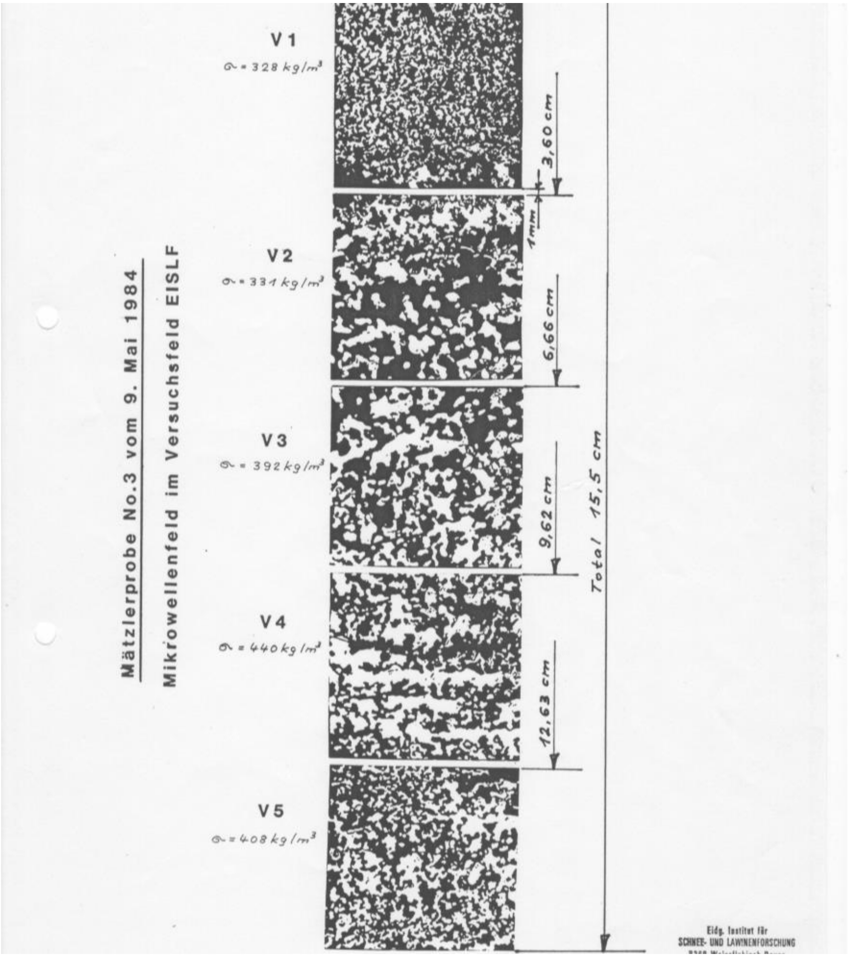
Why different choices?

- To be flexible with non-obvious correlation functions:





Micro-CT is not the only way!



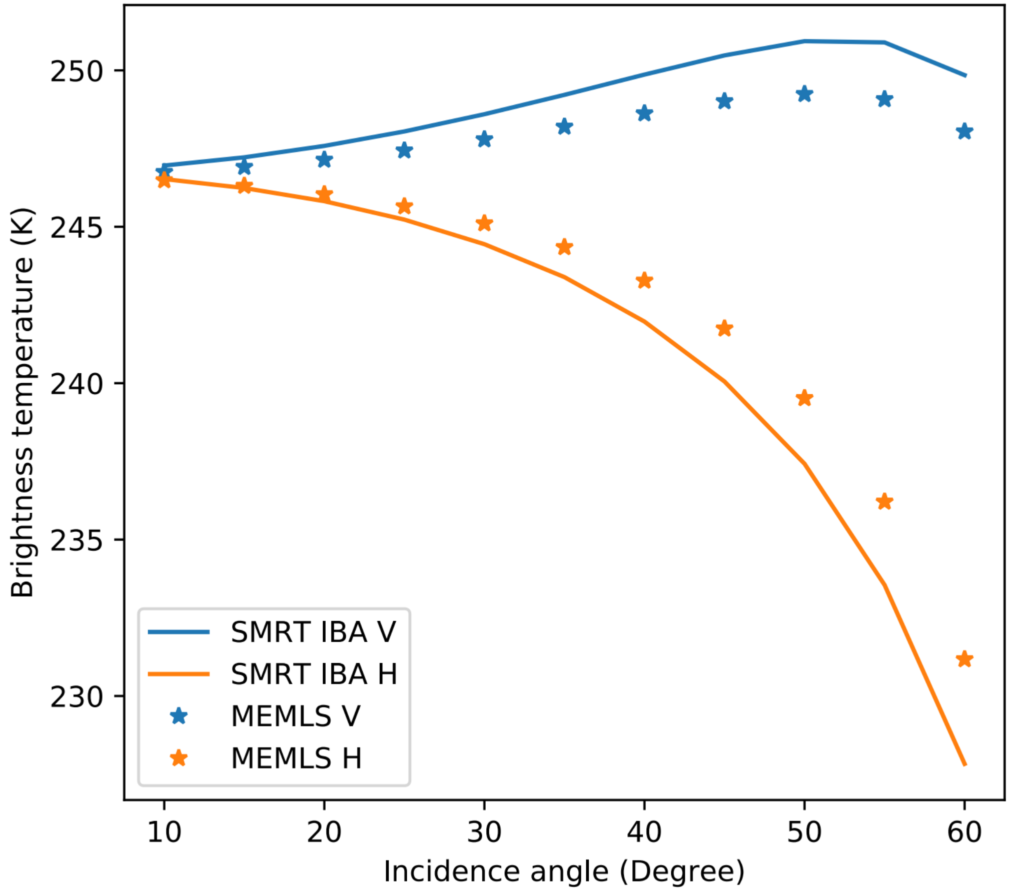
SMRT: Legacy

To facilitate model inter-comparison:

- Shallow wrappers for MEMLS, HUT, DMRT-QMS (no code)

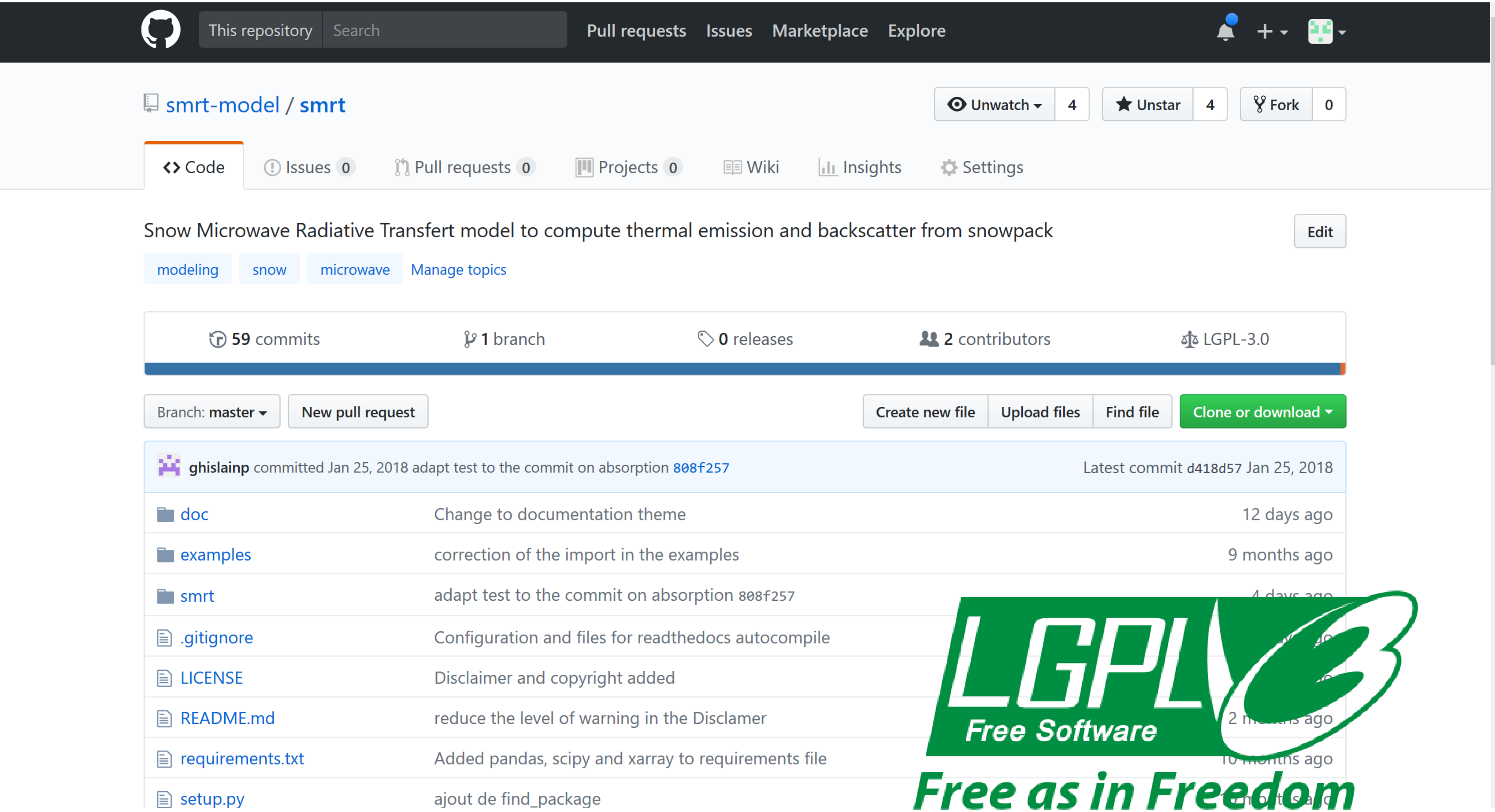
```
# general import for smrt
from smrt import make_snowpack, make_model, sensor

# import for memls
from smrt.utils import memls_legacy

# prepare snowpack
pc = 0.2e-3
snowpack = make_snowpack(thickness=[10], microstructure_model="exponential",
                         density=[300], temperature=[265], corr_length=pc)

# create the sensor
theta = range(10, 80, 5)
radiometer = sensor.passive(37e9, theta)

# create the EM Model
m = make_model("iba", "dort")

# run the model
sresult = m.run(radiometer, snowpack)

# run MEMLS matlab code
mresult = memls_legacy.run(radiometer, snowpack)

# outputs
plt.plot(theta, sresult.TbV(), 'r-', label='SMRT V')
plt.plot(theta, sresult.TbH(), 'r--', label='SMRT H')
```

SMRT: intercomparison



Picard et al., submitted to GMD

https://github.com/smrt-model/smrt



git pull

https://www.smrt-model.science

SMRT Home Getting Started Documentation Contribute

SMRT: Snow Microwave Radiative Transfer model

SMRT is an **active / passive microwave radiative transfer model for multilayer snow** written in Python. It was developed with European Space Agency support in order to investigate the representation of the snow microstructure, the main driver of scattering.

SMRT is modular, so allows easy intercomparisons between different modelling approaches in a plug-and-play way. SMRT proposes different electromagnetic theories to compute scattering (DMRT, IBA, Rayleigh independent, ...). In the case of IBA, different microstructure representations can be used (Sticky Hard Spheres, Exponential, Gaussian random field, ...). The current version proposes only one radiative transfer solver (DORT) but this can be extended. Last but not least, wrappers are included to run MEMLS, HUT and DMRT-QMS models (in their original matlab code) from within SMRT. Whilst there is plenty to get started with, there are more theoretical advances that can be made. SMRT is intended to be a community model - all are welcome to use it, and to contribute to its development!

SMRT uses the lastest python version (e.g. 3.6) but also works with earlier versions (2.7, 3.4 and higher). The code is open source and is hosted on github.

Getting started with SMRT

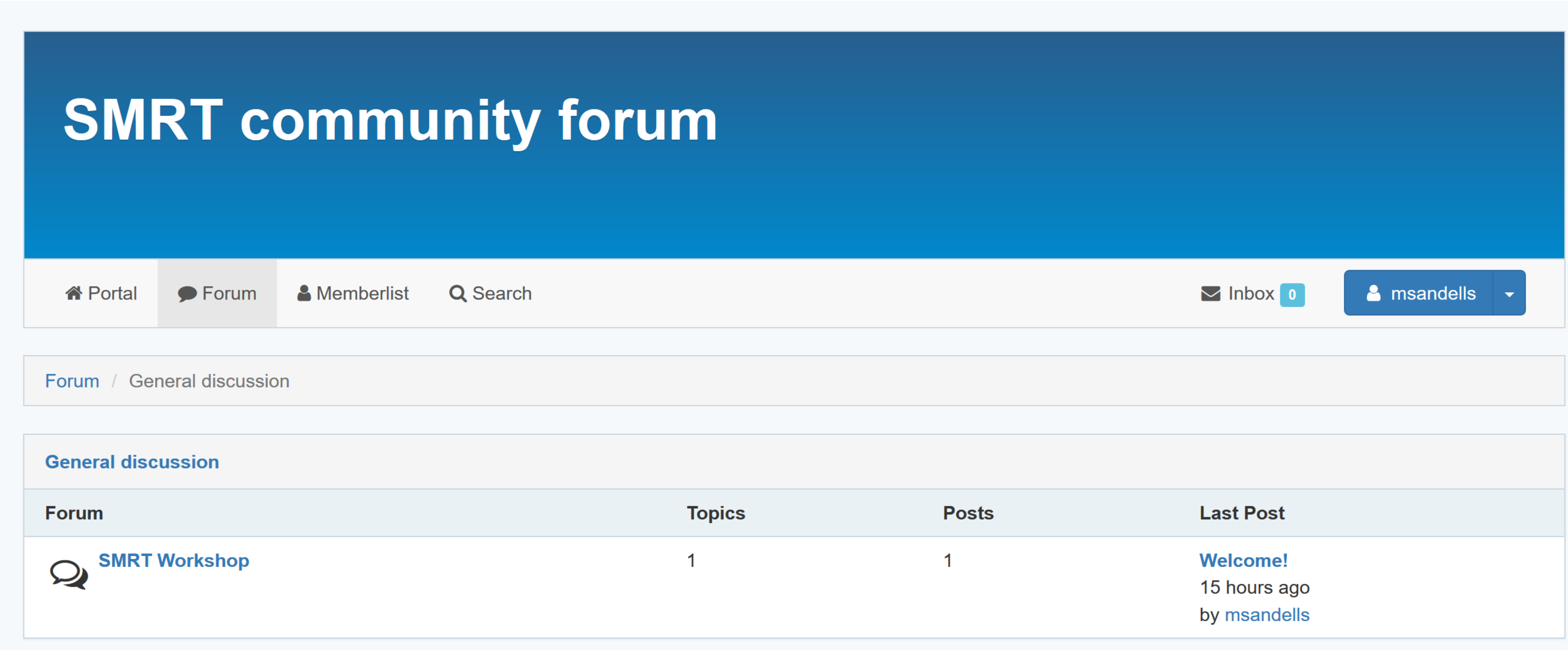
Using SMRT is easy

You must write a (short) driver code, SMRT is a library that your driver code call to perform a calculation. This is simple, there are four main steps in a typical driver code:

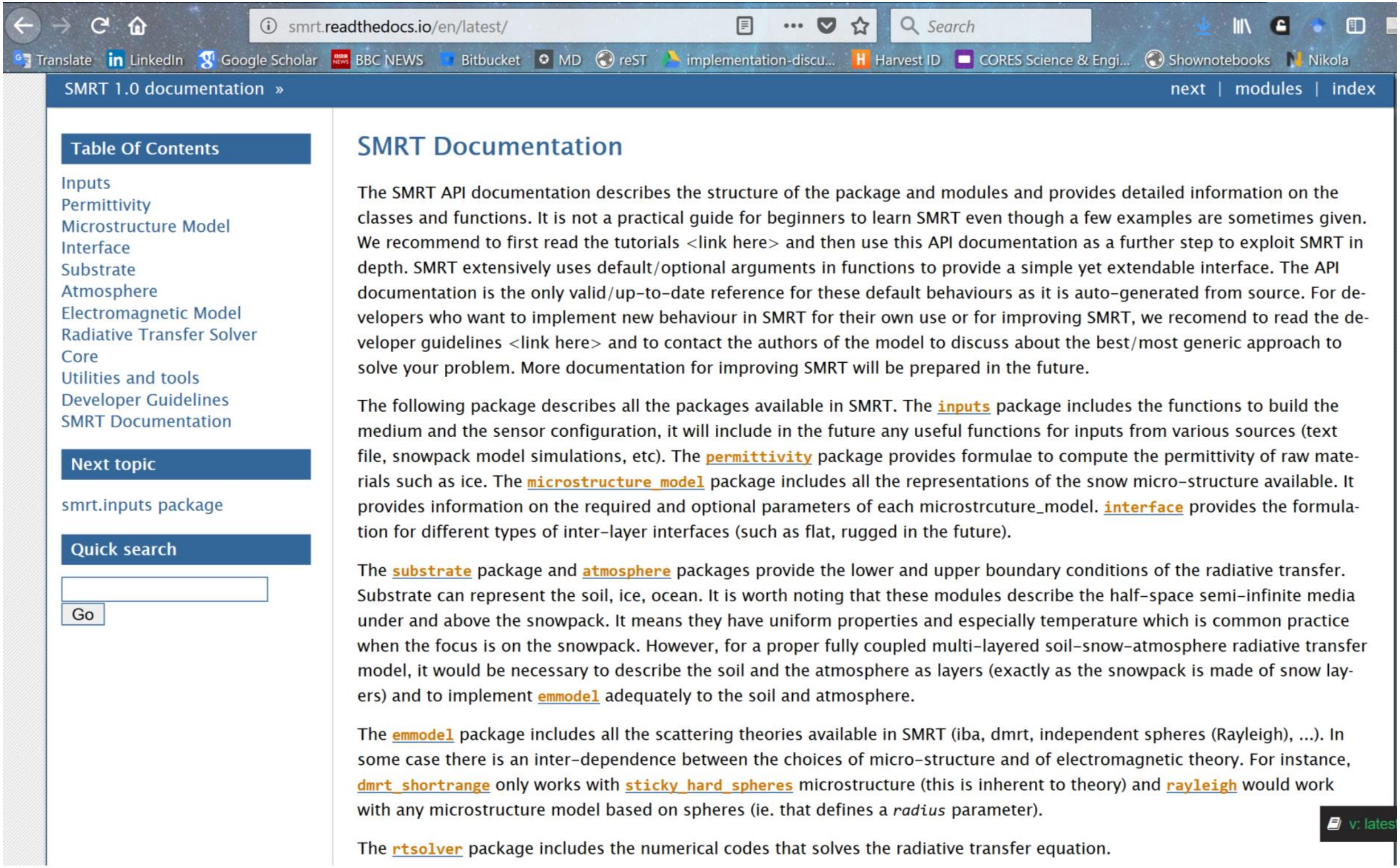
1. construct a snowpack (either from field data or snowpack model).

```
snowpack = make_snowpack(thickness=10.,   # 10 m deep snowpack
                         microstructure_model="sticky_hard_spheres",
                         density=320.0,   # 320 kg/m3
                         temperature=260, # 260 K
                         radius=100e-6)   # 100 microns
```

http://community.smrt-model.science



Online documentation



smrt.readthedocs.io/en/latest/

Translate · LinkedIn · Google Scholar · BBC NEWS · Bitbucket · MD · reST · implementation-discu... · Harvest ID · CORES Science & Engi... · Shownotebooks · Nikola

SMRT 1.0 documentation » next | modules | index

Table Of Contents

- Inputs
- Permittivity
- Microstructure Model
- Interface
- Substrate
- Atmosphere
- Electromagnetic Model
- Radiative Transfer Solver
- Core
- Utilities and tools
- Developer Guidelines
- SMRT Documentation

Next topic

smrt.inputs package

Quick search

Go

SMRT Documentation

The SMRT API documentation describes the structure of the package and modules and provides detailed information on the classes and functions. It is not a practical guide for beginners to learn SMRT even though a few examples are sometimes given. We recommend to first read the tutorials <link here> and then use this API documentation as a further step to exploit SMRT in depth. SMRT extensively uses default/optional arguments in functions to provide a simple yet extendable interface. The API documentation is the only valid/up-to-date reference for these default behaviours as it is auto-generated from source. For developers who want to implement new behaviour in SMRT for their own use or for improving SMRT, we recomend to read the developer guidelines <link here> and to contact the authors of the model to discuss about the best/most generic approach to solve your problem. More documentation for improving SMRT will be prepared in the future.

The following package describes all the packages available in SMRT. The `inputs` package includes the functions to build the medium and the sensor configuration, it will include in the future any useful functions for inputs from various sources (text file, snowpack model simulations, etc). The `permittivity` package provides formulae to compute the permittivity of raw materials such as ice. The `microstructure_model` package includes all the representations of the snow micro-structure available. It provides information on the required and optional parameters of each microstrcuture_model. `interface` provides the formulation for different types of inter-layer interfaces (such as flat, rugged in the future).

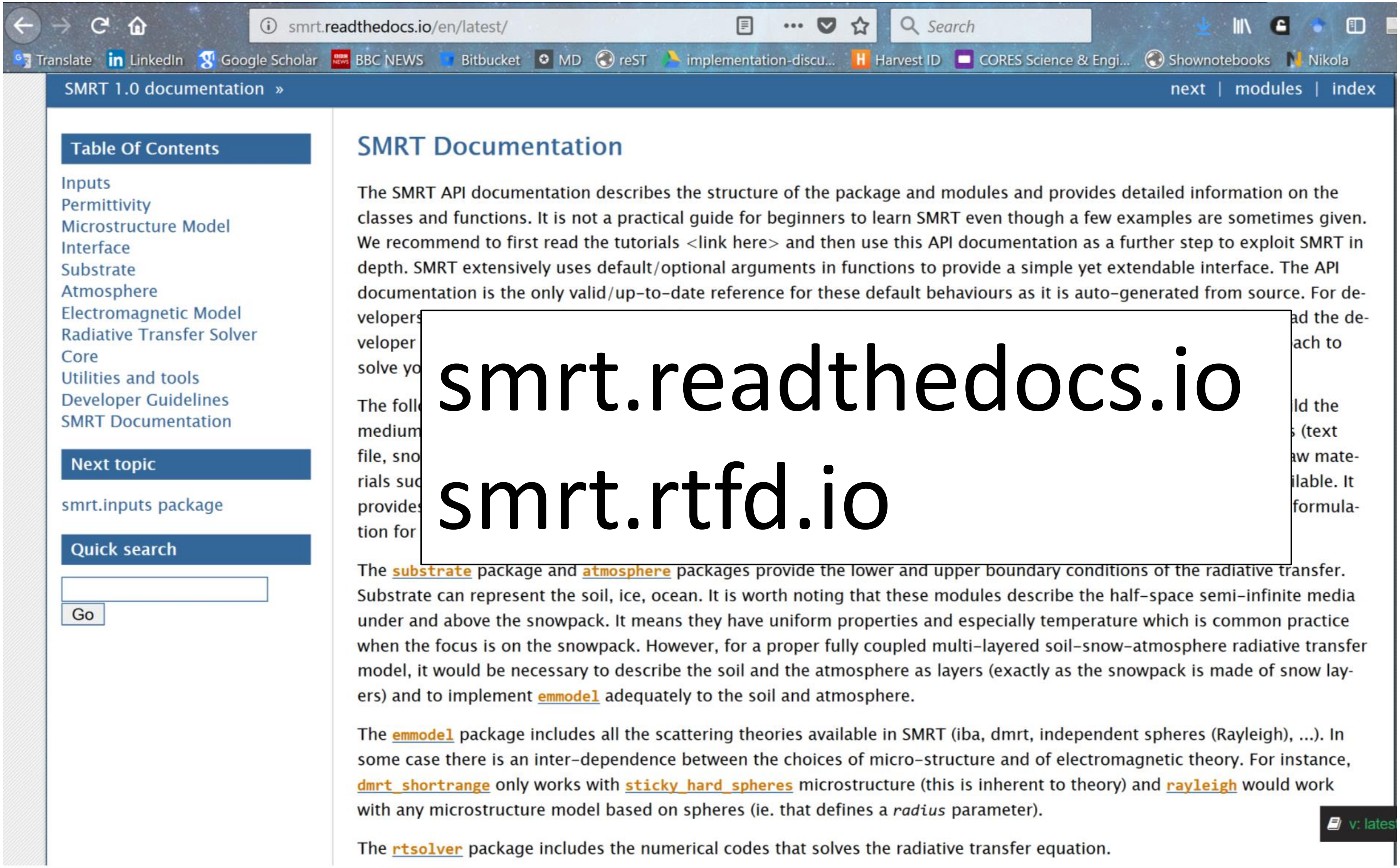
The `substrate` package and `atmosphere` packages provide the lower and upper boundary conditions of the radiative transfer. Substrate can represent the soil, ice, ocean. It is worth noting that these modules describe the half-space semi-infinite media under and above the snowpack. It means they have uniform properties and especially temperature which is common practice when the focus is on the snowpack. However, for a proper fully coupled multi-layered soil-snow-atmosphere radiative transfer model, it would be necessary to describe the soil and the atmosphere as layers (exactly as the snowpack is made of snow layers) and to implement `emmodel` adequately to the soil and atmosphere.

The `emmodel` package includes all the scattering theories available in SMRT (iba, dmrt, independent spheres (Rayleigh), ...). In some case there is an inter-dependence between the choices of micro-structure and of electromagnetic theory. For instance, `dmrt_shortrange` only works with `sticky_hard_spheres` microstructure (this is inherent to theory) and `rayleigh` would work with any microstructure model based on spheres (ie. that defines a *radius* parameter).

The `rtsolver` package includes the numerical codes that solves the radiative transfer equation.

v: latest

Online documentation



smrt.readthedocs.io

smrt.rtfd.io

SMRT

- Why a new model ?
We don't need a new model (yet) but we need:
a repository of microwave community knowledge
= merge all RT models / theories in one code base, one framework
with extended capabilities to explore the micro-structure
with multi mode capabilities (passive, radar, altimeter)
with easier access for beginners and non-specialists
using modern and more efficient languages and programming techniques

SMRT: flexibility, python, git, forum, documentation

We hope SMRT will be taken up by the community as a **joint** endeavour

https://www.smrt-model.science
https://github.com/smrt-model/smrt

http://community.smrt-model.science
smrt.readthedocs.io