

An abstract background on the left side of the slide, composed of numerous overlapping triangles in various shades of blue, green, yellow, orange, and pink, creating a mosaic-like effect.

Biostatistics Software Engineering in Pharma

Turing-Roche Knowledge Sharing Series, 12 Dec 2022

Daniel Sabanés Bové

Statistical Engineering Lead
Data Science Acceleration, Pharma Product Development Data Sciences, Roche

12 December 2022 | public

Acknowledgements

Thanks to everyone involved!

Data Science Acceleration:

Sarah McGough

Chris Harbron

James Black

Ryan Copping

Statistical Engineering Team:

See slide later

Additional Colleagues:

Juha-Pekka Perttola

Ross Farrugia

Thomas Neitmann

Vincent Shen

Jaime Pires

Coline Zeballos

Ning Leng

Kieran Martin

Dinakar Kulkarni

Pawel Rucki

Turing:

Vicky Hellon

Ed Chalstrey

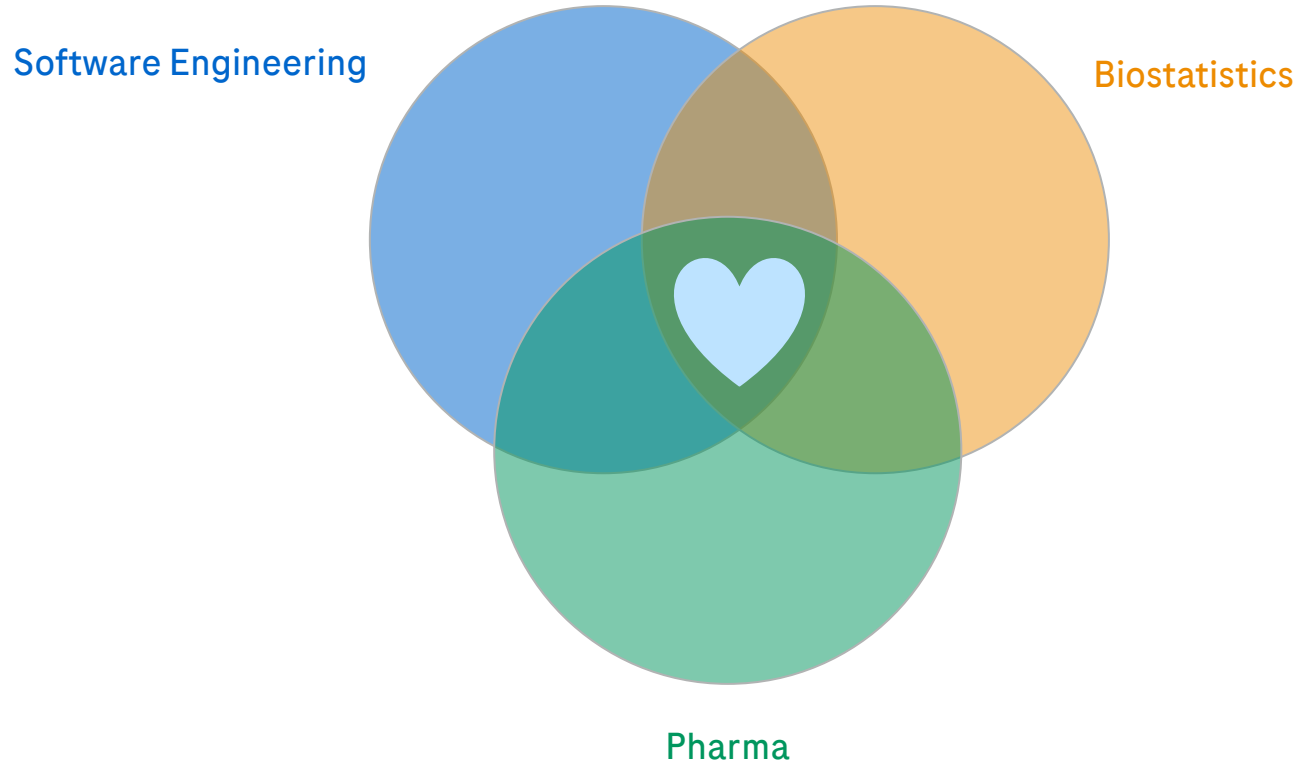


Table of contents

1. **Introduction**
2. **Statistical Engineering team**
3. **Open Source collaboration**

Introduction

How can Biostatistics look like in Pharma?

Different roles, but all work with biological data

- **Different roles:**

- Biostatistician
- Statistical programmer
- Real world data scientist
- Data scientist
- Patient reported outcomes scientist
- etc.

- Matched by **different departments**

- Typically part of Biometrics, Biometry, Data Sciences etc.

- **Commonality:**

- Working with biological (and often medical) data (from clinical trials or other sources)
- **Practically analysing these data sets** or planning experiments to generate them
- Collaborating with each other and importantly other disciplines (in particular scientists, medical doctors)



How do we analyse the data?

... not by hand with our calculators!



- We use **software**:
 - Proprietary: SAS, Stata, SPSS, ...
 - Open source: R, Python, ...
- We **need to write code** in most cases
 - Data wrangling takes a lot of time
 - Data set derivation in specific formats (sometimes other colleagues doing this for us)
 - Trying different statistical methods with different options
 - Setting up simulation studies (for planning)
 - Summarising results in tables or plots
- Bottom line:
 - There is **no (applied) biostatistics without writing code**

Roche Pharma is moving towards open source solutions

Clinical Reporting with R will be default from 2023

- At Roche Pharma Development Data Sciences:
 - For all of our **new molecules from 2023 onwards**, **R based solutions will form the default** reporting workflow for any GxP relevant & exploratory work.
 - This means that at some point in the future all of our clinical studies will be analyzed using R (with other tools supporting)
 - We **continue to offer SAS, Python etc.** to enable our Statisticians and Analytical Data Scientists to choose the best tool for the task at hand.
- Motivation:
 - Allow easier **blending of data science roles**
 - **New graduates know R** already, but no proprietary languages
 - R is a **more modern**, functional programming language
 - **Shiny** (interactive apps) and **markdown/Quarto** (reproducible docs)
 - We want to **enable much more efficient ways of working**

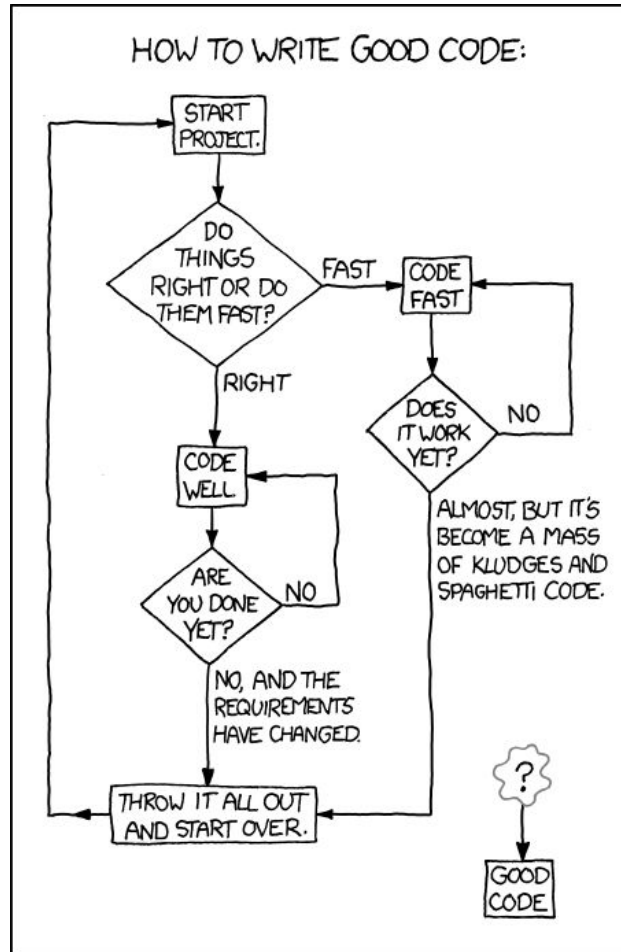


Why take Software Engineering in Biostatistics seriously?

Because the internally produced code is critical to the business

- **The code we write is critical to the business**
 - Most of fit-for-purpose project code is written internally (in contrast to external vendor e.g.)
 - This is increasing with the growing use of R replacing proprietary software.
- **Analysis code used in multiple projects needs to be scaled** towards being:
 - reliable
 - extensible
 - adaptable
 - reproducible
 - intuitive
 - scalable
 - maintainable
- Pharma industry is well regulated, hence **software might need to be validated on the target systems**
 - Details here out of scope...
 - ...but the **code needs to be well developed, tested and documented.**
- This work **takes experience and time.**

Recipe?



How Research Software Engineering (RSE) can help

RSE is a well known field in academia, less so far in the industry

- **Apply best coding practices** (transferred from open source/tech/computer science):
 - version control
 - code review
 - unit testing (micro level)
 - integration tests (macro level)
 - continuous integration
 - reproducibility
 - etc.
- **Dedicated RSE teams can ...**
 - cultivate these best coding practices as a “center of excellence”
 - develop the cross-project backbone code and avoid time-intensive copy/pasting/adapting code
 - work together with other companies on open source packages
- This talk:
 - **Share our experience in building such an RSE in Biostatistics team at Roche**



Statistical Engineering team

Mission: Accelerate Innovation

The Statistical Engineering team's mission is to co-create tools accelerating the use of new statistical methods and analysis of new data types by project teams in order to achieve maximum impact through new actionable insights.

Remit

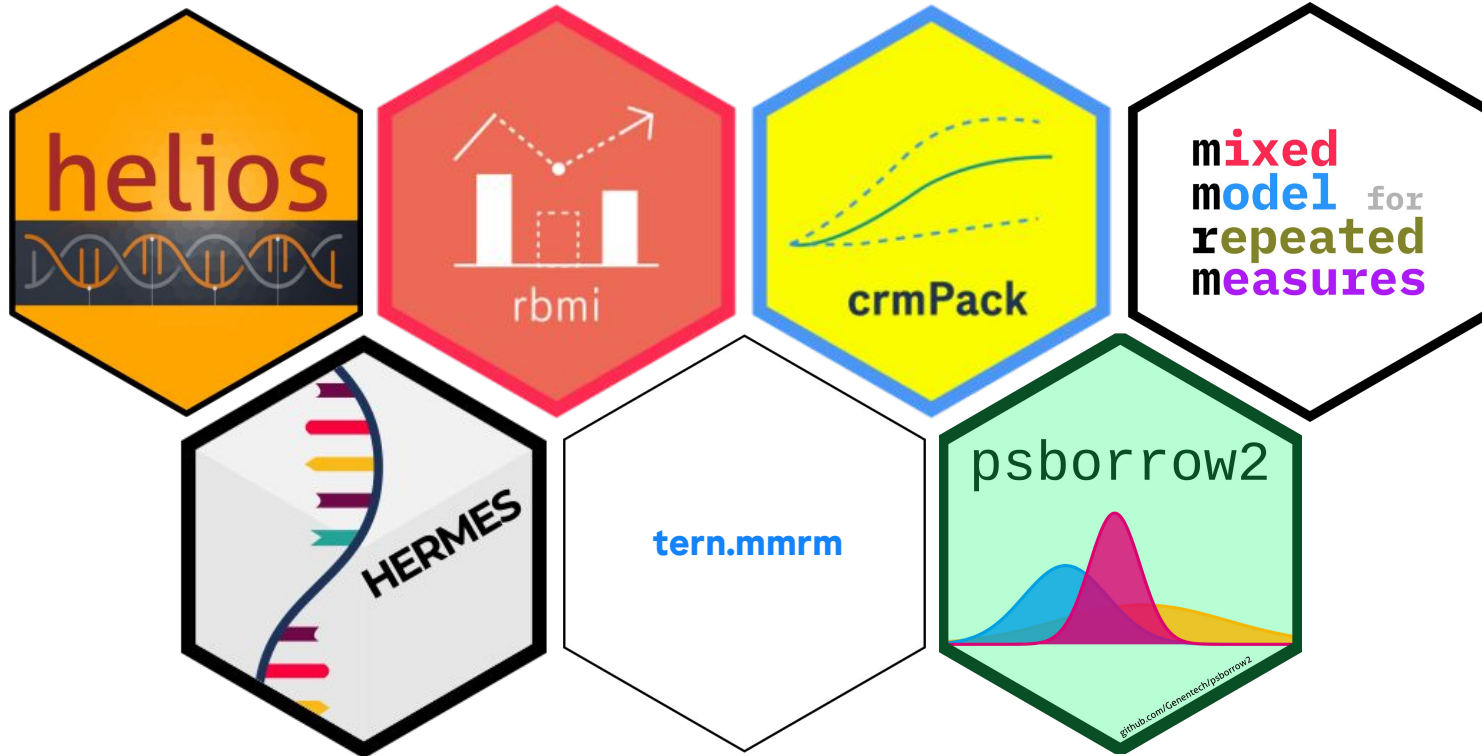
We engineer **statistical software tools**.

- Typically, these are
 - **R packages,**
 - **Shiny (teal) modules,** or
 - **how-to templates**
- Aim to build high-quality tools
 - which fulfill the
before mentioned requirements
- Users are typically from our own data science department but can come from the whole organization.
- Focus on R due to department strategy but open to other languages as needed
 - e.g. Stan or C++ inside R packages
 - e.g. exploring Julia for special tasks



Example Products: R-packages

All of these - except helios - are open source



Meet our Team

Roughly 6 FTE total from 14 members



Benoit Falquet



Craig Gower-Page



Daniel Sabanes Bove



John Kirkpatrick



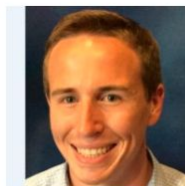
Xiaoyan Fang



Isaac Gravestock



Matt Secrest



Alex Richardson



Wojciech Wójciak



Liming Li



Stella Banjo



Guanya Peng



Philippe Boileau



Doug Kelkhoff

(as of 11 December 2022)

Resourcing strategy: Rotations

Rotations

- Currently the **majority** of the team are “rotations” from other groups (limited time, limited %)
- **Mix of levels** can be managed well through agile team organization
- Also rotations beyond our department, and across the globe, adding to the **diversity** of our team
- Also supervise internships and Master students on interesting projects

Advantages

- **Connection to the business** via incoming rotations
- Build on and **increasing internal expertise**
- **Export of skill sets** to the business
- Enabling diverse **career paths**

Challenges

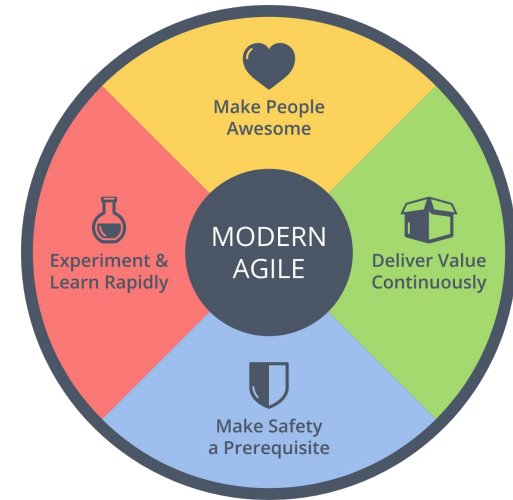
- Limitation of parallel projects / **speed** due to few long term members acting as leads
- Finding strong **candidates** (technical bar)



Agile Ways of Working

... proved to be very effective for us

- We follow the **agile principles** (<https://agilemanifesto.org/>):
 - **Individuals and interactions** over processes and tools
→ prefer chat rooms, meetings; avoid email!
 - **Working software** over comprehensive documentation
→ tests and automatically generated docs
 - **Customer collaboration** over contract negotiation
→ user input from the start to the end
 - **Responding to change** over following a plan
→ we don't have detailed long term plans, but only a strategic direction
- **Product specific teams** (e.g. for one R-package)
 - Working always on a GitHub / Gitlab **kanban board** to keep the overview
 - Meet online at least 2-3 times per week in **stand-up meetings** to remove blockers and touch base
 - **Continuous integration** via GitHub actions + thorough unit tests to ensure working software at all times, supported by separate horizontal team



Open Source collaboration

External Collaboration: Investing in Open Source

Why do we Invest in Open Source?

- **Direct Value** from Open Source:
 - Attracting new **talents**
 - Maintain high **quality bar** for our tools (as we need to pass external reviews)
 - Enables **synergistic collaboration** with other companies and academic institutions
 - **Regulators can use and trust our tools** e.g. when reviewing submissions that use them
- Indirect Value:
 - **Thought Leadership** on critical topics
 - Opportunity to define **industry standards**
 - **Publications** are attractive for authors



New Umbrella for Open Source efforts: The SWE WG

Software Engineering working group (SWE WG)

- Associated with the **American Statistical Association (ASA) Biopharmaceutical section (BIOP)**
- Connected with R consortium via ASA / Joe Rickert
- **Already 30 members** from more than 20 organizations already - and open for new members!
- Primary objective: together **engineer statistical tools** which have been missing so far
 - first product: **mrm**
- Secondary objective: develop and **promote best practices** for engineering (high-quality, open source) statistical tools



Statistical Engineering = Our Plug into the SWE WG

Practising fluent collaboration across companies and organizations

internal

external

Internal projects

Statistical Engineering



Joint new projects



MERCK



Open-source collaboration



Open-sourced projects



MSD



SWE WG as umbrella facilitating fluent collaborations

Outlook

Beyond our own organization and towards fluid collaboration beyond borders

- **Increasing importance of open source programming languages** within Pharma ...
 - is a huge opportunity ...
 - ... but needs to be handled with care = increased focus on Software Engineering
- **Working together across organizations** is key ...
 - to become more efficient
 - to obtain industry standards
 - to get the buy-in from regulators
 - to improve outcomes over closed source proprietary solutions



Thank you for your time - time for questions!



Few quick References:

- Homepage of the ASA BIOP SWE WG: rconsortium.github.io/asa-biop-swe-wg
- Recently released new R-package for MMRM: openpharma.github.io/mmrn
- Agile principles: agilemanifesto.org and modernagile.org
- RSE: researchsoftware.org

Doing now what patients need next