HMC | HELMHOLTZ PROJECT HERMES

HELMHOLTZ RICH METADATA SOFTWARE PUBLICATIONS

HERMES Kickoff Workshop 2021-11-12

www.software-metadata.pub

Research Software Directory

past - present - future

Carlos Martinez-Ortiz

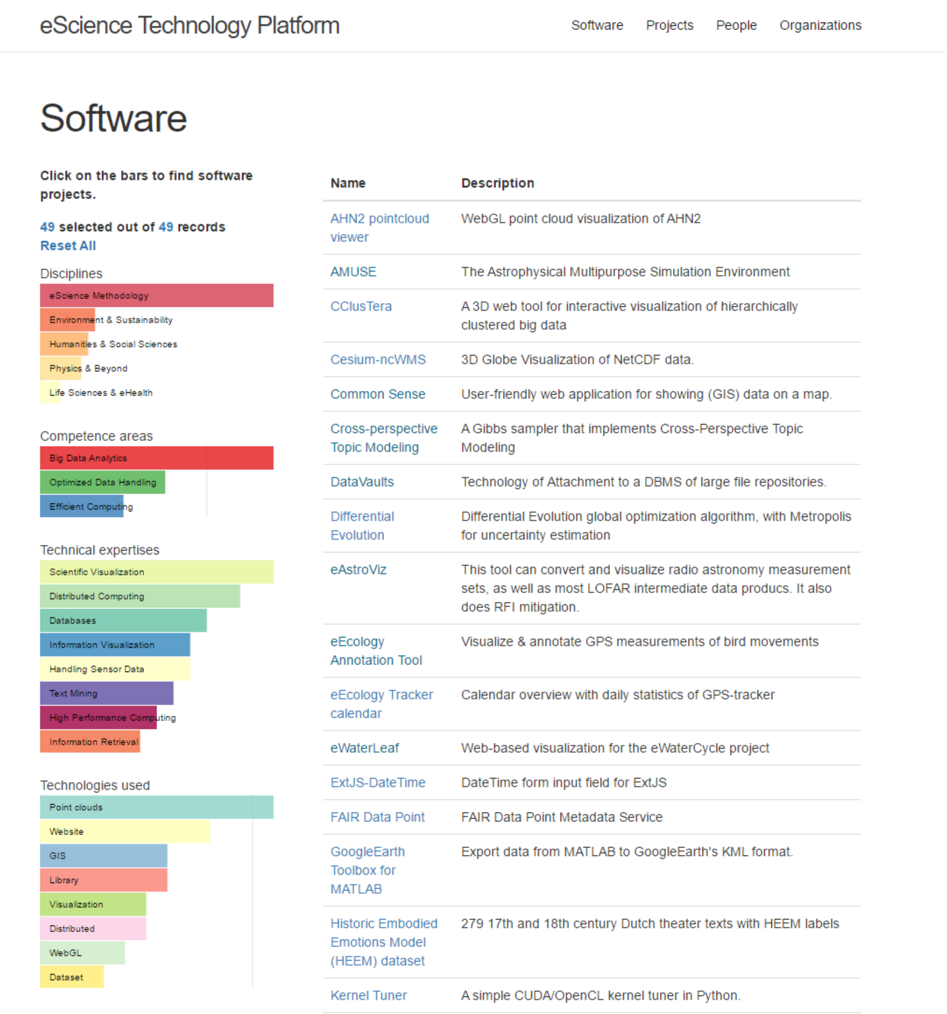
netherlands eScience center

The early days

2012-2017

netherlands eScience center

First baby (e-)steps: the eScience Technology Platform



eScience Technology Platform

Software

Click on the bars to find software projects.

49 selected out of 49 records

Reset All

Disciplines

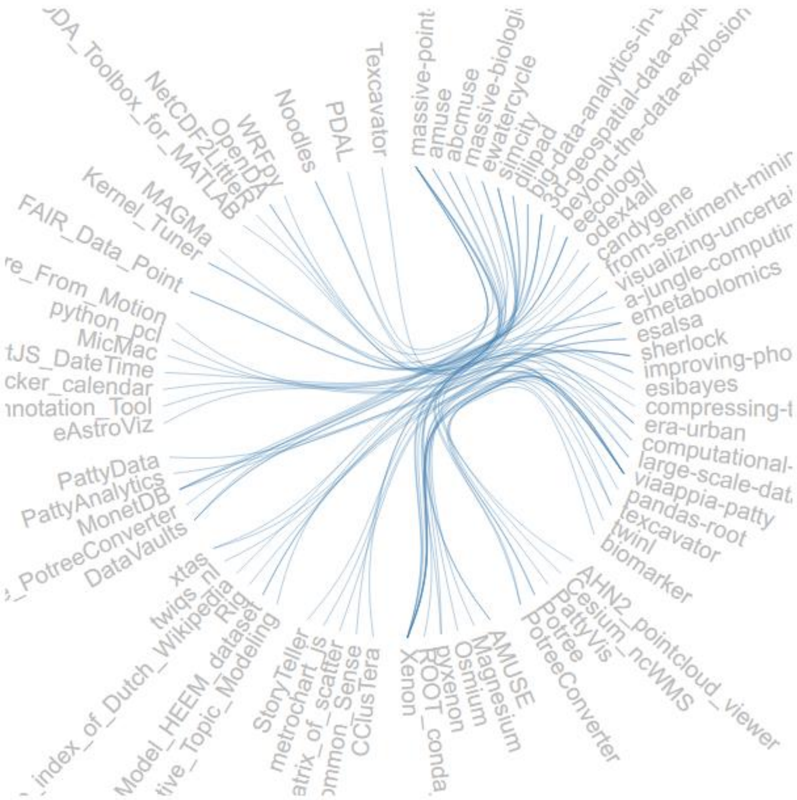
Competence areas

Technical expertises

Technologies used

Name	Description
AHN2 pointcloud viewer	WebGL point cloud visualization of AHN2
AMUSE	The Astrophysical Multipurpose Simulation Environment
CClusTera	A 3D web tool for interactive visualization of hierarchically clustered big data
Cesium-ncWMS	3D Globe Visualization of NetCDF data.
Common Sense	User-friendly web application for showing (GIS) data on a map.
Cross-perspective Topic Modeling	A Gibbs sampler that implements Cross-Perspective Topic Modeling
DataVaults	Technology of Attachment to a DBMS of large file repositories.
Differential Evolution	Differential Evolution global optimization algorithm, with Metropolis for uncertainty estimation
eAstroViz	This tool can convert and visualize radio astronomy measurement sets, as well as most LOFAR intermediate data producs. It also does RFI mitigation.
eEcology Annotation Tool	Visualize & annotate GPS measurements of bird movements
eEcology Tracker calendar	Calendar overview with daily statistics of GPS-tracker
eWaterLeaf	Web-based visualization for the eWaterCycle project
ExtJS-DateTime	DateTime form input field for ExtJS
FAIR Data Point	FAIR Data Point Metadata Service
GoogleEarth Toolbox for MATLAB	Export data from MATLAB to GoogleEarth's KML format.
Historic Embodied Emotions Model (HEEM) dataset	279 17th and 18th century Dutch theater texts with HEEM labels
Kernel Tuner	A simple CUDA/OpenCL kernel tuner in Python.

eStep Software used in Projects



Problem: how do know who we reach?

eScience self assessment 2017

“We have only just started to quantitively assess the impact of our software. However, it appears to be very challenging – if not impossible – to develop a standard set of reliable measures that provide a representative view of the popularity, usage or impact of our software.”

“To address some of the challenges in quantifying and monitoring the impact of our software activities, we are setting up an Alliance with CWI on the monitoring and assessment of the impact of software.”

netherlands eScience center

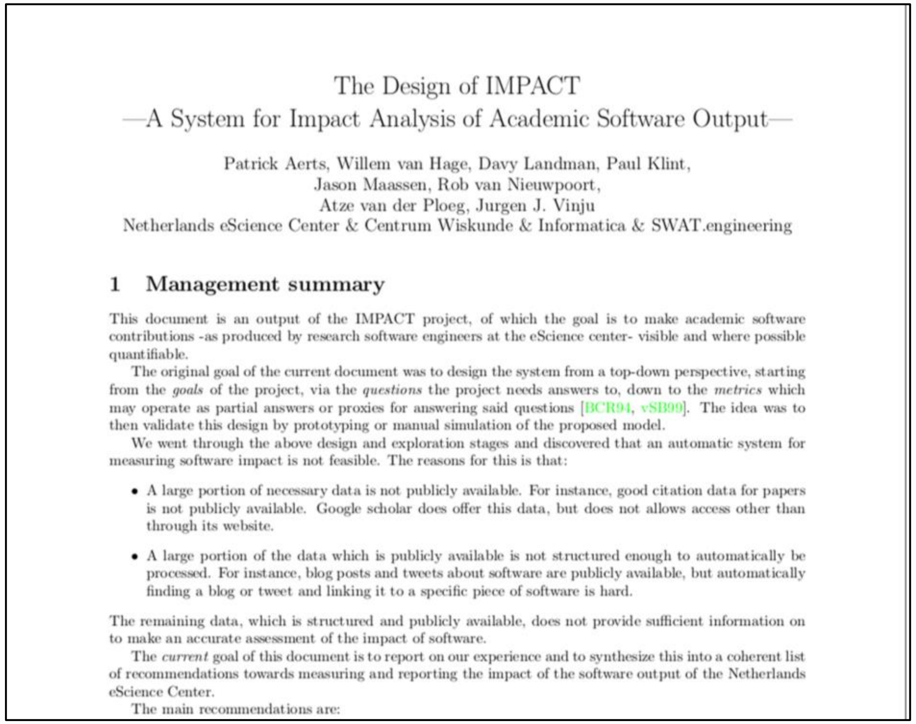
Self-assessment
November 2013 - October 2016

“To measure is to know” (Lord Kelvin)

The IMPACT alliance (2017-2018)

Alliance with the Software Analysis and Transformation (SWAT) group of CWI with the goal of making the impact of our software visible and quantifiable.

Conclusion:
Not enough (structured) data available to assess the impact



The Design of IMPACT
—A System for Impact Analysis of Academic Software Output—

Patrick Aerts, Willem van Hage, Davy Landman, Paul Klint,
Jason Maassen, Rob van Nieuwpoort,
Atze van der Ploeg, Jurgen J. Vinju
Netherlands eScience Center & Centrum Wiskunde & Informatica & SWAT.engineering

1 Management summary

This document is an output of the IMPACT project, of which the goal is to make academic software contributions -as produced by research software engineers at the eScience center- visible and where possible quantifiable.

The original goal of the current document was to design the system from a top-down perspective, starting from the *goals* of the project, via the *questions* the project needs answers to, down to the *metrics* which may operate as partial answers or proxies for answering said questions [BCR94, vSB99]. The idea was to then validate this design by prototyping or manual simulation of the proposed model.

We went through the above design and exploration stages and discovered that an automatic system for measuring software impact is not feasible. The reasons for this is that:

- A large portion of necessary data is not publicly available. For instance, good citation data for papers is not publicly available. Google scholar does offer this data, but does not allows access other than through its website.
- A large portion of the data which is publicly available is not structured enough to automatically be processed. For instance, blog posts and tweets about software are publicly available, but automatically finding a blog or tweet and linking it to a specific piece of software is hard.

The remaining data, which is structured and publicly available, does not provide sufficient information on to make an accurate assessment of the impact of software.

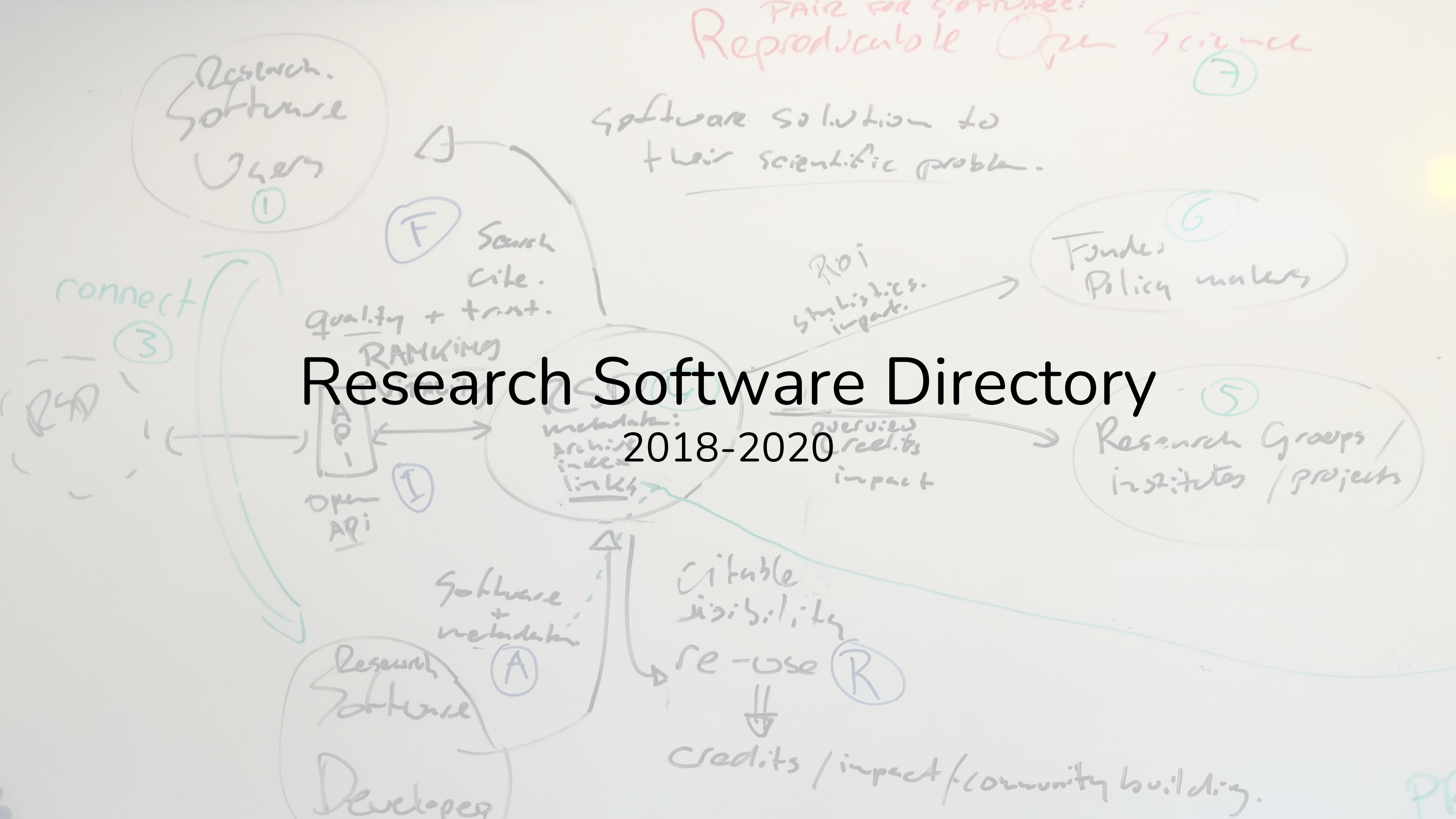
The *current* goal of this document is to report on our experience and to synthesize this into a coherent list of recommendations towards measuring and reporting the impact of the software output of the Netherlands eScience Center.

The main recommendations are:

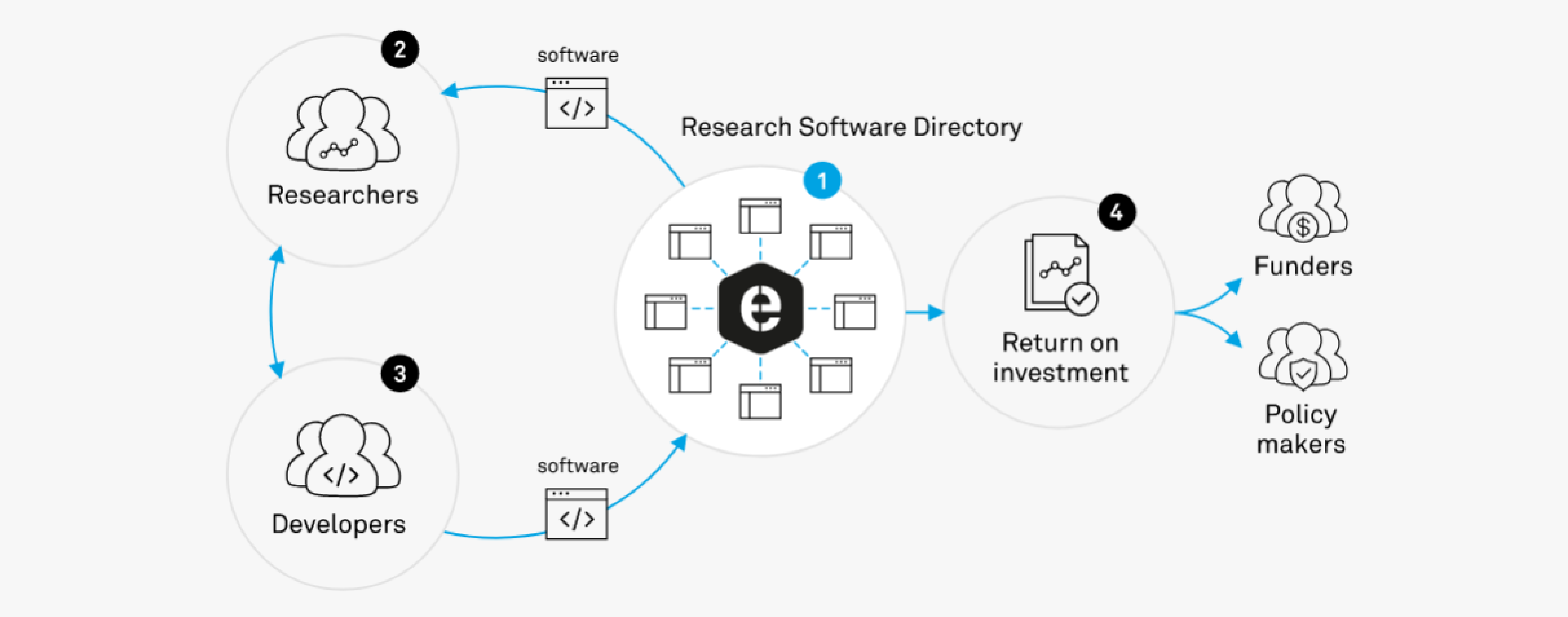
“To measure is to know” (Lord Kelvin)

Research Software Directory

2018-2020



Research Software Directory



Target Groups:

- Researchers, RSEs (2): find, judge relevance, citation info
- Developers, eRSEs (3): encouraging FAIR software development, show impact
- Institutes, funders, policy makers (4): insights into reuse and impact

Survey eScience Symposium 2017

Q: How do you “find” software

A: context is very important:

- what are my colleagues using?
- who else is using it?
- in which projects?
- what are the results?
- how active is it?
- who is contributing?
- who to contact?

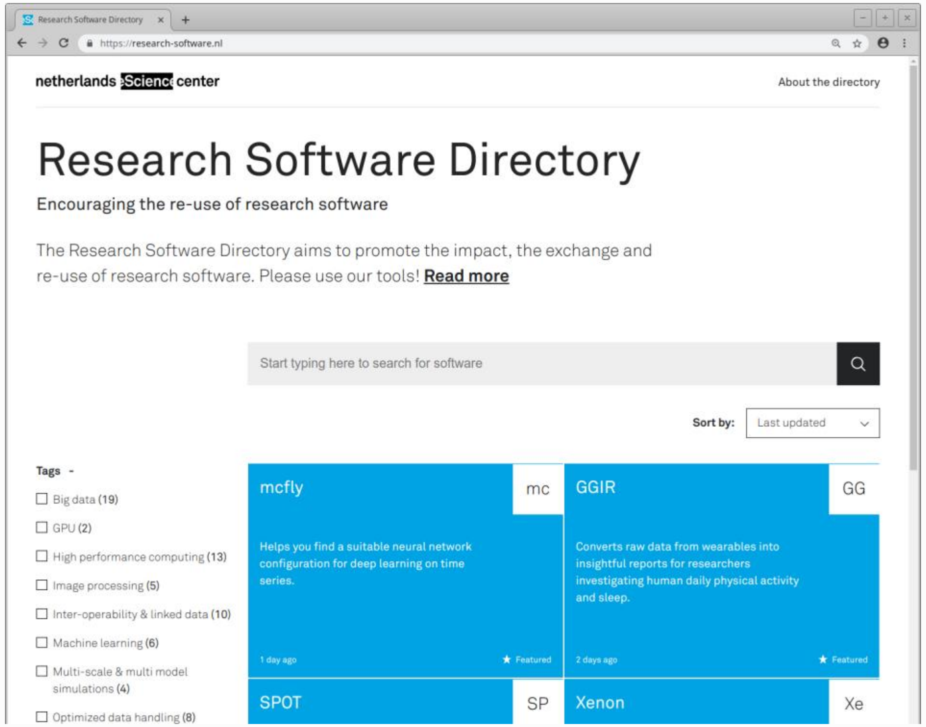
Sources of software:

- conferences
- papers
- google

People are unsure on how to cite software

Research Software Directory

- Aggregates data from many sources:
  - RSEs, GitHub, Zenodo, Zotero, CFF, Medium, YouTube, …
- Present this data to humans in a clear, concise, and structured way
- Expose this data using machine readable formats
  - schema.org, CodeMeta, OAI-PMH, …
- Provide citation information:
  - BibTex, EndNote, RIS, …



Research Software Directory
https://research-software.nl
netherlands eScience center
About the directory

Research Software Directory

Encouraging the re-use of research software

The Research Software Directory aims to promote the impact, the exchange and re-use of research software. Please use our tools! **Read more**

Start typing here to search for software
Sort by: Last updated
Tags -
□ Big data (19)
□ GPU (2)
□ High performance computing (13)
□ Image processing (5)
□ Inter-operability & linked data (10)
□ Machine learning (6)
□ Multi-scale & multi model simulations (4)
□ Optimized data handling (8)

mcfly — mc

Helps you find a suitable neural network configuration for deep learning on time series.

1 day ago ★ Featured

GGIR — GG

Converts raw data from wearables into insightful reports for researchers investigating human daily physical activity and sleep.

2 days ago ★ Featured

SPOT — SP

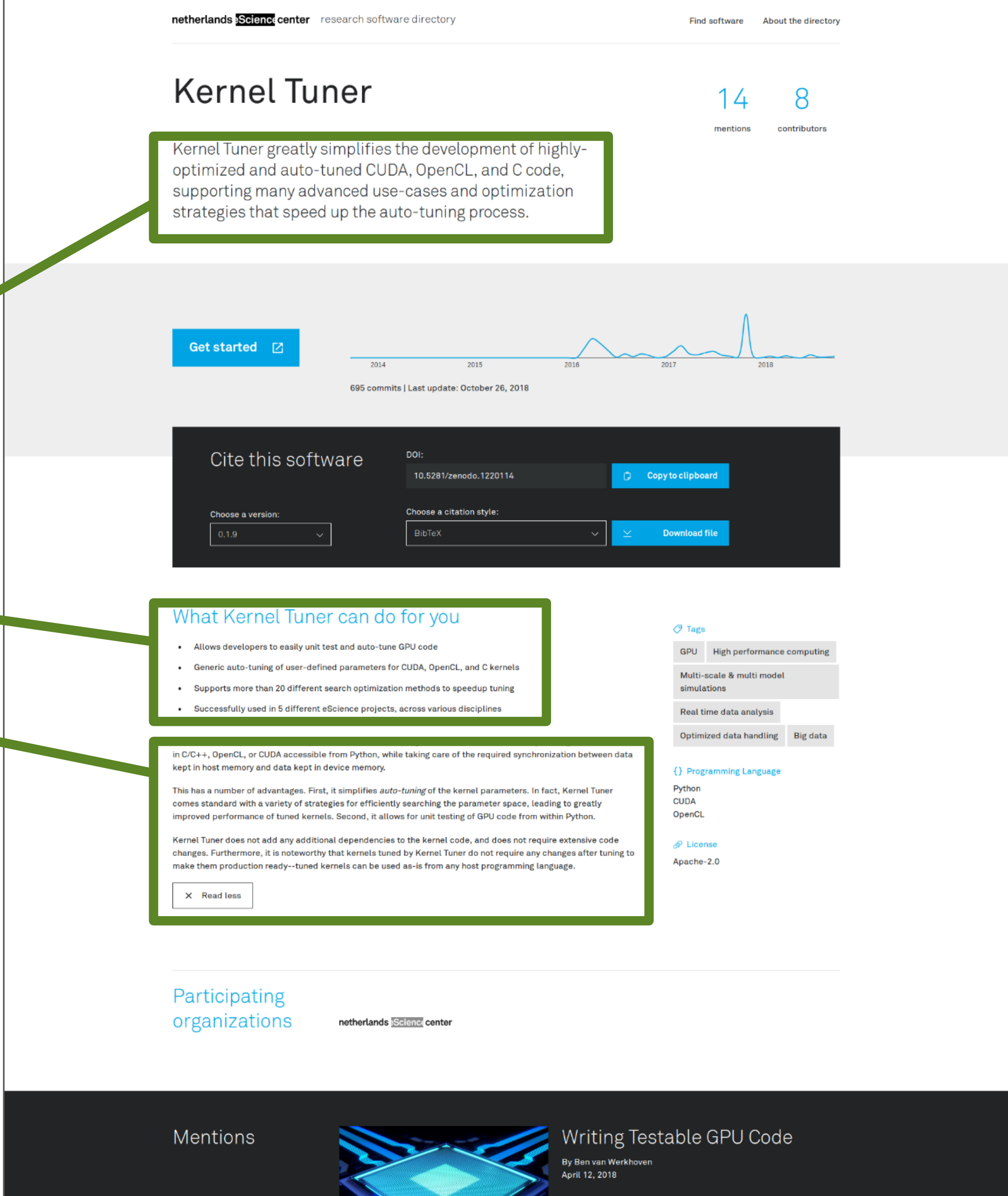
Xenon — Xe

Software findability (for humans)

What is the software for?

What problem does it solve?

For which research domain?



netherlands eScience center research software directory

Kernel Tuner

14 mentions

8 contributors

Kernel Tuner greatly simplifies the development of highly-optimized and auto-tuned CUDA, OpenCL, and C code, supporting many advanced use-cases and optimization strategies that speed up the auto-tuning process.

695 commits | Last update: October 26, 2018

Cite this software

DOI: 10.5281/zenodo.1220114

Choose a version: 0.1.9

Choose a citation style: BibTeX

What Kernel Tuner can do for you

- Allows developers to easily unit test and auto-tune GPU code
- Generic auto-tuning of user-defined parameters for CUDA, OpenCL, and C kernels
- Supports more than 20 different search optimization methods to speedup tuning
- Successfully used in 5 different eScience projects, across various disciplines

in C/C++, OpenCL, or CUDA accessible from Python, while taking care of the required synchronization between data kept in host memory and data kept in device memory.

This has a number of advantages. First, it simplifies *auto-tuning* of the kernel parameters. In fact, Kernel Tuner comes standard with a variety of strategies for efficiently searching the parameter space, leading to greatly improved performance of tuned kernels. Second, it allows for unit testing of GPU code from within Python.

Kernel Tuner does not add any additional dependencies to the kernel code, and does not require extensive code changes. Furthermore, it is noteworthy that kernels tuned by Kernel Tuner do not require any changes after tuning to make them production ready--tuned kernels can be used as-is from any host programming language.

Tags

GPU

High performance computing

Multi-scale & multi model simulations

Real time data analysis

Optimized data handling

Big data

Programming Language

Python
CUDA
OpenCL

License

Apache-2.0

Participating organizations

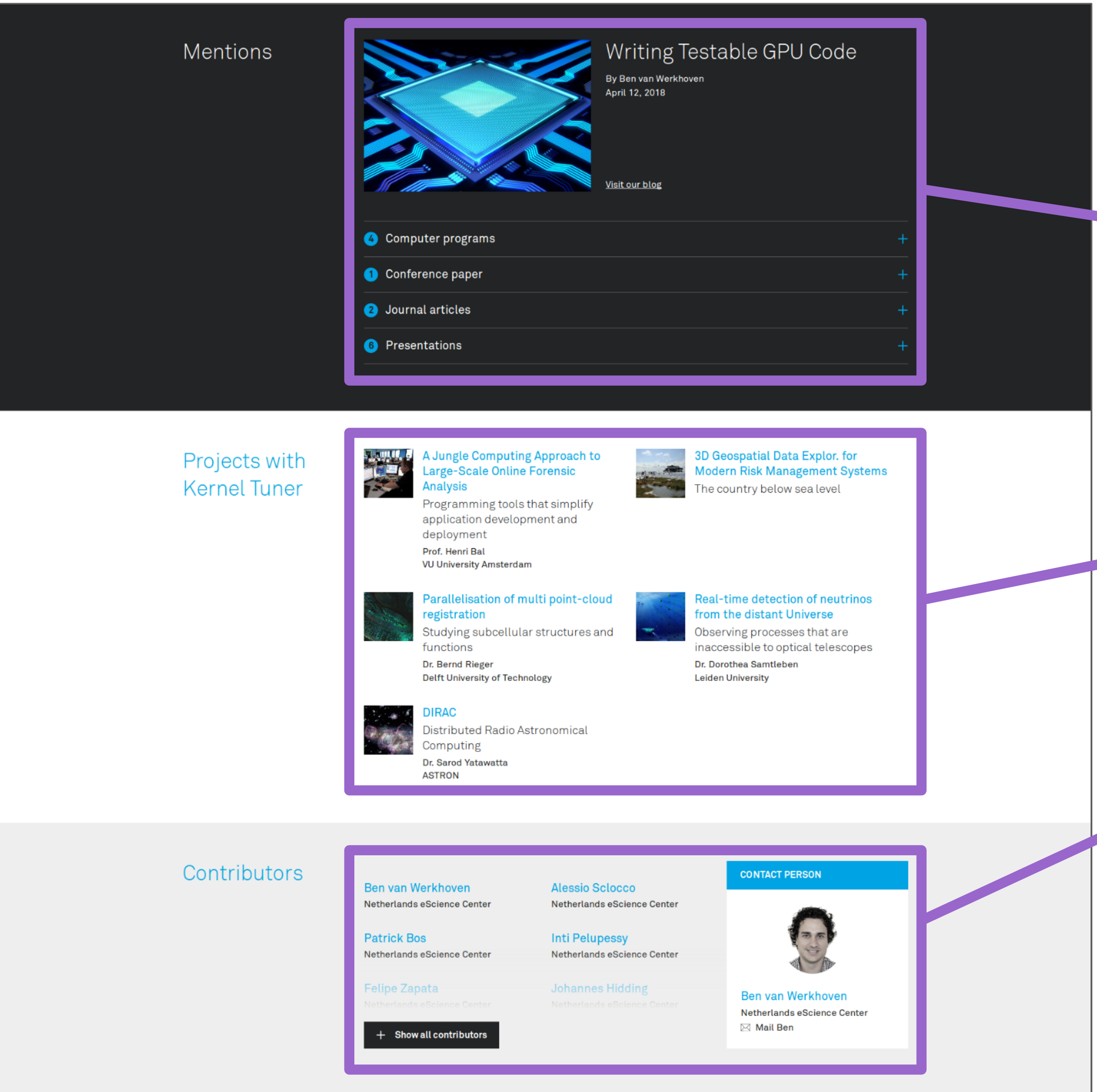
netherlands eScience center

Mentions

Writing Testable GPU Code

By Ben van Werkhoven
April 12, 2018

Academic and social context



Mentions

Writing Testable GPU Code

By Ben van Werkhoven
April 12, 2018

Visit our blog

- 4 Computer programs +
- 1 Conference paper +
- 2 Journal articles +
- 6 Presentations +

Projects with Kernel Tuner

A Jungle Computing Approach to Large-Scale Online Forensic Analysis
Programming tools that simplify application development and deployment
Prof. Henri Bal
VU University Amsterdam

3D Geospatial Data Explor. for Modern Risk Management Systems
The country below sea level

Parallelisation of multi point-cloud registration
Studying subcellular structures and functions
Dr. Bernd Rieger
Delft University of Technology

Real-time detection of neutrinos from the distant Universe
Observing processes that are inaccessible to optical telescopes
Dr. Dorothea Samtleben
Leiden University

DIRAC
Distributed Radio Astronomical Computing
Dr. Sarod Yatawatta
ASTRON

Contributors

Ben van Werkhoven
Netherlands eScience Center

Alessio Sclocco
Netherlands eScience Center

Patrick Bos
Netherlands eScience Center

Inti Pelupessy
Netherlands eScience Center

Felipe Zapata
Netherlands eScience Center

Johannes Hidding
Netherlands eScience Center

CONTACT PERSON

Ben van Werkhoven
Netherlands eScience Center
✉ Mail Ben

Papers, presentations, blogs, videos, …

Projects

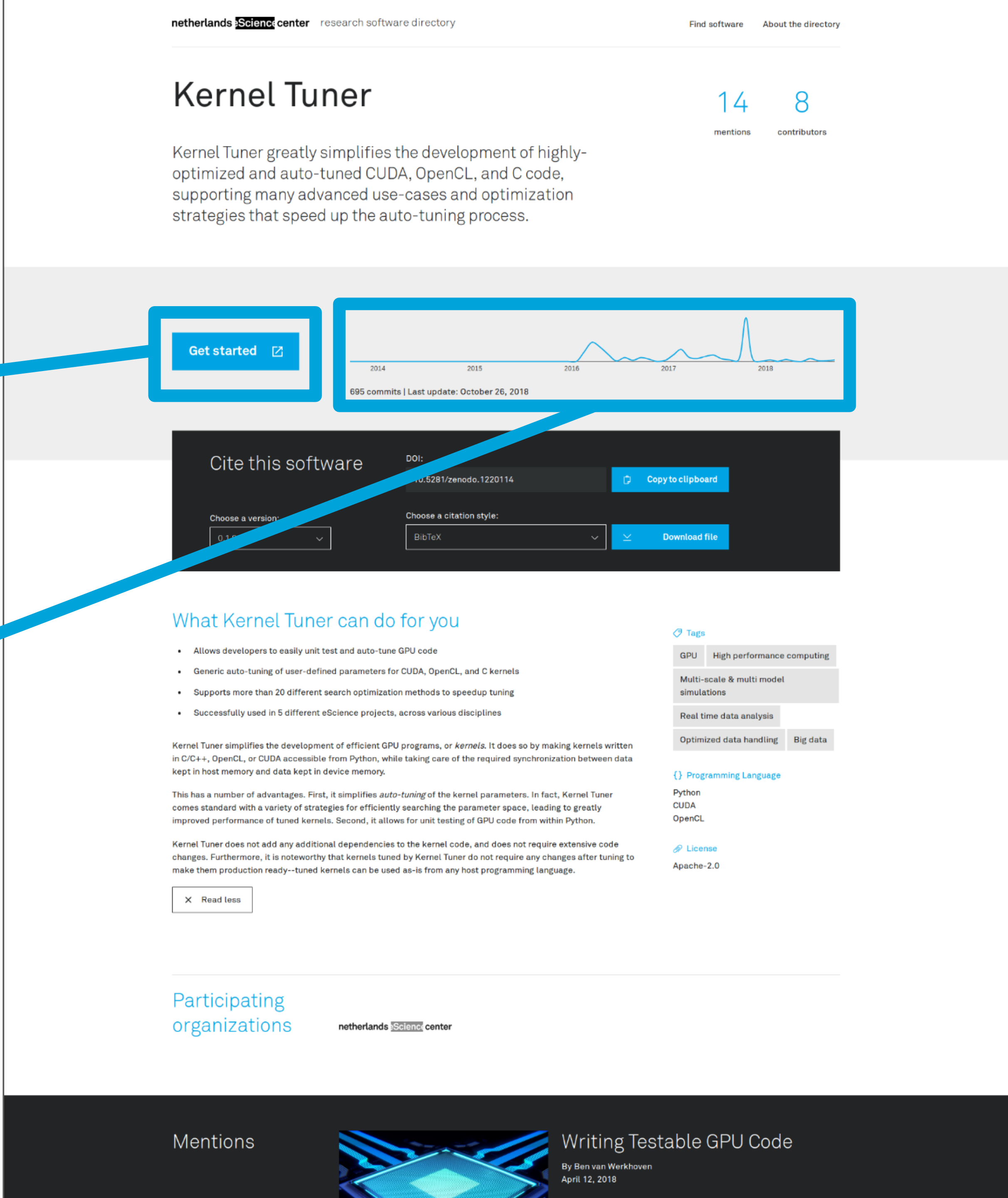
People

netherlands eScience center

Development activity

How to get started

Development activity



netherlands eScience center research software directory

Find software About the directory

Kernel Tuner

14 mentions

8 contributors

Kernel Tuner greatly simplifies the development of highly-optimized and auto-tuned CUDA, OpenCL, and C code, supporting many advanced use-cases and optimization strategies that speed up the auto-tuning process.

Get started

695 commits | Last update: October 26, 2018

Cite this software

DOI: 10.5281/zenodo.1220114

Copy to clipboard

Choose a version: 0.1

Choose a citation style: BibTeX

Download file

What Kernel Tuner can do for you

- Allows developers to easily unit test and auto-tune GPU code
- Generic auto-tuning of user-defined parameters for CUDA, OpenCL, and C kernels
- Supports more than 20 different search optimization methods to speedup tuning
- Successfully used in 5 different eScience projects, across various disciplines

Kernel Tuner simplifies the development of efficient GPU programs, or *kernels*. It does so by making kernels written in C/C++, OpenCL, or CUDA accessible from Python, while taking care of the required synchronization between data kept in host memory and data kept in device memory.

This has a number of advantages. First, it simplifies *auto-tuning* of the kernel parameters. In fact, Kernel Tuner comes standard with a variety of strategies for efficiently searching the parameter space, leading to greatly improved performance of tuned kernels. Second, it allows for unit testing of GPU code from within Python.

Kernel Tuner does not add any additional dependencies to the kernel code, and does not require extensive code changes. Furthermore, it is noteworthy that kernels tuned by Kernel Tuner do not require any changes after tuning to make them production ready--tuned kernels can be used as-is from any host programming language.

✕ Read less

Tags

GPU | High performance computing | Multi-scale & multi model simulations | Real time data analysis | Optimized data handling | Big data

{} Programming Language

Python
CUDA
OpenCL

License

Apache-2.0

Participating organizations

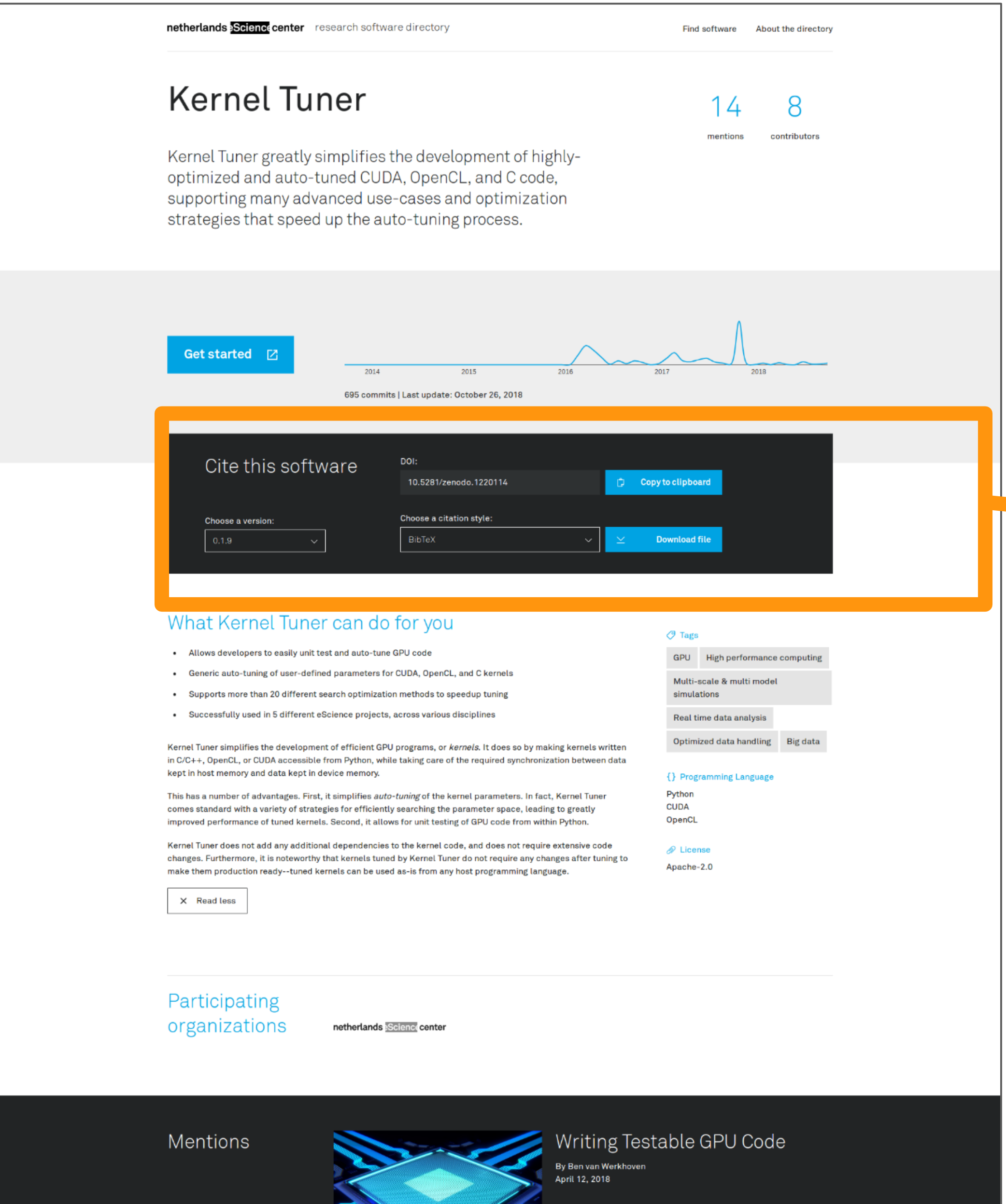
netherlands eScience center

Mentions

Writing Testable GPU Code

By Ben van Werkhoven
April 12, 2018

Encourage software citation and reproducibility



netherlands eScience center research software directory

Find software About the directory

Kernel Tuner

14 mentions 8 contributors

Kernel Tuner greatly simplifies the development of highly-optimized and auto-tuned CUDA, OpenCL, and C code, supporting many advanced use-cases and optimization strategies that speed up the auto-tuning process.

Get started

695 commits | Last update: October 26, 2018

Cite this software

DOI: 10.5281/zenodo.1220114

Copy to clipboard

Choose a version: 0.1.9

Choose a citation style: BibTeX

Download file

What Kernel Tuner can do for you

- Allows developers to easily unit test and auto-tune GPU code
- Generic auto-tuning of user-defined parameters for CUDA, OpenCL, and C kernels
- Supports more than 20 different search optimization methods to speedup tuning
- Successfully used in 5 different eScience projects, across various disciplines

Kernel Tuner simplifies the development of efficient GPU programs, or *kernels*. It does so by making kernels written in C/C++, OpenCL, or CUDA accessible from Python, while taking care of the required synchronization between data kept in host memory and data kept in device memory.

This has a number of advantages. First, it simplifies *auto-tuning* of the kernel parameters. In fact, Kernel Tuner comes standard with a variety of strategies for efficiently searching the parameter space, leading to greatly improved performance of tuned kernels. Second, it allows for unit testing of GPU code from within Python.

Kernel Tuner does not add any additional dependencies to the kernel code, and does not require extensive code changes. Furthermore, it is noteworthy that kernels tuned by Kernel Tuner do not require any changes after tuning to make them production ready--tuned kernels can be used as-is from any host programming language.

✕ Read less

Tags

GPU | High performance computing | Multi-scale & multi model simulations | Real time data analysis | Optimized data handling | Big data

{} Programming Language

Python
CUDA
OpenCL

License

Apache-2.0

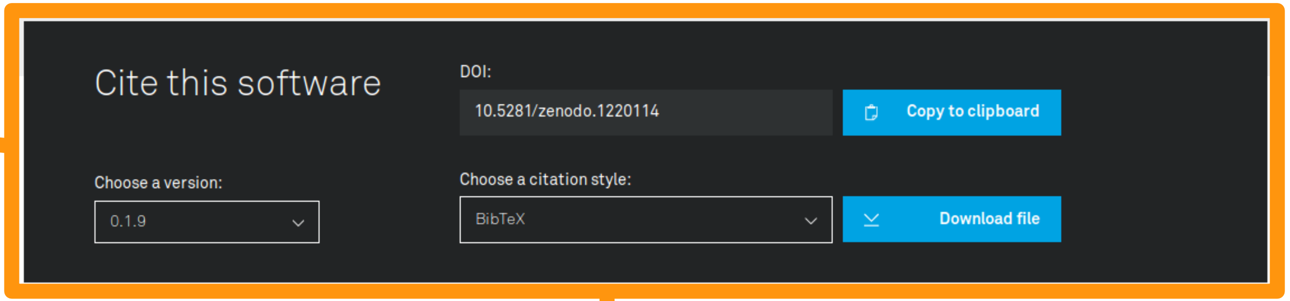
Participating organizations

netherlands eScience center

Mentions

Writing Testable GPU Code

By Ben van Werkhoven
April 12, 2018



Cite this software

DOI: 10.5281/zenodo.1220114

Copy to clipboard

Choose a version: 0.1.9

Choose a citation style: BibTeX

Download file

Software citation

The reaction

A lot of interest

HUMBOLDT-UNIVERSITÄT ZU BERLIN

EUROPEAN OPEN
SCIENCE CLOUD

SURF

the hyve

CWI Centrum Wiskunde & Informatica

PROCESS

ePLAN

DTL
DUTCH TECHCENTRE FOR LIFE SCIENCES

VU VRIJE UNIVERSITEIT AMSTERDAM

KB LAB

OpenAIRE

university of groningen

Amsterdam UMC

Universiteit Leiden

Data Archiving and Networked Services
DANS

SURF SARA

HSF
HEP Software Foundation

PLOS

GitHub

CLARIAH
Common Lab Research Infrastructure
for the Arts and Humanities

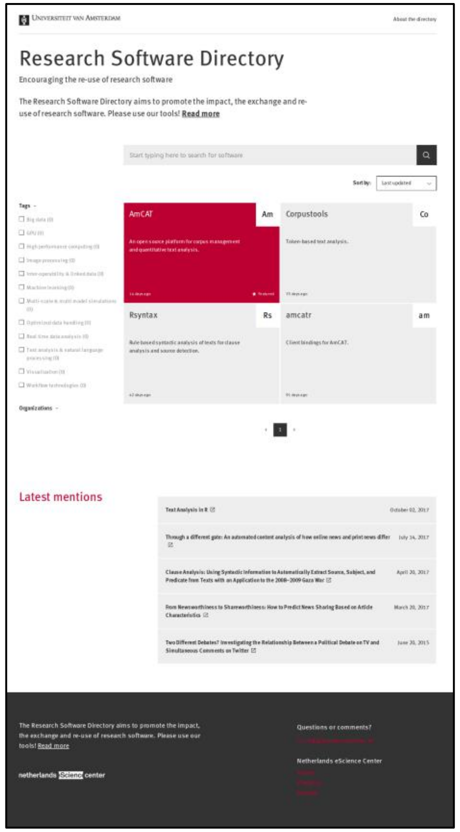
DIGITAL HUMANITIES LAB

Other RSD-pilots

- We have been asked to setup independent RSDs for PROCESS (H2020), the UU Digital Humanities Lab, and the Communication Science group of the UvA

- These are "proofs of concept" using project or group specific branding

- DLR and Leipzig University Library have set up their own pilot



Universiteit van Amsterdam | About the directory

Research Software Directory

Encouraging the re-use of research software

The Research Software Directory aims to promote the impact, the exchange and re-use of research software. Please use our tools! **Read more**

Start typing here to search for software

Sort by: Last updated

Tags

- □ Big data (0)
- □ GPU (0)
- □ High performance computing (0)
- □ Image processing (0)
- □ Inter-operability & linked data (0)
- □ Machine learning (0)
- □ Multi-scale & multi model simulations (0)
- □ Optimized data handling (0)
- □ Real time data analysis (0)
- □ Text analysis & natural language processing (0)
- □ Visualization (0)
- □ Workflow technologies (0)

Organizations

AmCAT (Am): An open source platform for corpus management and quantitative text analysis.

Corpustools (Co): Token-based text analysis.

Rsyntax (Rs): Rule based syntactic analysis of texts for clause analysis and source detection.

amcatr (am): Client bindings for AmCAT.

‹ 1 ›

Latest mentions

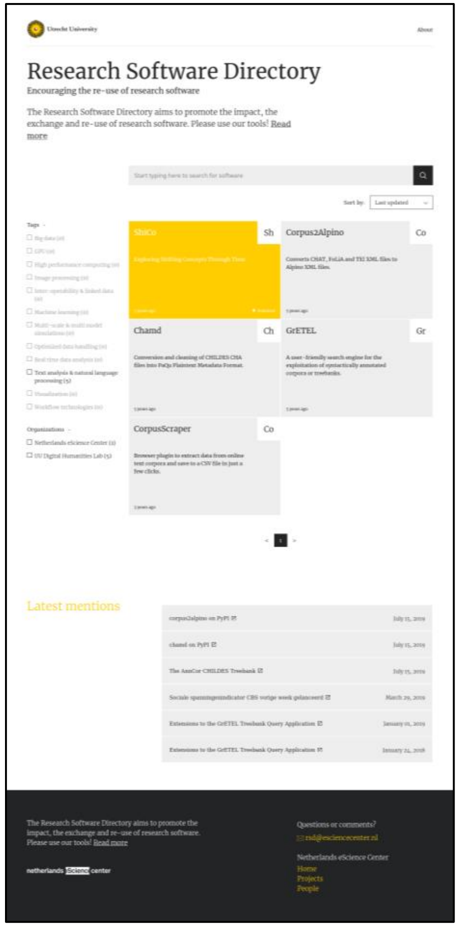
Text Analysis in R	October 02, 2017
Through a different gate: An automated content analysis of how online news and print news differ	July 14, 2017
Clause Analysis: Using Syntactic Information to Automatically Extract Source, Subject, and Predicate from Texts with an Application to the 2008–2009 Gaza War	April 20, 2017
From Newsworthiness to Shareworthiness: How to Predict News Sharing Based on Article Characteristics	March 20, 2017
Two Different Debates? Investigating the Relationship Between a Political Debate on TV and Simultaneous Comments on Twitter	June 20, 2015

The Research Software Directory aims to promote the impact, the exchange and re-use of research software. Please use our tools! Read more

netherlands eScience center

Questions or comments?

Netherlands eScience Center



Utrecht University | About

Research Software Directory

Encouraging the re-use of research software

The Research Software Directory aims to promote the impact, the exchange and re-use of research software. Please use our tools! Read more

Start typing here to search for software

Sort by: Last updated

Tags

- □ Text analysis & natural language processing (5)

Organizations

- □ Netherlands eScience Center (1)
- □ UU Digital Humanities Lab (5)

ShiCo (Sh): Exploring Shifting Concepts Through Time

Corpus2Alpino (Co): Converts CHAT, FoLiA and TEI XML files to Alpino XML files.

Chamd (Ch): Conversion and cleaning of CHILDES CHA files into PaQu Plaintext Metadata Format.

GrETEL (Gr): A user-friendly search engine for the exploitation of syntactically annotated corpora or treebanks.

CorpusScraper (Co): Browser plugin to extract data from online text corpora and save to a CSV file in just a few clicks.

< 1 >

Latest mentions

corpus2alpino on PyPI	July 15, 2019
chamd on PyPI	July 15, 2019
The AnnCor CHILDES Treebank	July 15, 2019
Sociale spanningsindicator CBS vorige week gelanceerd	March 29, 2019
Extensions to the GrETEL Treebank Query Application	January 01, 2019
Extensions to the GrETEL Treebank Query Application	January 24, 2018

The Research Software Directory aims to promote the impact, the exchange and re-use of research software. Please use our tools! Read more

netherlands eScience center

Questions or comments?

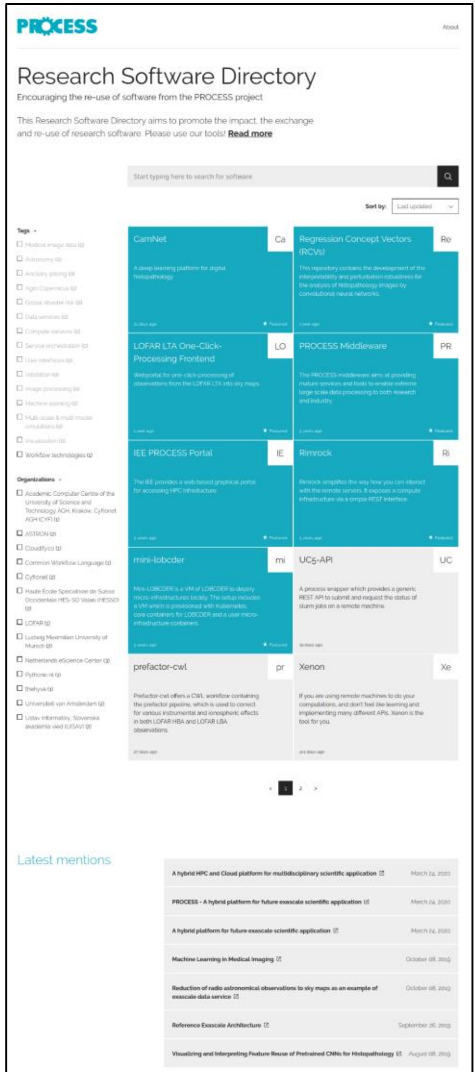
rsd@esciencecenter.nl

Netherlands eScience Center

Home

Projects

People



PROCESS | About

Research Software Directory

Encouraging the re-use of software from the PROCESS project

This Research Software Directory aims to promote the impact, the exchange and re-use of research software. Please use our tools! **Read more**

Start typing here to search for software

Sort by: Last updated

Tags

- □ Medical image data
- □ Astronomy
- □ Ancillary pricing
- □ Agro Copernicus
- □ Global disaster risk
- □ Data services
- □ Compute services
- □ Service orchestration
- □ User interfaces
- □ Validation
- □ Image processing
- □ Machine learning
- □ Multi-scale & multi model simulations
- □ Visualization
- □ Workflow technologies (1)

Organizations

- □ Academic Computer Centre of the University of Science and Technology AGH, Krakow, Cyfronet AGH (CYF) (1)
- □ ASTRON (2)
- □ Cloudifyco (1)
- □ Common Workflow Language (1)
- □ Cyfronet (2)
- □ Haute Ecole Specialisée de Suisse Occidentale HES-SO Valais (HESSO) (2)
- □ LOFAR (1)
- □ Ludwig Maximilian University of Munich (2)
- □ Netherlands eScience Center (3)
- □ Pythonic.nl (1)
- □ thehyve (1)
- □ Universiteit van Amsterdam (2)
- □ Ustav informatiky, Slovenska akademia vied (UISAV) (2)

CamNet (Ca): A deep learning platform for digital histopathology.

Regression Concept Vectors (RCVs) (Re): This repository contains the development of the interpretability and perturbation robustness for the analysis of histopathology images by convolutional neural networks.

LOFAR LTA One-Click-Processing Frontend (LO): Webportal for one-click-processing of observations from the LOFAR LTA into sky maps.

PROCESS Middleware (PR): The PROCESS middleware aims at providing mature services and tools to enable extreme large scale data processing to both research and industry.

IEE PROCESS Portal (IE): The IEE provides a web based graphical portal for accessing HPC infrastructure.

Rimrock (Ri): Rimrock simplifies the way how you can interact with the remote servers. It exposes a compute infrastructure via a simple REST interface.

mini-lobcder (mi): Mini-LOBCDER is a VM of LOBCDER to deploy micro-infrastructures locally. The setup includes a VM which is provisioned with Kubernetes, core containers for LOBCDER and a user micro-infrastructure containers.

UC5-API (UC): A process wrapper which provides a generic REST API to submit and request the status of slurm jobs on a remote machine.

prefactor-cwl (pr): Prefactor-cwl offers a CWL workflow containing the prefactor pipeline, which is used to correct for various instrumental and ionospheric effects in both LOFAR HBA and LOFAR LBA observations.

Xenon (Xe): If you are using remote machines to do your computations, and don't feel like learning and implementing many different APIs, Xenon is the tool for you.

< 1 2 >

Latest mentions

A hybrid HPC and Cloud platform for multidisciplinary scientific application	March 24, 2020
PROCESS - A hybrid platform for future exascale scientific application	March 24, 2020
A hybrid platform for future exascale scientific application	March 24, 2020
Machine Learning in Medical Imaging	October 08, 2019
Reduction of radio astronomical observations to sky maps as an example of exascale data service	October 08, 2019
Reference Exascale Architecture	September 26, 2019
Visualizing and Interpreting Feature Reuse of Pretrained CNNs for Histopathology	August 08, 2019

Open issues

What is still missing

- Aggregated information on impact and re-use
- The numbers we need annual reports and reviews
- The insight we need for our day-to-day work

More issues

- Part of the content has to be maintained manually
- Data entry UI is sub-optimal
- Technical stack is too complicated
- Despite the external interest, uptake is not great

- Current “it’s open-source so host your own” approach does not work!

The future (?)

Lot’s of plans

- Improving the user friendliness for data entry
- Improving the internal data curation procedures
- Post mortem of failed adoptions
- Generated project pages
- Right side – dashboard
- RSD-as-a-service
- Flexible branding
- Improve community engagement
- Support for non-technical users
- Improved authentication
- Technical clean-up
- Support for more data sources
- ….

netherlands eScience center

Let's stay in touch

🌐 www.eScienceCenter.nl

✉ info@esciencecenter.com

☎ +31 (0)20 460 4770

Thank you!

Where to learn more about project HERMES?

Stephan Druskat, DLR, PI, *@stdruskat*

Oliver Bertuch, FZJ, PI, *@poi_ki_lo_therm*

Guido Juckeland, HZDR, PI, *@GuidoJuckeland*

Oliver Knodel, HZDR, *@olikno1*

Tobias Schlauch, DLR, *@TobiasSchlauch*

- Find us on Twitter
- Write an email to team@software-metadata.pub
- Go to software-metadata.pub