

Alexander Kononov

Centre for Interdisciplinary Research in Computational Algebra  
School of Computer Science  
University of St Andrews

# CoDiMa CCP: 5 years on



Manchester, 23 May 2019



# Collaborative Computational Projects

<http://www.ccp.ac.uk/current.html>

Home

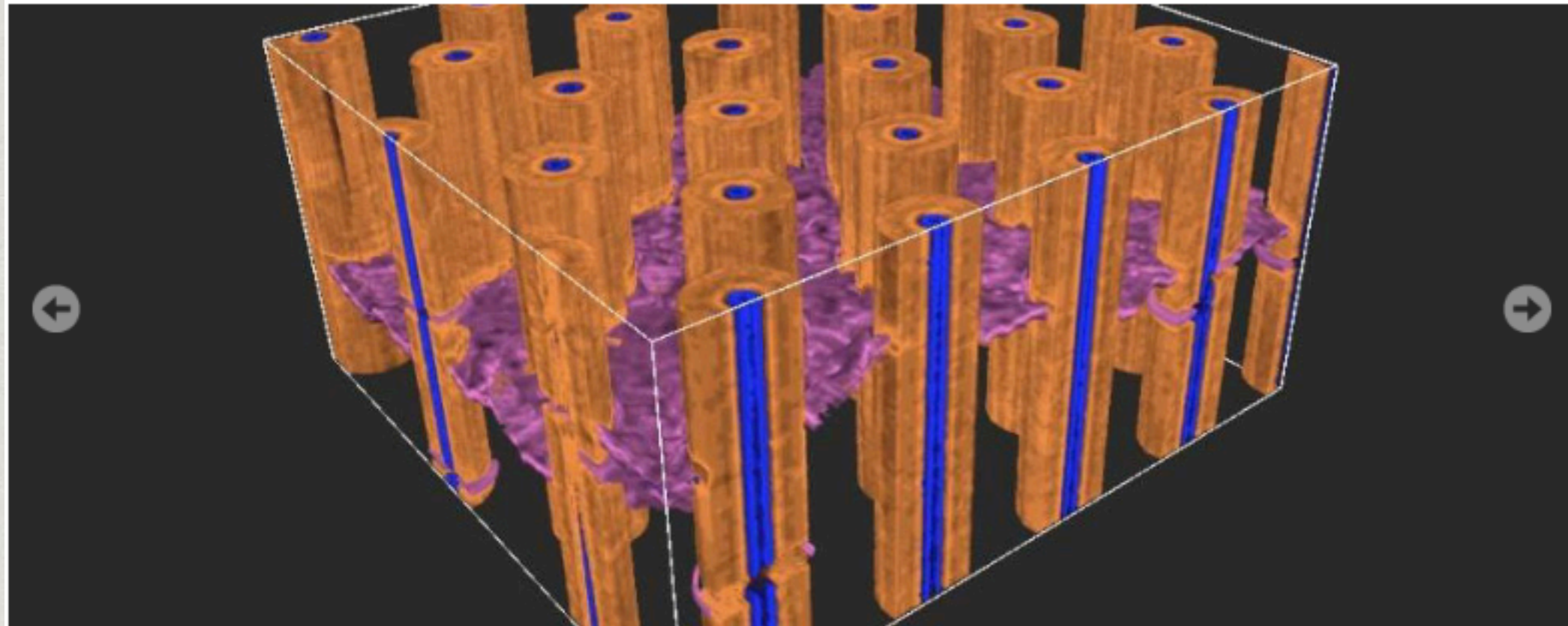
About CCPs

Current CCPs

Former CCPs

Steering Panel

Contact



## About the CCPs

The CCPs enrich UK computational science and engineering research in various ways. They provide a software infrastructure on which important individual research projects can be built. They support both the R&D and exploitation phases of computational research projects. They ensure the development of software which makes optimum use of the whole range of hardware available to the scientific community, from the desktop to the most powerful national supercomputing facilities. The training activities of CCPs have been outstandingly successful, benefiting several hundred students and post-doctorates each year.

The main activities of the CCPs are to:

- Carry out flagship code development projects
- Maintain and distribute code libraries

Flagship projects represent innovative software developments at the leading edge of a CCP's area of science or engineering. They normally last for three years and may support a PDRA associated with the project. At the end of a flagship project, the resulting software usually becomes part of the code library. CCPs maintain, distribute and develop the new code according to demand from member and user research programmes.



2014 call with the focus on networking and widening participation activities

## **Collaborative Computational Projects: Networking and Core Support**

**Call type: Invitation for proposals**

**Closing date: 16:00 07 October 2014**

**Related themes: Engineering, ICT, Mathematical sciences, Physical sciences, Research infrastructure**

### **Summary**

In both the recently published EPSRC E-infrastructure roadmap and the EPSRC Software as an Infrastructure strategy the importance of software development and the need to invest in people and training in this area has been strongly highlighted.

Collaborative Computational Projects (CCPs) bring together the major UK groups in a given field of computational research to tackle large-scale scientific software development projects, maintenance, distribution, training and user support. They play an important role in EPSRC's ongoing ability to deliver its Software as an Infrastructure strategy and as community based networks and projects they provide a focal point for communities to identify their scientific software requirements and take a strategic approach to software support in a particular field.

# Partners on the CoDiMa proposal

## **University of St Andrews**

- Steve Linton
- Alexander Konovalov

## **University of Birmingham**

- Sergey Shpectorov

## **University of Glasgow**

- Phil Trinder

## **Heriot-Watt University**

- Jim Howie
- Hans-Wolfgang Loidl

## **University of Leicester**

- Rick Thomas

## **University of Manchester**

- Alexandre Borovik

## **University of Newcastle**

- Andrew Duncan

## **University of Oxford**

- Dmitrii Pasechnik

## **Queen Mary, University of London**

- Leonard Soicher

## **University of Warwick**

- John Cremona
- Derek Holt



# Success!

Computational  
Discrete  
Mathematics

- \* CoDiMa is the Collaborative Computational Project (CCP) in the area of Computational Discrete Mathematics, supported by the EPSRC (EP/M022641/1).
- \* CoDiMa is centred on two open source software systems: GAP and SageMath which are widely used for research and teaching in abstract algebra, number theory, cryptography, combinatorics, graph theory, coding theory, optimisation and search, among other areas.
- \* The CCP aims to support the ecosystem of users, extenders and developers of these systems and encourage best practice in their use, and to support the more rapid uptake of new features such as parallel programming support.
- \* The project runs for 5 years starting from March 1st, 2015.



Training and  
dissemination  
activities



Underpinning  
software  
maintenance

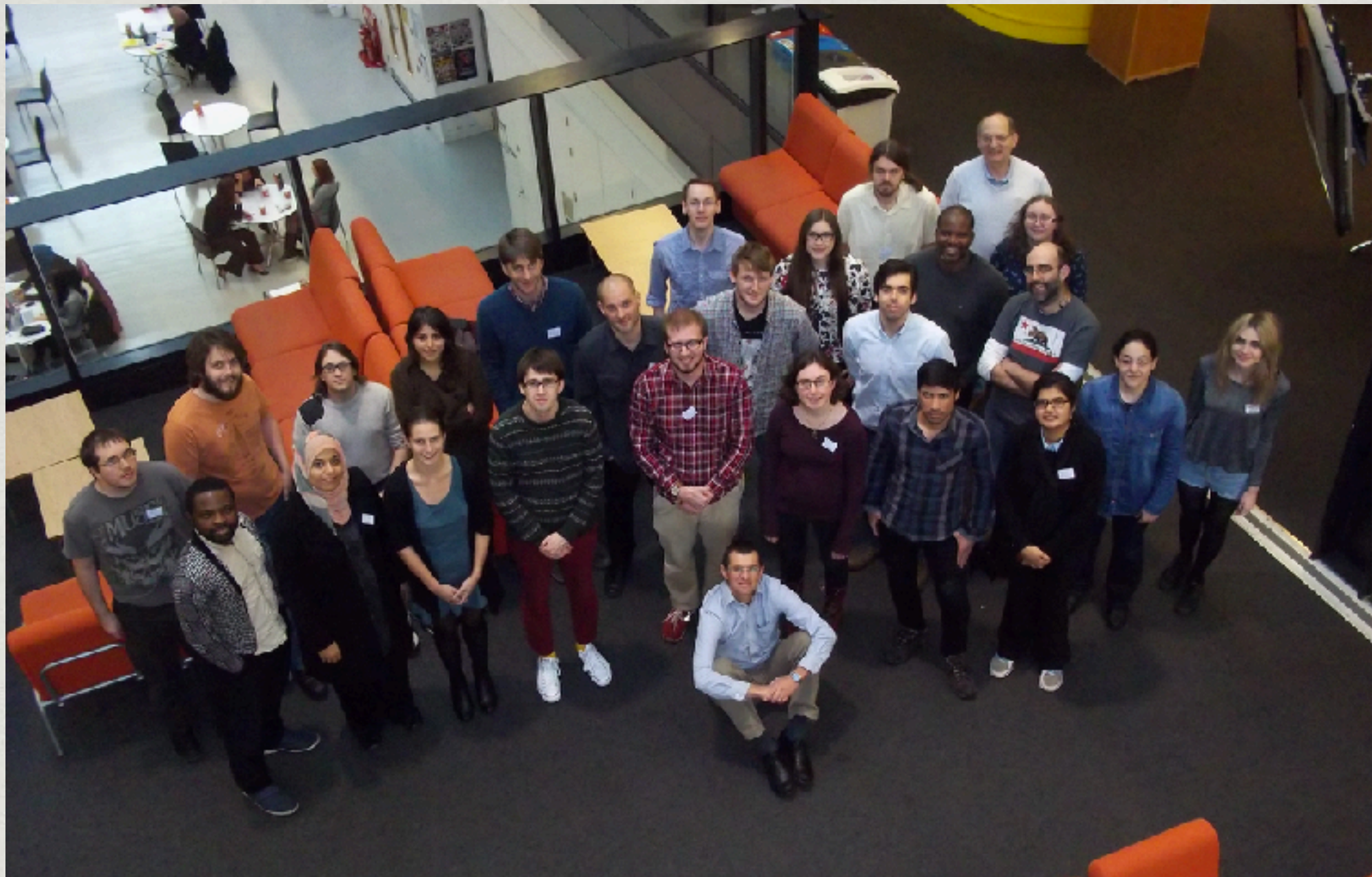


Collaborative  
visits for software  
development and  
computational  
research



# Training

- \* First CoDiMa Training School in Computational Discrete Mathematics (University of Manchester, November 16th-20th, 2015) <https://www.codima.ac.uk/school2015/>











# THE CARPENTRIES

We teach foundational coding and data science skills to researchers worldwide.



## What we do

The Carpentries teach foundational coding, and data science skills to researchers worldwide. Software Carpentry, Data Carpentry, and Library Carpentry workshops are based on our lessons. Workshop hosts, Instructors, and learners must be prepared to follow our [Code of Conduct](#).

[More >](#)



## Who we are

We are a diverse, global community of [volunteers](#). Our community includes [Instructors](#), [helpers](#), [Trainers](#), [Maintainers](#), [Mentors](#), community champions, [member organisations](#), supporters, workshop organisers, [staff](#) and a whole lot more.

[More >](#)



## Get involved

See all the [ways you can engage](#) with the Carpentries. Get information about upcoming events such as workshops, meetups, and discussions from our [community calendar](#), or from our twice-monthly [newsletter](#), *Carpentry Clippings*. Follow us on [Twitter](#), [Facebook](#), and [Slack](#).

[More >](#)

## Subscribe to our Newsletter "Carpentry Clippings"

Events, Community Updates, Teaching Tips, in your inbox, twice a month

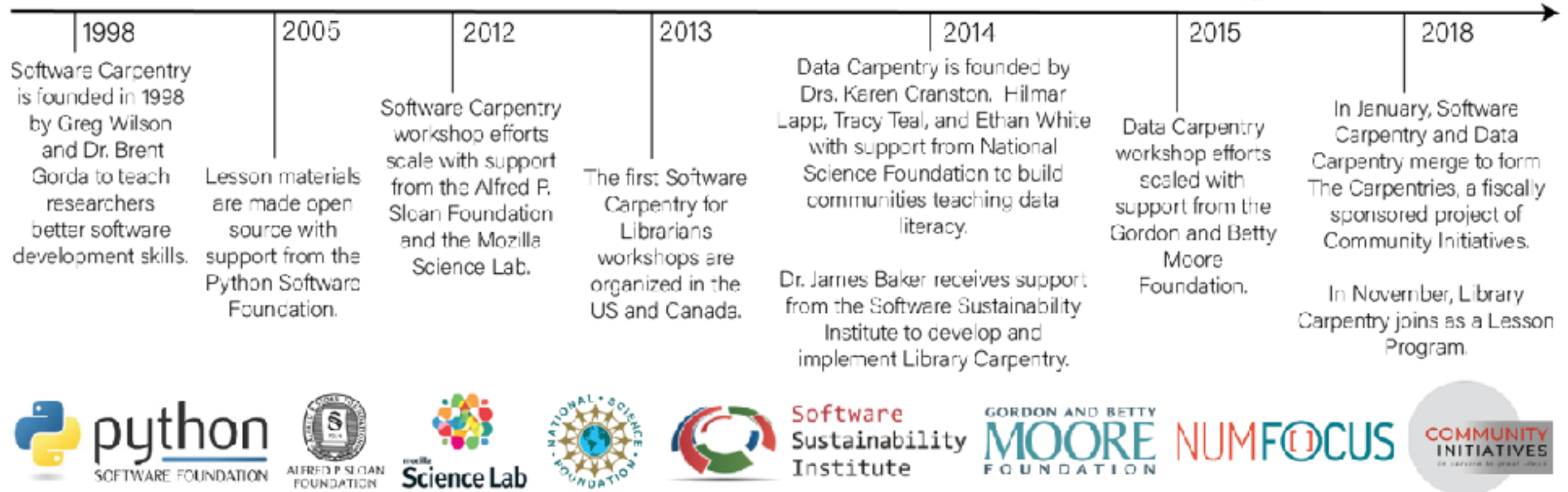
Subscribe



# A VERY BRIEF HISTORY OF



# THE CARPENTRIES



## Software Carpentry

- for those who need to program more effectively to solve their computational challenges
- not domain-specific
- modular: each Software Carpentry lesson is standalone

## Data Carpentry

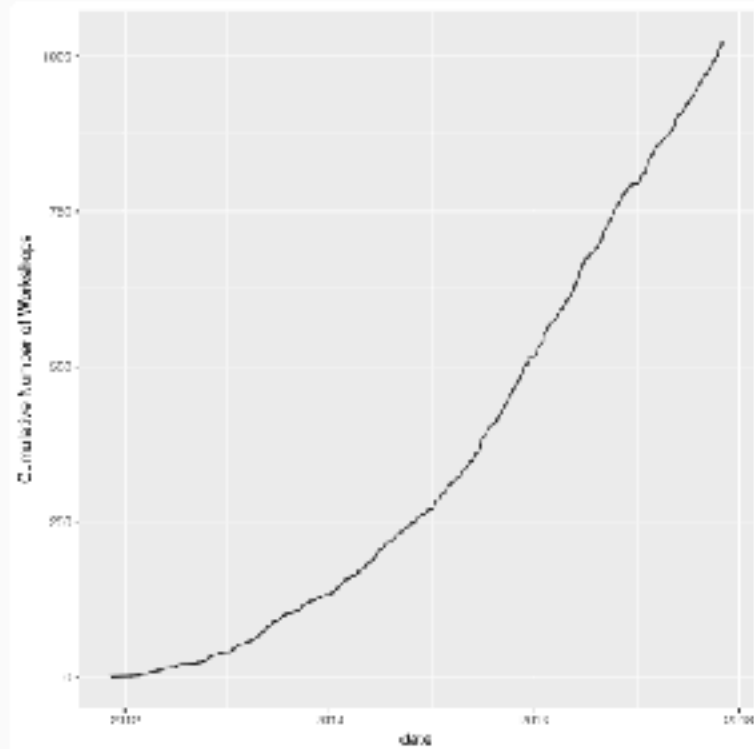
- domain-specific
- focus on best practices surrounding data
- present a full curriculum centred around a single data set

## Library Carpentry

- for those in library- and information-related roles
- focus on best practices in structuring data
- modular: each Library Carpentry lesson is standalone

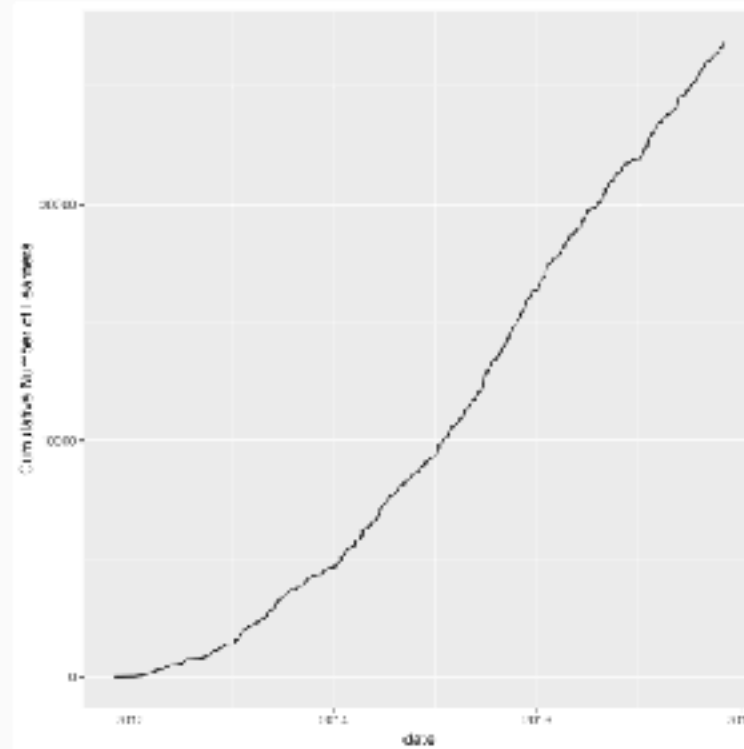


Workshops



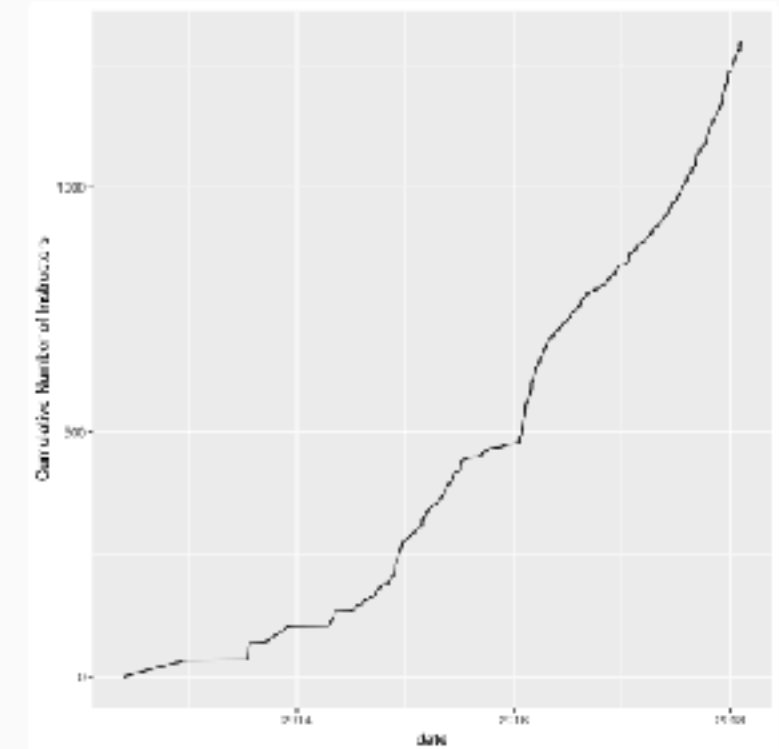
n=1023

Learners



n=26,872

Instructors



n=1297

Stats updated on 2018-02-05

**ONLY THE SOFTWARE CARPENTRY  
TRAINED MORE THAN 27,000 LEARNERS WORLDWIDE SINCE 2012**

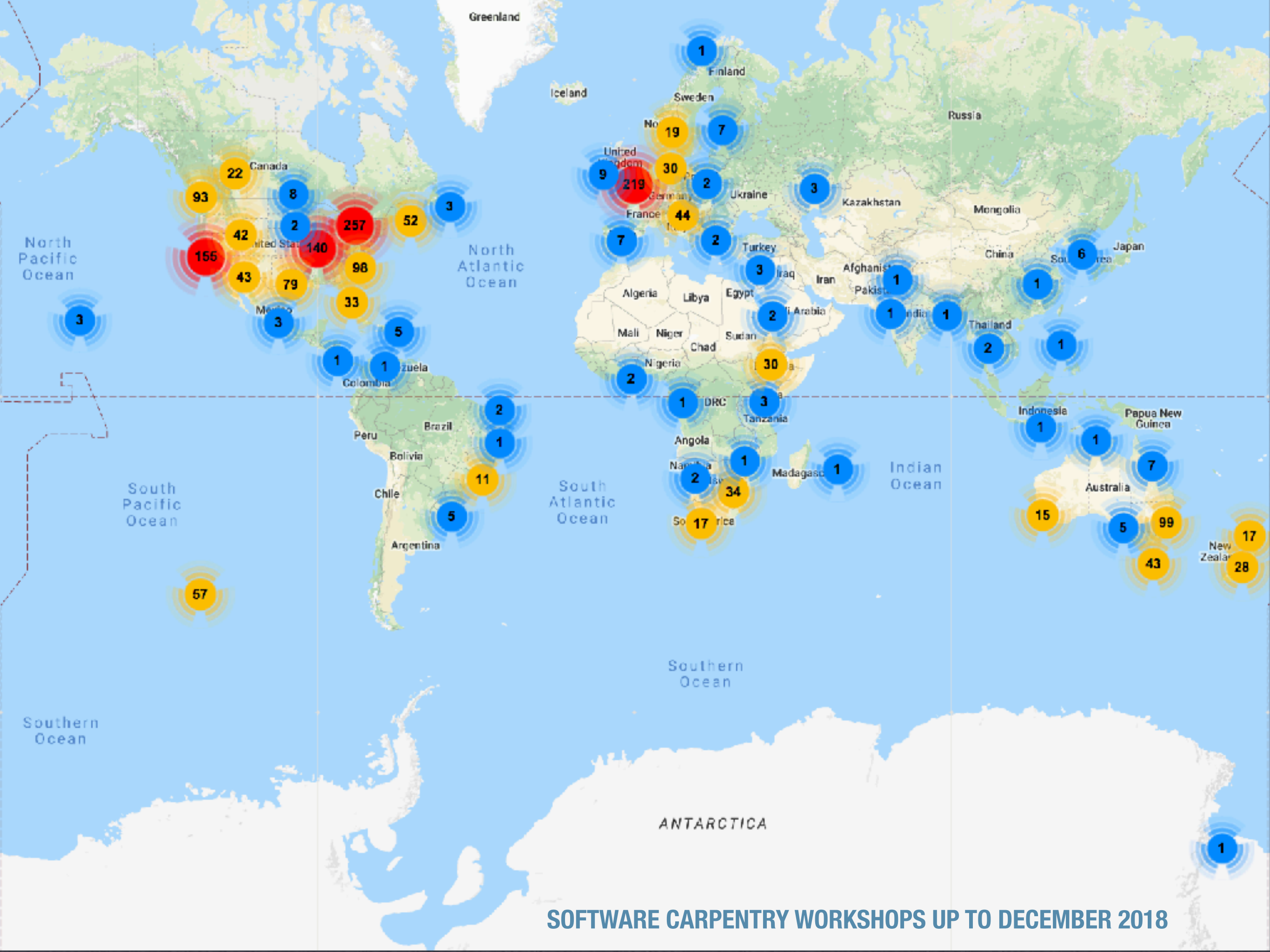
WORKSHOPS

1023

INSTRUCTORS

1297





























SOFTWARE CARPENTRY WORKSHOPS UP TO DECEMBER 2018



# Our Core Lessons in English




## SOME OF THE SOFTWARE CARPENTRY LESSONS

Lesson	Site	Repository	Reference	Instructor Guide	Maintainer(s)
The Unix Shell					<a href="#">Gabriel Devenyi</a> , Colin Morris, <a href="#">Will Pitchers</a>
Version Control with Git					<a href="#">Ivan Gonzalez</a> , <a href="#">Daisie Huang</a> , Nima Hejazi, Katherine Koziar, Madicken Munk
Programming with Python					<a href="#">Trevor Bekolay</a> , <a href="#">Valentina Staneva</a> , Anne Fouilloux, Maxim Belkin, Mike Trizna
Plotting and Programming in Python					<a href="#">Nathan Moore</a> , Allen Lee, Sourav Singh, Olav Vahtras
Programming with R					<a href="#">Daniel Chen</a> , Katrin Leinweber, Diya Das
R for Reproducible Scientific Analysis					<a href="#">Thomas Wright</a> , <a href="#">Naupaka Zimmerman</a> , <a href="#">Jeffrey Oliver</a> , David Mawdsley

[The lesson template](#) (used to build each lesson page) and the [workshop template](#) (used to build each scheduled workshop's website) are available on GitHub.

## Contributed

These are Carpentries style lessons that have been contributed by community members and are not part of Software Carpentry's official offerings. If you have a lesson to include here, please [contact us](#) so we can review your lesson for inclusion here. To be considered, lessons should use our template, be hosted in an original GitHub repo (not a fork), be offered under an open license, have been taught at least once, and be actively maintained (with at least one lead maintainer).

Lesson	Site	Repository	Maintainer(s)
Testing and Continuous Integration with Python			<a href="#">Katy Huff</a>
Programming with GAP			<a href="#">Alexander Konovalov</a>



	Setup	Download files required for the lesson
00:00	1. First session with GAP	Working with the GAP command line
00:40	2. Some more GAP objects	Further examples of immediate and positional objects and operations with them
01:00	3. Functions in GAP	Functions as a way of code re-use
01:55	4. Using regression tests	Test-driven development
02:45	5. Small groups search	Modular programming: putting functions together How to check some conjecture for all groups of a given order?
03:40	6. Attributes and Methods	How to record information in GAP objects
04:30	Finish	



# Training

- \* Second CoDiMa Training School in Computational Discrete Mathematics (International Centre for Mathematical Sciences, Edinburgh, October 17th-21st, 2016) <https://www.codima.ac.uk/school2016/>





# Training

- \* Second CoDiMa Training School in Computational Discrete Mathematics (International Centre for Mathematical Sciences, Edinburgh, October 17th-21st, 2016) <https://www.codima.ac.uk/school2016/>







## Dr. Alexander Konovalov

Senior Research  
Fellow in  
Computational  
Algebra

 University of St Andrews

 Github

 Stackoverflow

 Google Scholar

 ORCID

 Impactstory

# Publishing Software Carpentry lesson on GAP

🕒 5 minute read

 **Published:** November 22, 2016

Following the [2nd CoDiMa training school](#), I have published the [Software Carpentry lesson on GAP](#) via Zenodo: see [10.5281/zenodo.167362](https://zenodo.org/record/105281/files/167362). The lesson is based on the problem of determining an average order of an element of a finite group, and finding examples of groups for which the average order of their elements is an integer. First I have heard about this problem when Steve Linton used it in a talk in order to quickly demonstrate some GAP features to a general scientific audience. I have tried to expand on it in my talk in Newcastle in May 2015 (see the blog post [here](#)), and decided to proceed with it.

Indeed, the problem of determining an average order of the element of the group is simple enough to not to distract learners too much from the intended learning outcomes of the lesson. An undergraduate algebra course is sufficient for its understanding. Moreover, those not familiar with the group theory still should be able to follow the lesson just by grasping the idea that there is a mathematical structure called group, and we need to find an

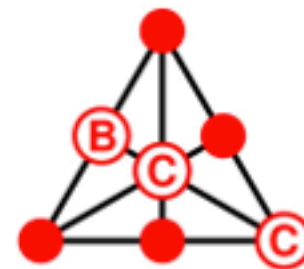
# Training

- \* GAP tutorial as a part of the Summer School on Finite Geometry in Brighton on 26th-30th June 2017  
<https://ftmakroglu05.github.io/Finite-Geometry-School/>
- \* Given by Chris Jefferson (St Andrews)



## Summer School on Finite Geometry

**26 - 30 June 2017, Brighton, UK**



[Home](#)

[Programme](#)

[Registration](#)

[Accommodation](#)

[Short Talks](#)

[Venue](#)

[Social Events](#)

[Participants](#)



# Training

- \* GAP tutorial on August 13th-14th, 2017 – a satellite event to Groups St Andrews 2017 in Birmingham  
<https://www.codima.ac.uk/gsta2017/>





I practise more examples.  
This session help me  
How to create a function.

Many thanks

good example  
average order - good  
to write on real  
problems

very clear explanations  
good structure and  
layout.  
at a good pace for me  
at a good level for me  
I learnt a lot  
reps matter, attributes,  
properties, nps... etc

Most useful  
to help install  
GAP on my computer  
and get it working

The sections on  
regression tests and  
the examples on small  
groups were very interesting!

Very useful  
course!  
Many examples  
of calculations  
in groups.

1. Programming with GAP
2. Attributes and Methods
3. Using regression tests

Variadic functions  
Attributes  
Tracing

All examples were very  
useful and so clear

Useful. - learning how to get  
time a program take.  
- learning how to print information  
as the program is running.  
- test files  
- access help in browser instead  
of gap terminal.

There was some  
technical problems  
in my computer, but  
we fixed it.

Thanks

The TestOrder function  
was presented a bit too fast.  
I didn't understand the part about  
collection very well.

Maybe more examples for  
the section about attributes  
and Methods

- Is it possible to know  
how calculate kernel of  
morphisms? Especially rep.
- Calculate interaction  
quotient

THANKS!



# Training

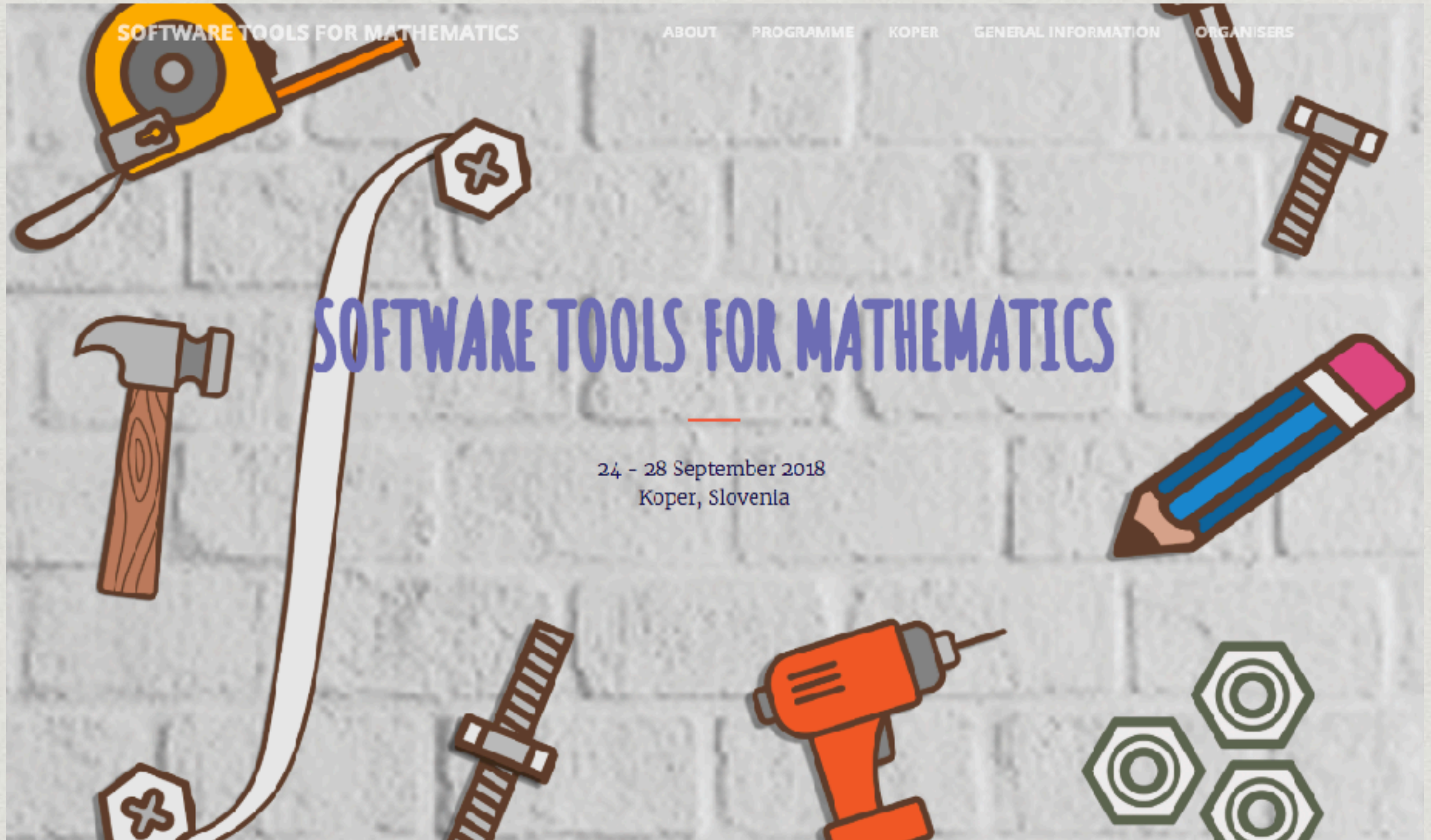
- \* GAP tutorial on July 16th and 20th, 2018 – a satellite event to the 20th Postgraduate Group Theory Conference (PGTC 2018) in St Andrews <https://www.codima.ac.uk/pgtc2018/>





# More training

- \* GAP Tutorial at “Software Tools for Mathematics” workshop (24-28 September, Koper, Slovenia) <http://stm.famnit.upr.si/>







SUMMER SCHOOL "GAP IN ALGEBRAIC RESEARCH", AACHEN, NOVEMBER 2018



# More training in 2019

- \* GAP tutorial for the participants of the weekly “Semigroups and Digraphs in GAP meeting in St Andrews (February - May 2019) - AlexK together Michael Torpey
  - \* <https://github.com/gap-packages/Semigroups/wiki/Wednesday-Meeting-Minutes>
- \* GAP tutorial on July 22nd and 26h, 2019 – a satellite event to the 21st Postgraduate Group Theory Conference (PGTC 2019) in Birmingham - AlexK together Michael Torpey
  - \* <https://www.codima.ac.uk/pgtc2019/>
- \* GAP Tutorial in GAP-Singular School (August 2019, Lambrecht, Germany) - Michael Torpey
  - \* <https://opendreamkit.org/meetings/2019-04-02-GAP Singular Meeting/>



# Workshops

- \* **First Joint GAP-SageMath Days** (University of St Andrews, January 18th-22nd, 2016, with a GAP coding sprint on January 13th-16th)
  - \* <https://www.gapdays.de/gap-sage-days2016/>
- \* Followed by the Workshop “**Knowledge representation in mathematical software and databases**” organised by the Horizon 2020 OpenDreamKit project in St Andrews on January 25th-27th, 2016
  - \* <https://opendreamkit.org/2015/12/08/WP6StAndrewsMeeting/>



# Workshops

- \* Workshop “**Computational Mathematics with Jupyter**” (International Centre for Mathematical Sciences, Edinburgh, January 16th-29th, 2017), organised jointly with the OpenDreamKit project
- \* <https://opendreamkit.org/meetings/2017-01-16-ICMS/>



# Computational Mathematics with Jupyter





# This workshop!

- \* CoDiMa workshop 2019 (University of Manchester, May 23rd-25th, 2019)
- \* <https://circa-standrews.github.io/codima2019workshop/>
- \* Group photo today after lunch!



# Collaborative visits programme

- \* For members of the UK computational discrete mathematics community, CoDiMa may have an opportunity to provide full or partial travel support for:
  - \* short research visits for collaborative research and software development work (as a host or as a visitor)
  - \* attending conferences and workshops to promote mathematical software there by means of lectures and tutorials, and/or to learn crucial details necessary to implement mathematical algorithms
- \* We consider applications every two months. In 2019, the remaining deadlines are June 15th, August 15th, October 15th and December 15th.
- \* The main evaluation criteria are the focus of the proposed activity on the research software within the remits of the CoDiMa project, and its value for the UK computational discrete mathematics community
- \* Welcome to discuss intended application informally in advance
- \* Reporting and reimbursement after the visit
- \* See <https://www.codima.ac.uk/research-visits-programme/>



# Collaborative visits programme

Some reports are at <https://www.codima.ac.uk/category/visits/>



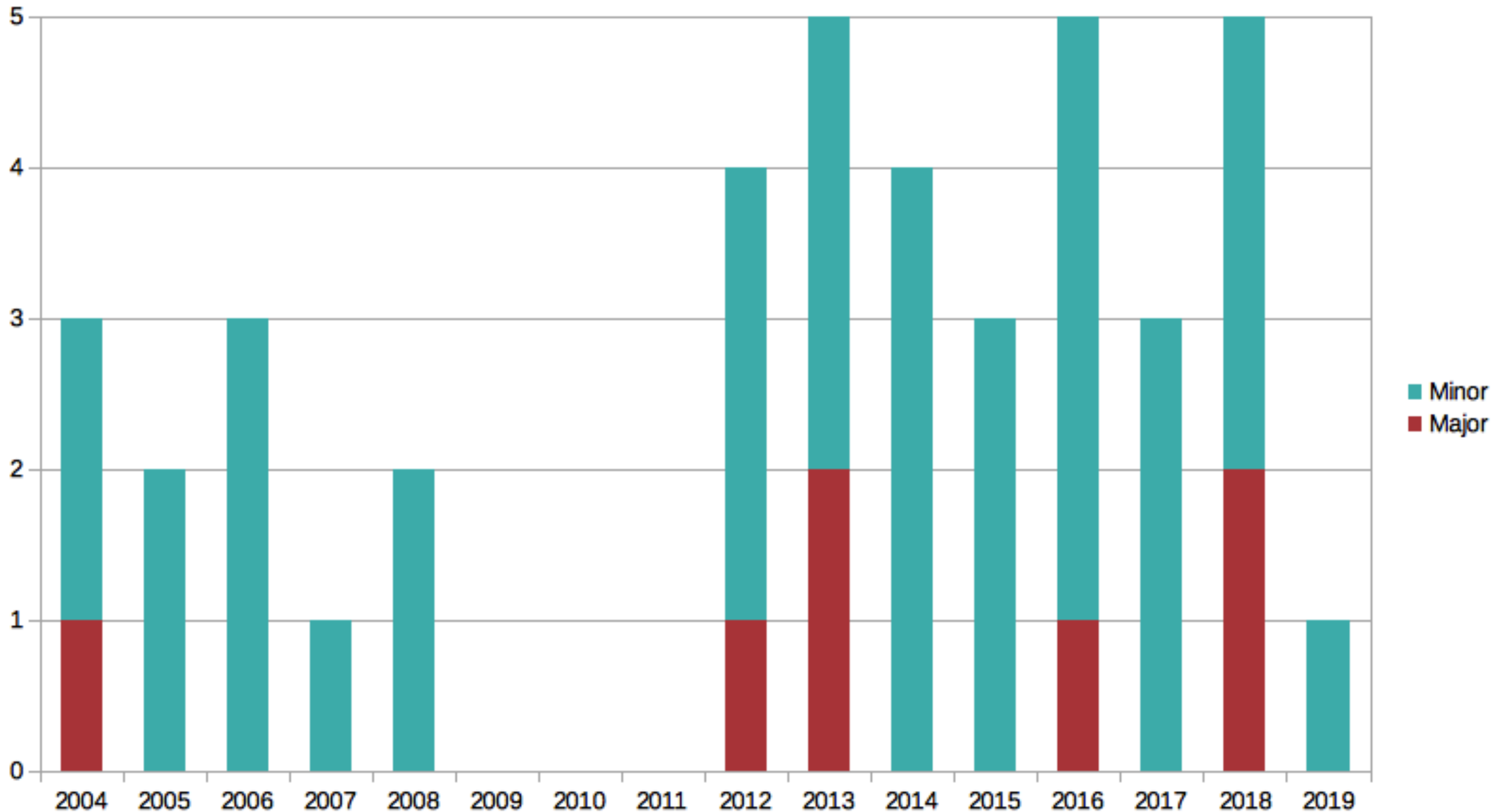


## Improved reliability of GAP and building a healthy package ecosystem

- \* Regular release cycle: 30 releases (6 major) since 2012
- \* Redistributed packages up 48% from 98 to 145
- \* Released in 2017 or later: 92%
- \* Packages with regression tests increased from 40 to 110
- \* Package authors community size up at least by 50%

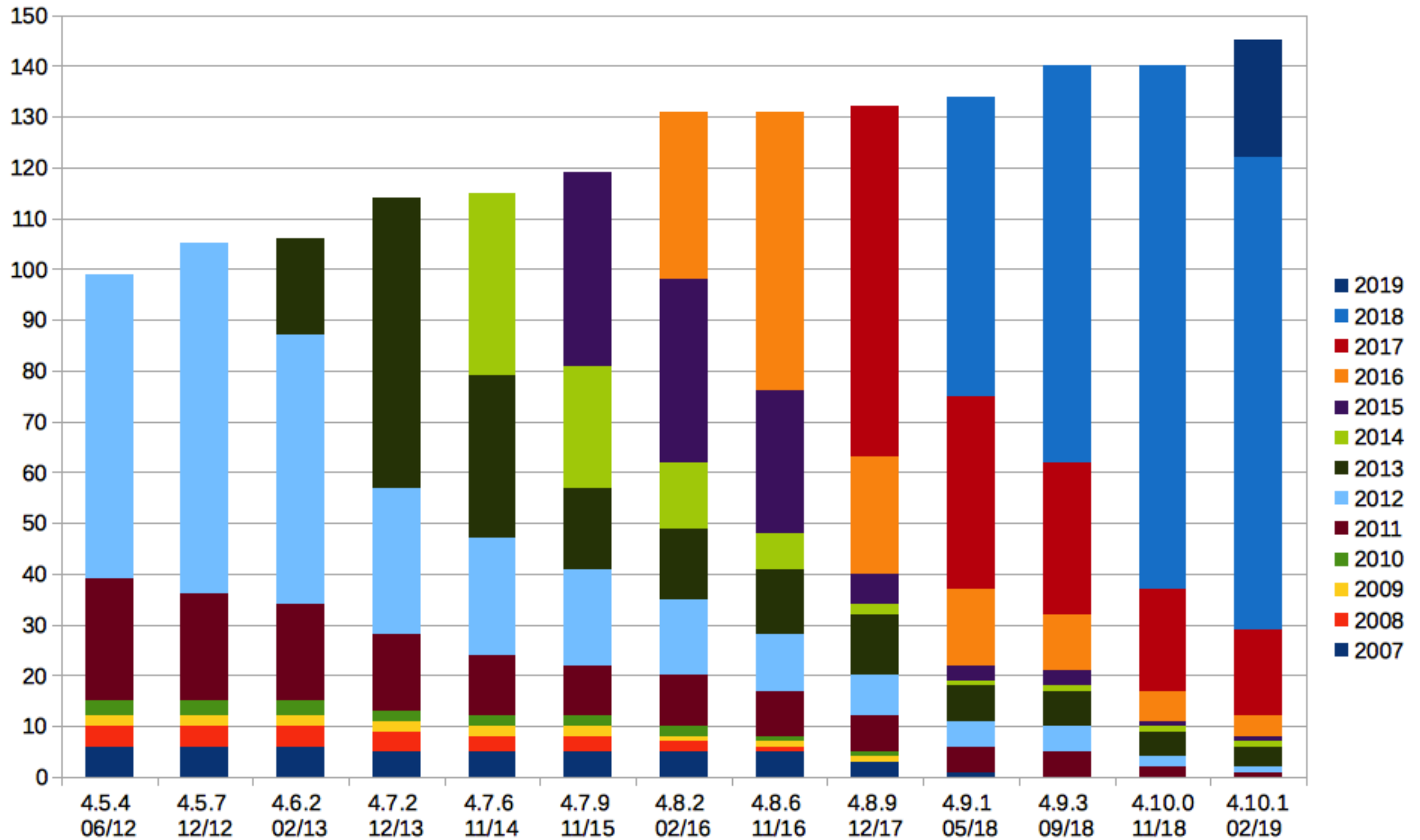


# Number of major and minor GAP releases per year since GAP 4.4 release



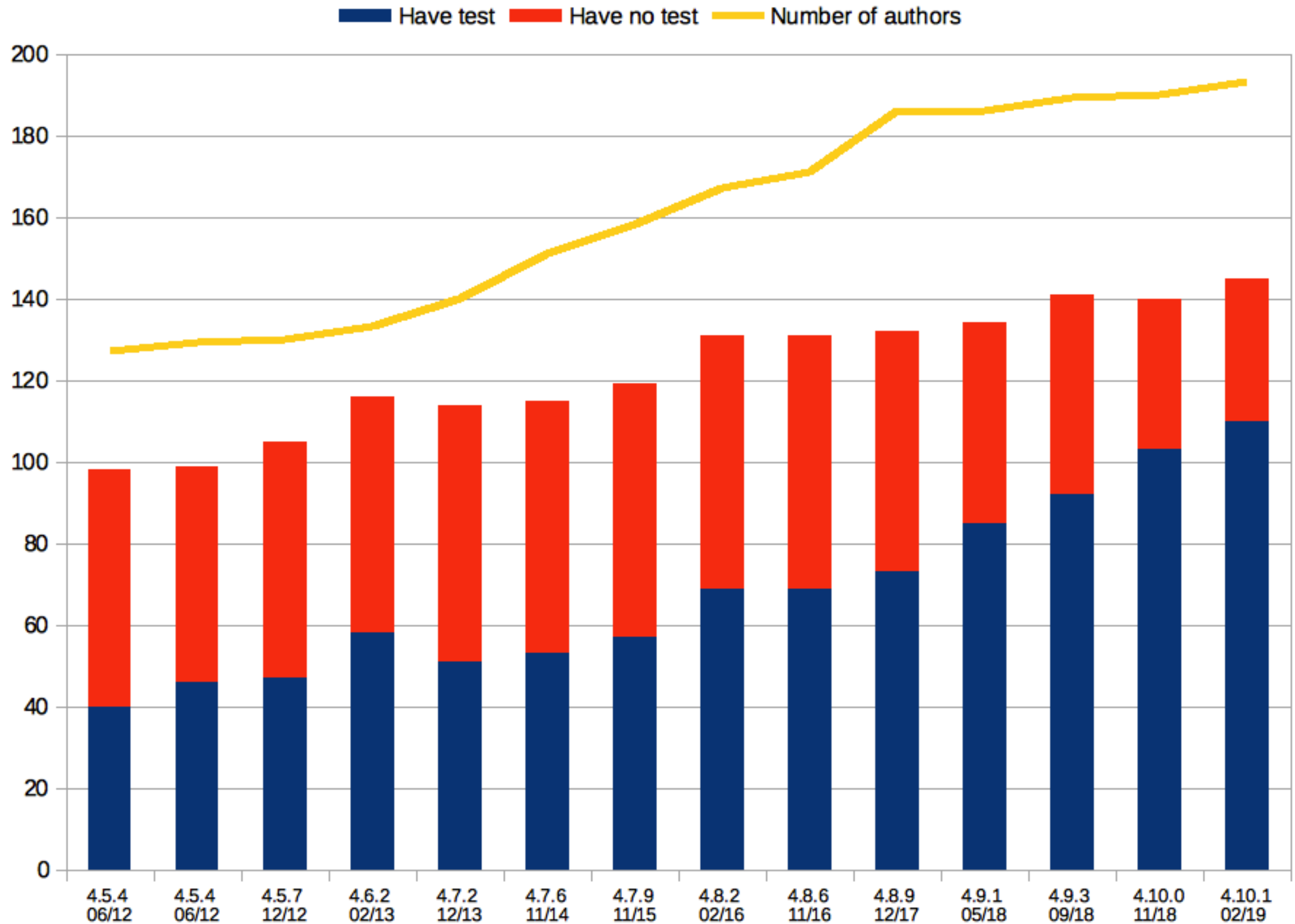


# Number of packages and their release year





# Number of packages, their authors, and packages with tests





# Tools to support package authors

- \* PackageMaker: (by Max Horn)
  - \* <https://github.com/gap-system/PackageMaker>
- \* Example package:
  - \* <https://github.com/gap-packages/example>
- \* ReleaseTools (by Max Horn):
  - \* <https://github.com/gap-system/ReleaseTools>
- \* GitHubPagesForGAP (by Max Horn):
  - \* <https://github.com/gap-system/GitHubPagesForGAP>
- \* Docker containers:
  - \* <https://hub.docker.com/r/gapsystem/>
- \* Standard setup for Travis CI and CodeCov (by Max Horn et al.)

✓	datastructures	# 610	
✓	smallgrp	# 384	
✗	gap-docker-pkg-tests-master	# 17	
✓	MathInTheMiddle	# 44	
✓	Semigroups	# 1459	



**xmodalg**

Latest commit 36 minutes ago



**MathInTheMiddle**

Latest commit 20 hours ago by mtorpey



**curlInterface**

Latest commit a day ago by mtorpey



**groupoids**

Latest commit a day ago