

# Project **INF** in the CRC **CIM**

**Particles**



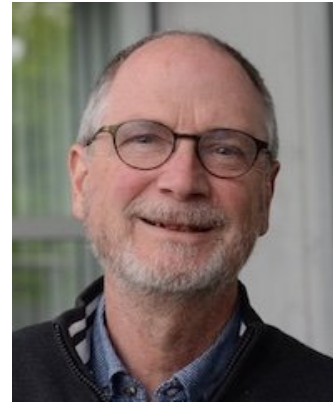
**Kevin Kröniger**  
TUD

**Astro**



**Angus Wright**  
RUB

**Simulations**



**Rainer Grauer**  
RUB

**Data Steward**



**Andreas Schramm**  
IT.Services RUB

# Spring 2022

## Available in the very near future

- GitLab for the CIM
- Compute cluster
- Data server

# Now 2023

## Available and could be used

- GitLab for the CIM
- Compute cluster
- Data server

## ■ GitLab for the CIM

- Where to find it ?** <https://gitlab.ruhr-uni-bochum.de/sfb1491>
- How to register?** Send an email request to me ([grauer@tp1.rub.de](mailto:grauer@tp1.rub.de)) or to **Andreas Schramm** ([Andreas.Schramm@ruhr-uni-bochum.de](mailto:Andreas.Schramm@ruhr-uni-bochum.de))
- What is it good for?** Documentation on how to reproduce a paper (DFG guidelines):  
paper, source codes, pre- and post processing scripts  
Must be approved by another member of another project !

Each project is different, thus no general applicable procedure.

Let's have a look at examples ...

## ■ GitLab for the CIM











 **SFB1491**   
Group ID: 2084  [Leave group](#)

Cosmic Interacting Matters - From Source to Signal









**Subgroups and projects**

Shared projects





Archived projects

- >   **CompSchool2023**  Maintainer  we discuss later
- >   **Part-F**   
Science area F - Fundamental properties of matter
- >   **Part-A**   
Science area A - Astrophysical signatures of cosmic-ray transport and interaction



## ■ GitLab for the CIM

- P **Part-A**   
Science area A - Astrophysical signatures of cosmic-ray transport and interaction
- A **A1**   
Multimessenger signatures of Galactic cosmic-ray transport
- A **A2**   
Cosmic-ray signatures in dwarf galaxies: astrophysical foreground, dark-matter background
- > ○ A **A3**   
Cosmic-ray transport in the transition region from Galactic to extragalactic origin
- A **A4**   
Magnetohydrodynamical halos of starforming galaxies
- A **A5**   
Disentangling cosmic-ray signatures in AGN-starburst composites
- A **A6**   
Multimessenger signatures of tidal disruption events
- A **A7**   
Density-dependence of the temporal structure in the multimessenger spectrum of blazars

## ■ GitLab for the CIM



- **P** **Part-F**   
Science area F - Fundamental properties of matter
  - > ○ **F** **F1**   
Propagation of fast charged particles in artificially generated MHD turbulence: implications for
  - **F** **F2**   
Particle acceleration via magnetic reconnection in different regimes
  - **F** **F3**   
Prompt lepton production in hadronic interactions
  - **F** **F4**   
Cross sections and hadronic interactions in particle- and astroparticle physics
  - **F** **F5**   
Dark-matter searches with gamma-ray telescopes and wide-field cosmological surveys
  - **F** **F6**   
Dark matter and gas galaxies
- > ○ **P** **Part-A**   
Science area A - Astrophysical signatures of cosmic-ray transport and interaction

## ■ GitLab for the CIM

**A** **A1**   
Group ID: 3902 

Multimessenger signatures of Galactic cosmic-ray transport



**Subgroups and projects** Shared projects Archived projects





**A** **A3**   
Group ID: 3904 

Cosmic-ray transport in the transition region from Galactic to extragalactic origin

**Subgroups and projects** Shared projects Archived projects

 **A** **Aerdker et al. 2023** 

▼  **F** **F1**   
Propagation of fast charged particles in artificially generated MHD turbulence: implications for cosmic-ray transport

	<b>L</b> <b>Luebke et al 2023</b> 	★ 0
	<b>W</b> <b>Walter et al 2023</b> 	★ 0

Presently, 4 papers are approved in GitLab. Needs to be improved !

## ■ GitLab for the CIM

Example



**Walter et al 2023** 

Project ID: 2542 

 Doc

Upload New File

 Papers

Upload New File

 Source

Upload New File

 README.md

Update README.md





# ■ GitLab for the CIM

## Walter et al 2022

---

Documentation and Code for the Paper Walter et al. 2022 <https://pubs.aip.org/aip/pop/article-abstract/29/7/072302/2844126/A-nonlinear-model-of-diffusive-particle>

Authors: Dominik Walter, Frederic Effenberger, Horst Fichtner, Yuri Litvinenko

This repo contains a documentation for the whole numerical Setup, the purpose and function of the separated codes and theirs distinct outputs. The details of the codes itself, will be discussed in different files, referenced here.

The main code is a variation of the open access fortran-code VLUGR3, named cartesian\_ref\_nonlin.f Here all the numerical solutions used in the paper are produced, it calculates the shock acceleration models for linear and nonlinear cases. The code produces an output in the form of a sol.dat file, where it gives the output-times as well as the solution at the points x,y,z,numsol. More details for the setup in this code, are given in "DocumentationMainCode".

The file exaktInt.f can take the file sol.dat and puts out several files, containing the spectrum at the shock for different points in time in the form of p,solexap^2,r,t,solnump^2. More details in "DocumentationexaktCode"

The file Spatial.f takes the solution provided by the main code and derives a integrated spatial distribution, by sorting and integrating the output. More details can be found in "DocumentationSpatial".

The file Contour.f takes the solution provided by the main code and provides the resulting diffusion-coefficient and a file that is suitable for 2D plots.

The files derived by exaktInt.f or Spatial.f can now easily be plotted by a plotting routine of choice ( e.g. gunplot).




In the file 'InstructionsforCompiling' you can find a short example, that shows how to compile and run the codes. Using the gfortran-compiler no additional compiler options were necessary, to compile the code.





A Link to the original VLUGR3 code can be found here: <https://dl.acm.org/doi/abs/10.1145/232826.232853>







A file diff.txt was edited, that compares the probia.f example of the official main code (link above) to our version of the code, hihghlighting the differences.

## ■ GitLab for the CIM

### Another example

 **Aerdker et al. 2023**   
Project ID: 2008 

 49 Commits  1 Branch  0 Tags  5.5 MiB Project Storage

Name	Last commit
 DSA_JupyterNotebooks	Fix file name in comment.
 VLUGBCODE	Restructuring of directories and typo corrections
 VLUGBDATA	Merge branch 'main' into cleanup
 VLUGBDOCUMENTATION	Update vlugr3 documentation
 .gitignore	Typos, additional comments, etc.
 README.md	Update vlugr3 documentation

# ■ GitLab for the CIM

## 1. Installation of CRPropa

Install CRPropa version (2777730f094640d09202e5cac65c50a0467bcd48), which can be found in this separate repository:

<https://github.com/sophieaerdker/CRPropa3/tree/DSAEnergyDependence>. Detailed information on how to install CRPropa can be found on the webpage (crpropa.desy.de) and on the public GitHub repository (github.com/crpropa/crpropa3).

## 2. Installation of VLUGR3

Install a Fortran compiler version (90+), which is suitable for your system. The used programs are stored in (./VLUGCODE/). A detailed documentation together with explanations on how to run them can be found in (./VLUGDOCUMENTATION). In this way the already produced data (./VLUGDATA) can be verified.

## 3. Running Simulation and Analysis

Run the jupyter notebooks. The notebooks themselves contain further explanations. Jupyter notebooks are independent of each other. To follow the publication we recommend the following order:

1. StationarySolution.ipynb
2. ParticleSplitting.ipynb
3. Constraints\_AdvectiveStep.ipynb (+ AdvectiveStep\_Steering.py)
4. Constraints\_DiffusiveStep.ipynb (+ DiffusiveStep\_Steering.py)
5. BurstLikeInjection.ipynb
6. EnergyDependentDiffusion.ipynb
7. InjectedSpectra.ipynb (+ InjectedSpectra\_Steering.py)
8. ObliqueShock.ipynb
9. RadialShock.ipynb (+ RadialShock\_Steering.py)



## ■ GitLab for the CIM

WE NEED YOUR HELP:

Look at the examples

and ... do it !

If you need help:      Kevin Kröninger (Particles)  
                                 Angus Wright (Astro)  
                                 Rainer Grauer (Simulations)

It is important in applying for the second phase.

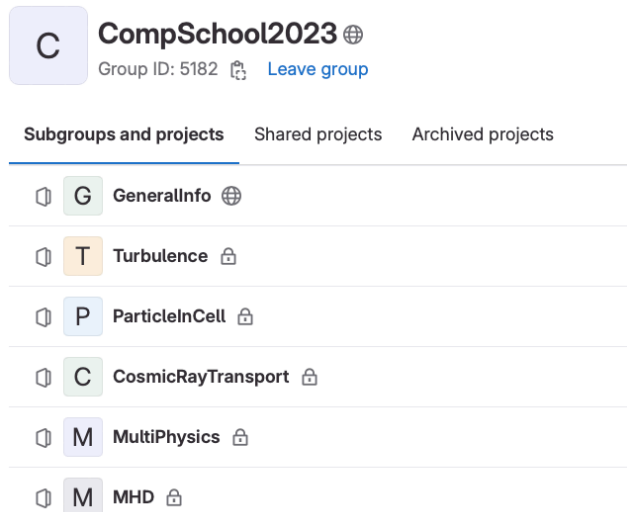
Responsibility: **PIs**

## ■ GitLab for the CIM

In addition:

Use the gitlab to store info and data for schools, workshops, seminar series ...

Let's have a look at an example ...



The screenshot shows the GitLab interface for a group named 'CompSchool2023'. The group ID is 5182, and there is a 'Leave group' link. Below the group name, there are three tabs: 'Subgroups and projects' (selected), 'Shared projects', and 'Archived projects'. Under the 'Subgroups and projects' tab, a list of subgroups is displayed, each with a folder icon, a letter in a colored box, the subgroup name, and a globe icon for public visibility:

- G** GeneralInfo (Public)
- T** Turbulence (Private)
- P** ParticleInCell (Private)
- C** CosmicRayTransport (Private)
- M** MultiPhysics (Private)
- M** MHD (Private)

# ■ Compute Cluster

galileo.cim.ruhr-uni-bochum.de

managed by Jürgen Möllenhoff@tp1

some technical data

- **1 x Head-Node** ([galileo.cim.rub.de](https://galileo.cim.rub.de), login and cluster management)
- ssh [galileo.cim.rub.de](https://galileo.cim.rub.de)
- cluster view via <https://galileo.cim.rub.de>
- **1 x AMD EPYC 7313P 16-Core Processor 3.0 GHz, 128MB L3 cache, (16 real kernels, 32 with HT)**
- **256 GB DDR4-3200MHz ECC-RAM**
- user directories located on 2 x 1 TB M.2 SSD as RAID1 → 1 TB can be used (/home)
- more information here: <https://www.gigabyte.com/Enterprise/Rack-Server/R272-Z30-rev-A00>
  
- **8 x Compute CPU-Nodes (c1-c8)**
- 4 nodes each share one cabinet
- access only via **SLURM** using the head node
- **2 x Intel(R) Xeon(R) Gold 5318Y CPU @ 2.10GHz, max. 3.4 GHz, 36 MB cache, 24 kernels (48 real kernels, 96 with HT)**
- **256 GB DDR4-3200MHz ECC-RAM**
- user directories and storage connected via NFSv4
- more information here: <https://www.gigabyte.com/Enterprise/High-Density-Server/H262-NO0-rev-100>

# ■ Compute Cluster

galileo.cim.ruhr-uni-bochum.de

## - **1 x Compute GPU-Node (c9)**

- access only via **SLURM** using the head-node
- **1 x AMD EPYC 7313P 16-Core Processor 3.0 GHz, 128MB L3 Cache, (16 real kernels, 32 with HT)**
- **128 GB DDR4-3200MHz ECC-RAM**
- **2 x NVIDIA® A100 - 80GB Tensor-Core-GPU** (<https://www.nvidia.com/de-de/data-center/a100/>)
- space for 8 GPU-cards, 6 cards could be added **needs to be improved**
- Benutzerverzeichnisse bzw. Speicherlaufwerk via NFS angebunden
- further information here: <https://www.gigabyte.com/Enterprise/GPU-Server/G292-Z20-rev-A00>

## - **1 x storage (fs)**

- access only via NFSv4 (/home/fs/users/)
- 1 x AMD EPYC 7313P 16-Core Processor 3.0 GHz, 128MB L3 cache, (16 real kernels, 32 with HT)
- 128 GB DDR4-3200MHz ECC-RAM
- **11 TB Speicherplatz via ZFS-RAID6, harddiscs RAID with 1 TB SSD-cache**
- We could add 7 more discs
- further information here: <https://www.gigabyte.com/Enterprise/Rack-Server/R272-Z30-rev-A00>

nodes connected **2 x 10 GB Ethernet**

**needs to be improved**

**Presently 22 users** (apply by writing an email to RG)

## ■ Data server

storing observational data (Dettmar)

similar to the John Hopkins turbulence database <https://turbulence.pha.jhu.edu/>

store data from large scale MHD simulations on JUWELS (Mike Wilbert)  
to use as turbulent background

HIT, various background fields, different resolutions, decaying and forced ...

# Thanks