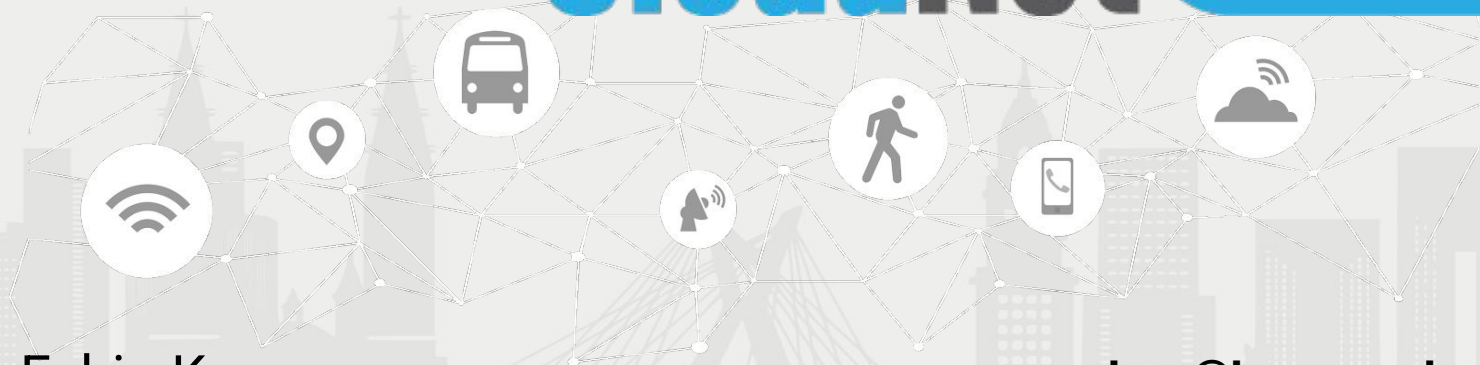


Research Software and the role of Research Software Engineers in Science



IEEE CloudNet



Prof. Fabio Kon
IME
University of São Paulo, Brazil

kon@ime.usp.br



Software is everywhere in Science



- Software is the most ubiquitous tool in contemporary science
- Software in Research (e.g., Linux, spreadsheets)

vs.

- Research Software
 - Established in Biology, Medicine, Physics, Chemistry, Engineering, Economics, Mathematics, Environment, ...
 - Growing in Social Sciences, Humanities, Arts, ...

- **Research Software** includes source code files, algorithms, scripts, computational workflows and executables that were **created during the research process or for a research purpose**
- Additional software components (e.g., operating systems, libraries, dependencies, packages, scripts, etc.) that are used for research but were **not created during or with a clear research intent** should be considered **software in research** and not Research Software
- This differentiation may vary between disciplines

- Research software is a component of our instruments
- Research software is the instrument
- Research software analyses research data
- Research software presents research results
- Research software assembles or integrates existing components into a working whole
- Research software is infrastructure or an underlying tool
- Research software facilitates distinctively research-oriented collaboration

How do we know research software is important?



- Funding
 - ~20% of NSF projects over 11 years topically discuss software in their abstracts (\$10b)
 - 2 of 3 main DOE ECP areas are research software (~\$4b)
 - \$300m of FY2021 NIH projects include “software development”
- Publications
 - Software intensive projects are a majority of current publications
 - Most-cited papers are methods and software
- Researchers
 - >90% of US/UK researchers use research software
 - ~65% would not be able to do their research without it
 - ~50% develop software as part of their research

Collected from <http://www.dia2.org> in 2017

Collected from <https://reporter.nih.gov> in 2022

Nangia and Katz; [10.1109/eScience.2017.78](https://doi.org/10.1109/eScience.2017.78)
“Top 100-cited papers of all time,” Nature, 2014
[10.1038/514550a](https://doi.org/10.1038/514550a)

S. Hettrick;
<https://www.software.ac.uk/blog/2016-09-12-its-impossible-conduct-research-without-software-say-7-out-10-uk-researchers>
S.J. Hettrick, et al.; [10.5281/zenodo.14809](https://doi.org/10.5281/zenodo.14809)
U. Nangia and D. S. Katz;
[10.6084/m9.figshare.5328442.v1](https://doi.org/10.6084/m9.figshare.5328442.v1)

<ReSA>

Research Software Alliance

Our vision: Research software and those who develop and maintain it
are recognised and valued as fundamental and vital to research
worldwide

Learn more

Where does research software come from?



- Significant fraction developed in research
- From the start of computing
 - Software appears around 1948
 - Research software (weather) in early 1950s
 - Software engineering starting in late 1960s, mostly initially applied to operational software (operating system, NASA flights, etc.)
- However:
 - Researchers (faculty) generally don't know good software practices
 - Software engineers generally don't understand research context
 - Students & postdocs generally don't know good software practices and don't stick around
 - Some postdocs do stay, join staff (perhaps unofficially)
 - Staff with research understanding and software engineering skills develop

- Most software produced by scientists is very bad!
- A lot of Research Software is written by physicists, biologists, mathematicians, economists, etc.
 - With no or very little training in Software Engineering and Computer Science
- A lot of research software is written by graduate students.
 - Whose goal is to get their degree, not to produce robust software to be used by other scientists
- Thus, most Research Software nowadays is not well architected, not well documented, hard to use => not sustainable, not reused
- Waste of resources, waste of public money

What about Computer Science?



- Are we producing good Research Software?
- What about the Cloud Networking Research community?
- First, let's talk a bit about Open Science...

International movement advocating that high quality research funded with public money must be available to all and, therefore, it must:

1. Publish openly the data it uses and produces
2. Publish openly the tools (e.g., source code) and methodology it uses
3. Publish the papers openly

There's no excuse for not opening your science



Unless you intend to use your results in a commercial project (e.g., by creating a startup company), there's no excuse to hide the means you used to achieve your results.

- Not publishing your source code is a bad practice
- Not publishing your data is a bad practice
- Not making your manuscripts available for free is a bad practice
 - Current APCs are outrageous, mainly for developing countries
 - But normally you're allowed to archive for free the accepted manuscript, so do that!

Now, let's answer these questions...



- Are we, CS researchers, producing good Research Software?
 - Yes
 - No
- What about the Networking Research community?
 - yes: ns-2, ns-3, WireShark, OMNET++, CloudSim, CloudStack, etc.
 - YES: TCP/IP stack
 - NO: historically most networking conferences don't give much importance to availability of artifacts
 - Question: how many papers contain fake results?

Papers will be evaluated based on the following criteria:

[...] iv) **Verifiability and Transparency:** [...]

*The **guiding principle** is that all research results should be accessible to the public and, if possible, **empirical studies should be reproducible**. In particular, we actively support the adoption of open artifacts and open source principles. We encourage all contributing authors to disclose (anonymized and curated) data/artifacts to increase reproducibility and replicability. **Note that sharing research artifacts is not mandatory for submission or acceptance. However, sharing is expected to be the default, and non-sharing needs to be justified.***

Upon submission to the research track, authors are asked

- to make their artifact available to the program committee or
- to indicate in the submission why they do not intend to make their data or study materials publicly available

And the networking community? - an anecdote



- Survey of Smart Grid Communications and Networking
- 54 Simulators/simulation works
- Only 15 of them have available software
- Only 12 have source code available
- Only 11 are open source software
- Only 3 were active
- But the protocols they use are not the industry standard
- Solution: libiec61850 on ns-3

TABLE II
CONSIDERED SIMULATORS

Name / Author	Year	Onjezky et al.	Mets et al.	Vogt et al.	Mihal et al.	Li et al.	Docquier et al.
MOSAik [27]	2012	✓	✓	✓	✓		
GECCO [28]	2012		✓			✓	
GridLAB-D [29]	2009	✓	✓			✓	
Godfrey et al. [30]	2010		✓	✓		✓	
EPOCHS [31]	2006		✓		✓	✓	
HELICS [32]	2017	✓			✓		
GridSpice [33]	2014		✓	✓			
GridSim [34]	2011		✓			✓	
IBCN Smart Grid Simulator [35]	2011		✓			✓	
DACCOSIM [36]	2015			✓	✓		
MECSYCO [37]	2015			✓	✓		
FNCS [38]	2014			✓	✓		
INSPIRE [39]	2013			✓	✓		
PowerNet [40]	2011			✓			
VPNET [41]	2011				✓	✓	
ORNL power system simulator [42]	2011		✓				
SGiC [43]	2011		✓				
Bergmann et al. [44]	2011		✓				
Kounev et al. [45]	2015			✓			
BoFit	2010			✓			
Ahmad et al. [46]	2015			✓			
GridQ [47]	2010			✓			
Wang et al. [48]	2012			✓			
Rajendram et al. [49]	2012			✓			
OpSIM [50]	2015			✓			
Ptolemy [51]	2003			✓			
Virgil [52]	2015			✓			
SMB [53]	2013			✓			
Andrén et al. [54]	2011			✓			
Manbachi et al. [55]	2015			✓			
Ding et al. [56]	2016			✓			
Bhor et al. [57]	2014			✓			
OrPHEUS [58]	2016			✓			
DACCOSIM-NG [59]	2019				✓		
CyDER [60]	2018				✓		
Nutaro et al. [61]	2007					✓	
Levesque et al. [62]	2012					✓	
Zhu et al. [63]	2011					✓	
MAPNET [64]	2011					✓	
SmartGridLab [65]	2012					✓	
SCORE [66]	2012					✓	
Branicky et al. [67]	2003					✓	
Tong et al. [68]	2010					✓	
SensorSim [69]	2000					✓	
Truetime [70]	2002					✓	
Prowler [71]	2003					✓	
VisualSense [72]	2004					✓	
Sidhu et al. [73]	2006						✓
Tarlochan et al. [74]	2007						✓
Kanabar et al. [75]	2010						✓
Juárez et al. [76]	2012						✓
Leon et al. [77]	2016						✓
STARS [78]	2019						✓
Ngo et al. [79]	2014						✓

Starting to change? Brazilian Symposium on Computer Networks and Distributed Systems - SBRC'2025



For the first time, we have: OPEN SCIENCE PRINCIPLES

- SBRC stimulates authors to adopt Open Science principles and practices. Therefore, authors are encouraged to disclose data sets, source code, tools, and other artifacts used in their research to promote transparency, reproducibility, and replicability of their work, for example, by including links to repositories or replication packages. Authors are also suggested to include an unnumbered section entitled "Availability of Artifacts" after the conclusion section, in which they can inform where research artifacts are available and how to access them. If it is not possible to make such artifacts available due to, for example, confidentiality or privacy issues, authors are suggested to include a statement about this impossibility. It is essential to highlight that sharing research artifacts is desirable, but it is not mandatory to submit papers or a criterion for acceptance of submitted papers.

What's missing for better Research Software



1. Open Science must be strongly encouraged
2. Researchers who produce it must be valued
 - a. What's best? 10 top papers or a software used by 10 research groups
3. Research agencies must provide funds to support its creation/maintenance
4. People who code it must be valued: **Research Software Engineers**

Latest CGI-FAPESP call (November 2022)



<https://fapesp.br/15733> (item 6.6)

Projects can request:

1. Technical training fellows
2. Up to 10% of the budget for making the software robust, reusable, well documented, thus sustainable.
 - a. e.g., a 4M reais project can use 400K just for software sustainability

Software Management Plan

Document describing how a project will manage the software it'll create.



SOFTWARE MANAGEMENT PLAN

1) Project information

Title:

Grant #:

Principal Investigator:

Researcher responsible for software management:

2) Software development team and its organization.

3) Main software features to be developed.

Practical guide to
Software
Management
Plans



Research Software Engineers - the RSE movement



Breakout group at Software Sustainability Institute's 2012 Collaborations Workshop, Cambridge, UK found:

- Lots of people already doing this work, but
- No common title
 - Chose Research Software Engineer (RSE)
- No community
 - Started associations/societies
- Not a profession
 - Defined career paths, structure

194 different job titles

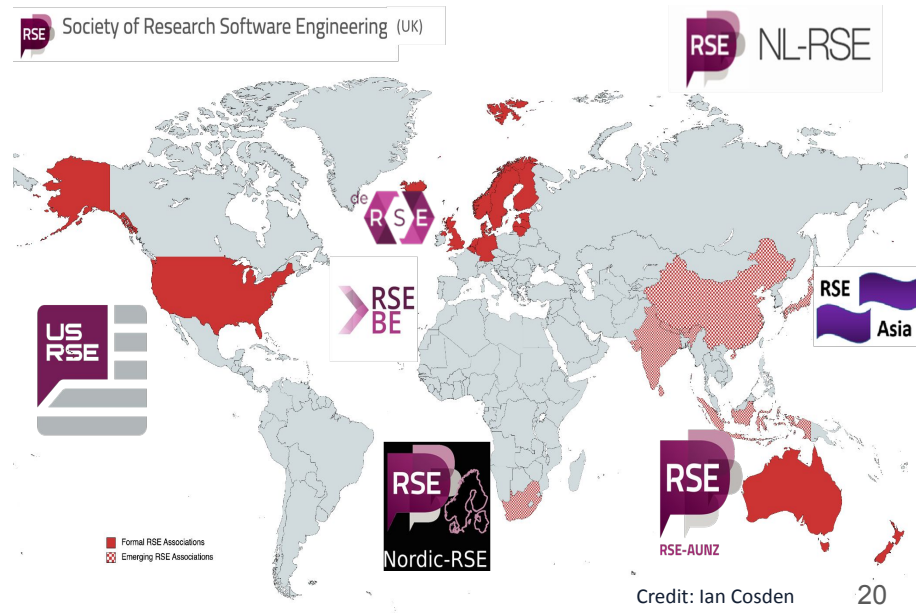
Casebooks Project Editor (Research Assistant / Associate) Climate Researcher (Research Associate) Clinical Study Programmer CoMPLEX Research Associate Computational Biologist / Bioinformatician Computational Scientist Computational Scientist in Computational Fluid Dynamics & Industrial Applications Computational Scientist in Structural Mechanics and Industrial Applications Computer Scientist Computer Vision Researcher Content Developer / Programmer Control Engineer-IMG - 3 posts CREATE Data Specialist Data Analyst Data Integration Coordinator Data Manager x3 Database and Software Engineer Database Manager / Researcher Database Programmer Digital Media Technician E-Learning Portal Manager (KTP Associate) e-Learning Systems Development Analyst e-Learning Systems Development Analyst (Moodle, SQL) E-Learning Web Developer E-Portfolio Learning Technologist Embedded Systems Engineer Engineering Technician Environmental Scientist EPSRC Studentship on Algorithmic Construction of Finsler-Lyapunov Functions Experimental Officer in Bioinformatics Experimental Psychologist Finance Assistant Gaia Alerts Software Developer Gaia Software Developer (Gaia UK Team) GIS Applications Specialist Graduate Programmer / Software Developer Graphics Programmer Health Data Manager / Scientist High Throughput Bioinformatician High Throughput Sequencing Bioinformatician (Windows) HIVE Manager / Co-ordinator HIVE Senior Researcher and Technical Lead Hydroinformatics Scientific Software Developer in Hydroinformatics Systems Developer Instrumentation Engineer Investigator Statistician IT Development Specialist IT Support Technician (Unix / Windows Systems) Knowledge Transfer Partnerships (KTP) Associate Knowledge Transfer Partnerships (KTP) Associate - Software Developer KTP Associate - Robot Vision Scientist (Research Fellow) KTP Associate (Fixed Term Contract for 24 months) KTP Associate (Precision Agriculture Data Analyst) KTP Associate â€” Graduate Web Developer KTP Associate: Electronics / Robotics Engineer Learning Technologist Leicester Respiratory BRIT IT Developer Linguist / Psycholinguist Maker Space Technician Marie Curie Early Stage Researcher Marie Curie Early Stage Researcher in Earth and Planetary Sciences Modelling Marine Earth Observation Scientists Medical Statistician Medical Statistician / Senior Medical Statistician Mobile Application Developer NASC IT Support - Programmer and Systems Administrator (Fixed Term Contract) PDRA on EU Project on Automated Multisensor Surveillance Planning Officer Policy Modeller 2011 Post - Doctoral Research Assistant INSTRON Post Doctoral Research Worker Post Doctoral Researcher in the application of Digital Technology Post-Doctoral Research Assistant in Simulation and Visualization Post-Doctoral Research Associate Post-Doctoral Research Associate (Pathogen Genomics) Post-Doctoral Research Fellow Postdoctoral Fellow - population genetics / evolutionary genetic Postdoctoral Fellow in Bioinformatics Postdoctoral Fellow in Cancer Therapeutics Postdoctoral Research Assistant Postdoctoral Research Associate Postdoctoral Research Fellow Postdoctoral Research Scientist Postdoctoral Researcher in Declarative (Logic and Functional) Programming Postdoctoral Researcher Postdoctoral statistician Postdoctoral Training Fellow - Statistical and Computational Genetics of Autism Principal / Senior Bioinformatician Principal Bioinformatician Product Development Engineer (Rail) Publishing Portal Web Developer Radio Frequency Engineer Reader in Computer Science Reporting Analyst Research (Software) Engineer Research Assistant Research Associate Research Fellow Research Image Data Manager, Biomedical Engineering Research Officer Research Officer â€” Social Protection Research postgraduate Research Programmer Research Scientist Research Scientist / Senior Research Scientist Research Scientist in Machine Learning and Computer Vision Research Software Developer Research Software Developer for the Herchel Smith Professor of Organic Chemistry Research Software Engineer Research Studentship Research Worker Researcher SAP Trainee Technical Analyst Scientific Officer with Michela Garofalo Scientist SEAH Studentship: Extracting epidemiological data from collections SEEG Data Archive Manager Senior / Research Associate in Clinical Integration and Image Analysis for Fetal Surgery Senior Analyst Programmer (Business Analysis) Senior Analyst / Programmer Senior Bioinformatician Senior Bioinformatician / Bioinformatician Senior Computational Statistician - Spatial Models Senior Data Acquisition Scientist / Data Acquisition Scientist Senior Data Manager Senior Database Administrator Senior IT Developer Analyst Senior Mathematical Modeller Senior Media Developer Senior Postdoctoral Researcher - Evolutionary and Computational Analysis of Infectious Disease (PhyloDynamics) Senior Research Assistant Senior Research Associate Senior Research Associate â€” Molecular Modelling & Simulation Senior Research Associate in Quantitative Clinical MRI Senior Research Fellow Senior Research Fellow / Research Fellow in Vibration Diagnostics and Prognostics / Digital Signal Processing Senior Research Laboratory Technician Senior Research Technician Senior Software Developer in Bioinformatics Senior Software Engineer / Software Engineer Senior Statistical Epidemiologist Senior Systems

Credit: Simon Hettrick

Today: a decade of Research Software Engineers



- Movement and term: Born in the UK
 - Late 2013 UKRSE Association forms with ~50 members
 - Now society, ~700 dues-paying members, ~5000-member community
- Also: Belgium, Germany, Netherlands, Nordic, Australia/New Zealand
- And US-RSE (<https://us-rse.org>), ~2000 members across universities, national labs, industry
- New associations forming in Africa & Asia
- Associations work on local issues collectively, and can coordinate



What makes software sustainable in general?



- **Useful** to a reasonable number of people
- Good external quality
 - Usability, correct, user documentation
- Good internal quality
 - Clean code, good software architecture, automated tests, developer's documentation
- Community of developers
 - Proprietary: paid by a company
 - Open source: paid staff and/or volunteers

Current challenges in RS and RSE in Cloud Networking



To produce high quality research software, 2 major sets of skills are required:

1. knowledge of best software engineering practices (automated testing, architectural and design patterns, agile methods, code quality, documentation, etc.)
 2. domain-specific knowledge: OS, scheduling, networking stacks, cloud, hardware, virtualization, security
- It's very hard to find professionals with good training in both of these aspects
 - Working in pairs is a good alternative

What's our homework?



- Changes in education
 - Valuing the production and sharing of high quality code
- Changes in scientific conferences
 - Making Open Science a 1st class citizen
- Changes in career paths (in universities and research centers)
- Changes in career promotion criteria
- Changes in funding agencies
 - And in ad-hoc reviews

Two ideas for funding agencies



1. For funded projects that produce software as an output, consider providing an additional grant at the end of the project specifically to invest in sustainability.
 - a. Example: 3 year research project + 1 to 3 year extension

2. New FAPESP call (November/2022):
 - a. Projects can request up to 10% of the budget for making the software robust, reusable, well documented, thus sustainable.

Let's go do that!

Practical guide to
Software
Management
Plans

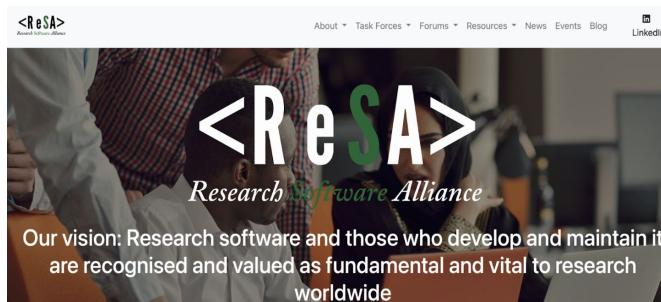


TABLE II
CONSIDERED SIMULATORS

Name / Author	Year	Onizky et al.	Meis et al.	Vogt et al.	Mihal et al.	Li et al.	Docquier et al.
MOSAIC [27]	2012	✓	✓	✓	✓	✓	
GECCO [28]	2012		✓	✓	✓	✓	
GridLAB-TS [29]	2009	✓	✓	✓	✓	✓	
Gridfey et al. [30]	2010		✓	✓	✓	✓	
INOCIS [31]	2006		✓	✓	✓	✓	
HELICS [32]	2017	✓	✓	✓	✓	✓	
GridSpace [33]	2014		✓	✓	✓	✓	
GridSim [34]	2011		✓	✓	✓	✓	
IBCN Smart Grid Simulator [35]	2011		✓	✓	✓	✓	
DACCOSIM [36]	2015		✓	✓	✓	✓	
MECS-CO [37]	2015		✓	✓	✓	✓	
FNCS [38]	2014		✓	✓	✓	✓	
INSPIRE [39]	2013		✓	✓	✓	✓	
PowerNet [40]	2011		✓	✓	✓	✓	
VPNET [41]	2011		✓	✓	✓	✓	
ORNL power system simulator [42]	2011		✓	✓	✓	✓	
SGC [43]	2011		✓	✓	✓	✓	
Bergmann et al. [44]	2011		✓	✓	✓	✓	
Konner et al. [45]	2015		✓	✓	✓	✓	
BoFit	2010		✓	✓	✓	✓	
Ahmed et al. [46]	2015		✓	✓	✓	✓	
GridIQ [47]	2010		✓	✓	✓	✓	
Wang et al. [48]	2012		✓	✓	✓	✓	
Rajendran et al. [49]	2012		✓	✓	✓	✓	
OpSim [50]	2015		✓	✓	✓	✓	
Phileas [51]	2005		✓	✓	✓	✓	
Verdi [52]	2015		✓	✓	✓	✓	
SMR [53]	2013		✓	✓	✓	✓	
Andon et al. [54]	2011		✓	✓	✓	✓	
Munachi et al. [55]	2015		✓	✓	✓	✓	
Ding et al. [56]	2016		✓	✓	✓	✓	
Bhor et al. [57]	2014		✓	✓	✓	✓	
OPPHUS [58]	2016		✓	✓	✓	✓	
DACCOSIM-NG [59]	2019		✓	✓	✓	✓	
CYDER [60]	2018		✓	✓	✓	✓	
Notaro et al. [61]	2007		✓	✓	✓	✓	
Levesque et al. [62]	2012		✓	✓	✓	✓	
Zhu et al. [63]	2011		✓	✓	✓	✓	
MAPKEY [64]	2011		✓	✓	✓	✓	
SmartGridLab [65]	2012		✓	✓	✓	✓	
SCORE [66]	2012		✓	✓	✓	✓	
Bruckner et al. [67]	2009		✓	✓	✓	✓	
Teng et al. [68]	2010		✓	✓	✓	✓	
ScourSim [69]	2000		✓	✓	✓	✓	
TrueTime [70]	2002		✓	✓	✓	✓	
Proxier [71]	2003		✓	✓	✓	✓	
VisualKone [72]	2004		✓	✓	✓	✓	
Stilhuys et al. [73]	2006		✓	✓	✓	✓	
Tarfehan et al. [74]	2007		✓	✓	✓	✓	
Kanabar et al. [75]	2010		✓	✓	✓	✓	
Juarez et al. [76]	2012		✓	✓	✓	✓	
Luon et al. [77]	2016		✓	✓	✓	✓	
STARS [78]	2019		✓	✓	✓	✓	
Ngo et al. [79]	2014		✓	✓	✓	✓	

Prof. Fabio Kon - kon@ime.usp.br
IME - University of São Paulo

Mininet-WiFi: Emulating software-defined wireless networks

RR Fontes, S Afzal, SHB Brito, MAS Santos, CE Rothenberg

2015 11th International conference on network and service management (CNSM) - 483 citations in 11/2024

Ramon dos Reis Fontes desenvolveu durante o doutorado sob orientação do Prof. Christian Rothenberg (Unicamp), o Mininet-WiFi - <https://github.com/intrig-unicamp/mininet-wifi>. O código do Mininet-WiFi é aberto, já passou de 800 citações (diretas e indiretas) e até hoje mantenho uma lista de discussão com centenas de membros de todo o mundo (maior parte massiva de estrangeiros). Outros resultados bacanas do desenvolvimento deste emulador foram contribuições para o Linux Kernel (<https://patchwork.kernel.org/project/linux-wireless/list/?series=&submitter=176431&state=3&q=&archive=&delegate=>) - se usas Linux, tem código meu em seu computador, mesmo que esteja adormecido :).