

m2r: Macaulay2 in



David J. Kahle
Associate Professor

Joint work with Chris O'Neill (UCD) and Jeff Sommars (UIC)



BAYLOR
UNIVERSITY

DEPARTMENT OF STATISTICAL SCIENCE

1. R and the algstat ecosystem

2. Demo

3. Internals

4. Connecting to Macaulay2

5. GitHub and contributing



and the
algstat Ecosystem

R is the lingua franca of the statistics world

R is the lingua franca of the statistics world

- rich collection of base object class systems (S3, S4, R6, ...)

R is the lingua franca of the statistics world

- rich collection of base object class systems (S3, S4, R6, ...)
- variety of ways to create new objects

R is the lingua franca of the statistics world

- rich collection of base object class systems (S3, S4, R6, ...)
- variety of ways to create new objects
- flexible mechanisms for extending the language
 - (including infix operators and overloading)

R is the lingua franca of the statistics world

- rich collection of base object class systems (S3, S4, R6, ...)
- variety of ways to create new objects
- flexible mechanisms for extending the language
 - (including infix operators and overloading)
- access to the operating system (rw, sockets, ...)

R is the lingua franca of the statistics world

- rich collection of base object class systems (S3, S4, R6, ...)
- variety of ways to create new objects
- flexible mechanisms for extending the language
 - (including infix operators and overloading)
- access to the operating system (rw, sockets, ...)
- simple ways to incorporate C++ subroutines (Rcpp)

R is the lingua franca of the statistics world

- rich collection of base object class systems (S3, S4, R6, ...)
- variety of ways to create new objects
- flexible mechanisms for extending the language
 - (including infix operators and overloading)
- access to the operating system (rw, sockets, ...)
- simple ways to incorporate C++ subroutines (Rcpp)
- thousands of add-on packages

RStudio is a popular IDE that facilitates coding in R, including

RStudio is a popular IDE that facilitates coding in R, including

- standard IDE tools for coding

RStudio is a popular IDE that facilitates coding in R, including

- standard IDE tools for coding
 - syntax highlighting, popups and tab-completion, projects, debugging, etc.

RStudio is a popular IDE that facilitates coding in R, including

- standard IDE tools for coding
 - syntax highlighting, popups and tab-completion, projects, debugging, etc.
- terminal window for shell scripting

RStudio is a popular IDE that facilitates coding in R, including

- standard IDE tools for coding
 - syntax highlighting, popups and tab-completion, projects, debugging, etc.
- terminal window for shell scripting
- document prep tools

RStudio is a popular IDE that facilitates coding in R, including

- standard IDE tools for coding
 - syntax highlighting, popups and tab-completion, projects, debugging, etc.
- terminal window for shell scripting
- document prep tools
 - (*LaTeX* and *Sweave/knitr* for PDFs, *RMarkdown* for PDFs, webpages, slides, etc.)

RStudio is a popular IDE that facilitates coding in R, including

- standard IDE tools for coding
 - syntax highlighting, popups and tab-completion, projects, debugging, etc.
- terminal window for shell scripting
- document prep tools
 - (`LaTeX` and `Sweave/knitr` for PDFs, `RMarkdown` for PDFs, webpages, slides, etc.)
- package development tools

RStudio is a popular IDE that facilitates coding in R, including

- standard IDE tools for coding
 - syntax highlighting, popups and tab-completion, projects, debugging, etc.
- terminal window for shell scripting
- document prep tools
 - (`LaTeX` and `Sweave/knitr` for PDFs, `RMarkdown` for PDFs, webpages, slides, etc.)
- package development tools
- easy deployment of webpages with R server-side backends

RStudio is a popular IDE that facilitates coding in R, including

- standard IDE tools for coding
 - syntax highlighting, popups and tab-completion, projects, debugging, etc.
- terminal window for shell scripting
- document prep tools
 - (`LaTeX` and `Sweave/knitr` for PDFs, `RMarkdown` for PDFs, webpages, slides, etc.)
- package development tools
- easy deployment of webpages with R server-side backends
- collaborative coding tools

R in industry



cultureamp



JOHN DEERE

KICK
STARTER
.COM

The
New York
Times



mozilla
Firefox



NORDSTROM



NIST



twitter

R is the lingua franca of the statistics world

R is the lingua franca of the statistics world

R is the natural platform for *applied algebraic statistics*

R is the lingua franca of the statistics world

R is the natural platform for *applied* algebraic statistics

Problem: R has no support for symbolic computing

R is the lingua franca of the statistics world

R is the natural platform for *applied* algebraic statistics

Problem: R has no support for symbolic computing

Solution: outsource computations

2011 : mpoly – data structures and methods for multivariate polynomials

2011 : mpoly – data structures and methods for multivariate polynomials

mpoly

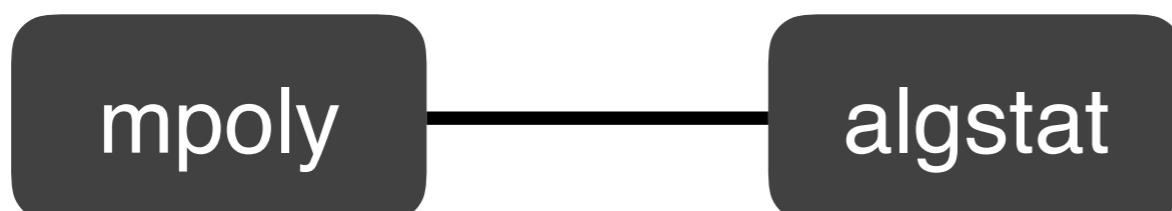
2011 : **mpoly** – data structures and methods for multivariate polynomials

2014 : **algstat** – algebraic statistical data analysis

mpoly

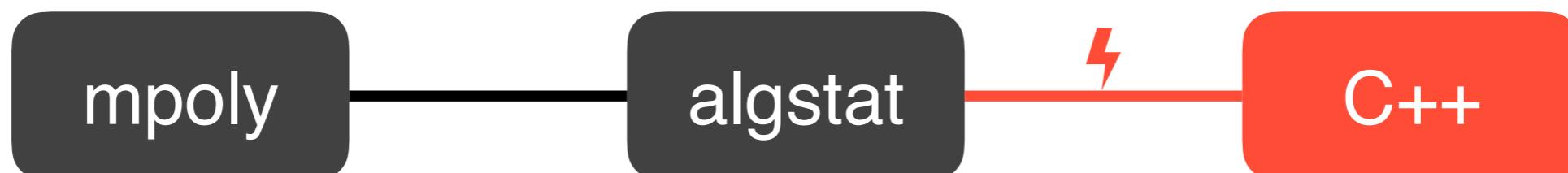
2011 : **mpoly** – data structures and methods for multivariate polynomials

2014 : **algstat** – algebraic statistical data analysis



2011 : `mpoly` – data structures and methods for multivariate polynomials

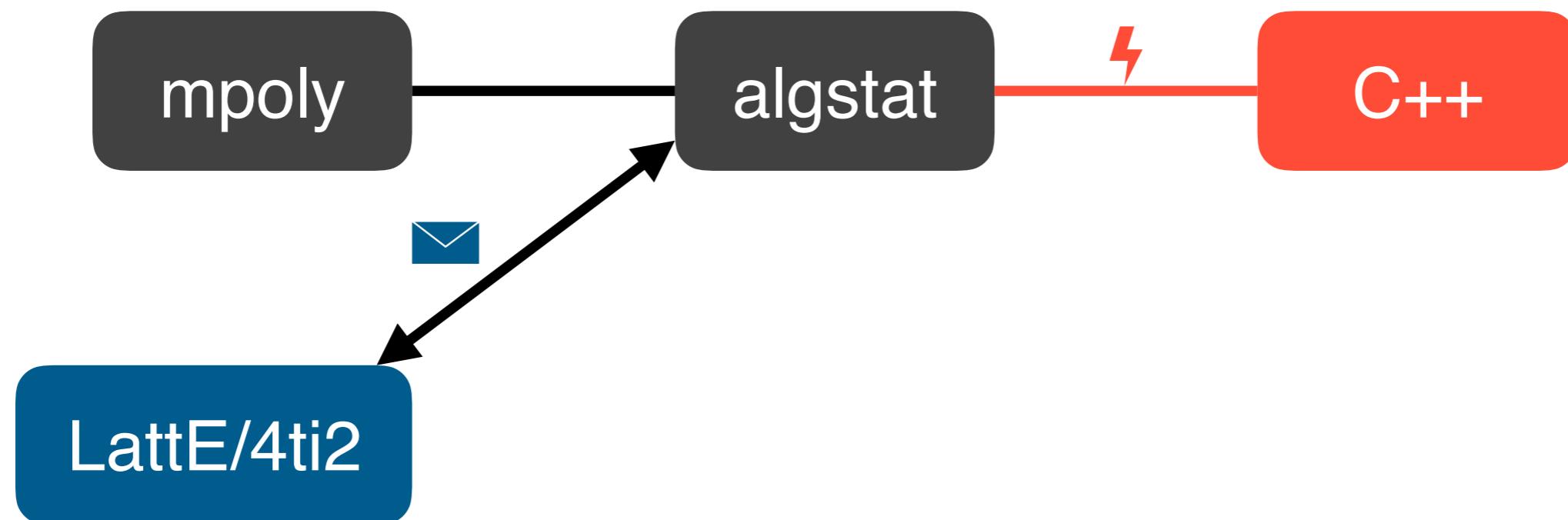
2014 : `algstat` – algebraic statistical data analysis



Objects computed on in place

2011 : `mpoly` – data structures and methods for multivariate polynomials

2014 : `algstat` – algebraic statistical data analysis



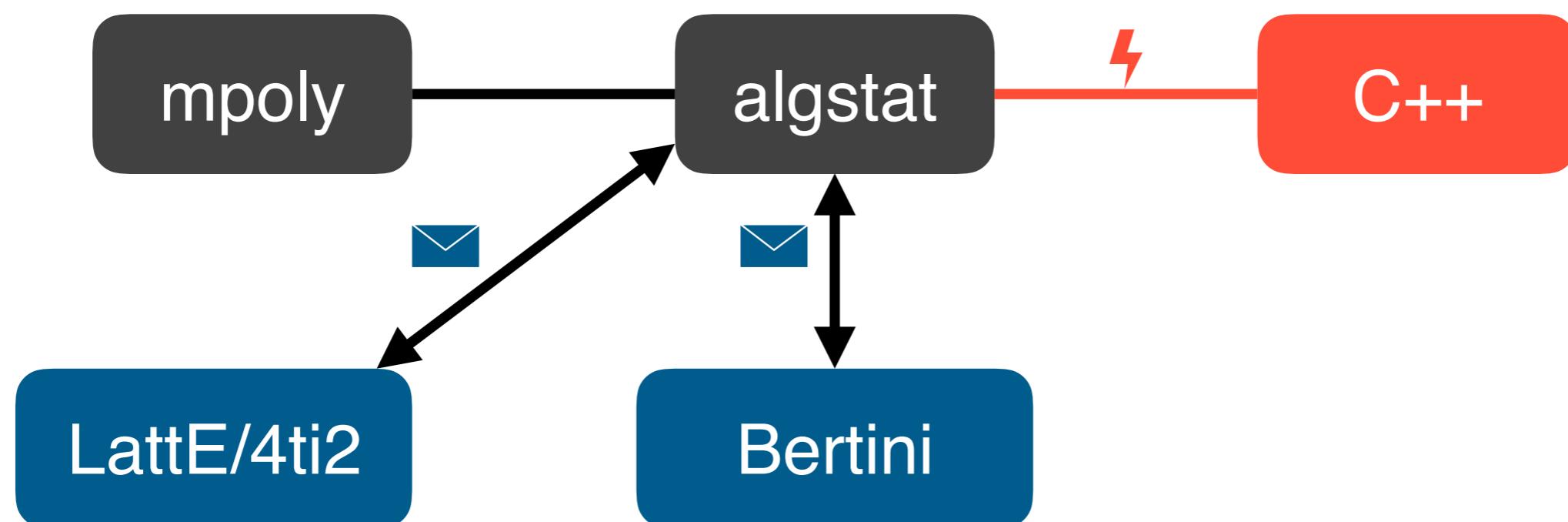
Objects computed on in place



R writes / program executes / R reads

2011 : `mpoly` – data structures and methods for multivariate polynomials

2014 : `algstat` – algebraic statistical data analysis



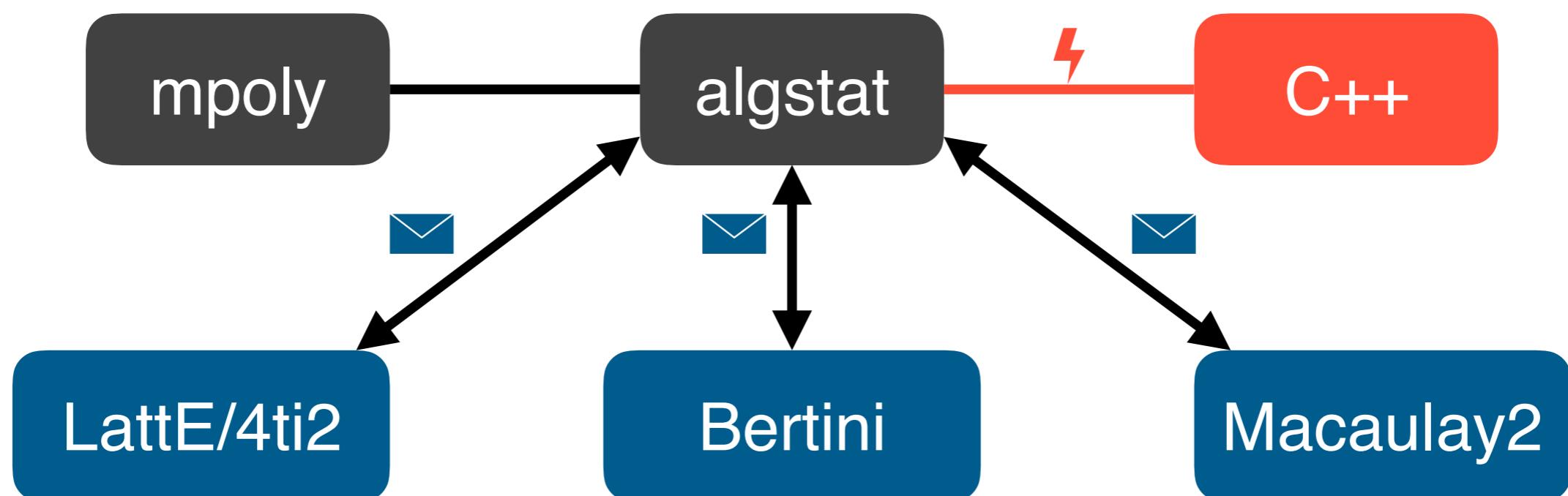
Objects computed on in place



R writes / program executes / R reads

2011 : mpoly – data structures and methods for multivariate polynomials

2014 : algstat – algebraic statistical data analysis



Objects computed on in place



R writes / program executes / R reads

2011 : `mpoly` – data structures and methods for multivariate polynomials

2014 : `algstat` – algebraic statistical data analysis

Then, connections needed their own packages!

2011 : **mpoly** – data structures and methods for multivariate polynomials

2014 : **algstat** – algebraic statistical data analysis

Then, connections needed their own packages!

2015 : **latter** – LattE/4ti2

2011 : **mpoly** – data structures and methods for multivariate polynomials

2014 : **algstat** – algebraic statistical data analysis

Then, connections needed their own packages!

2015 : **latter** – LattE/4ti2

2016 : **m2r** – Macaulay2

2011 : **mpoly** – data structures and methods for multivariate polynomials

2014 : **algstat** – algebraic statistical data analysis

Then, connections needed their own packages!

2015 : **latter** – LattE/4ti2

2016 : **m2r** – Macaulay2

2017 : **tropical** – Tropical geometry

2011 : **mpoly** – data structures and methods for multivariate polynomials

2014 : **algstat** – algebraic statistical data analysis

Then, connections needed their own packages!

2015 : **latter** – LattE/4ti2

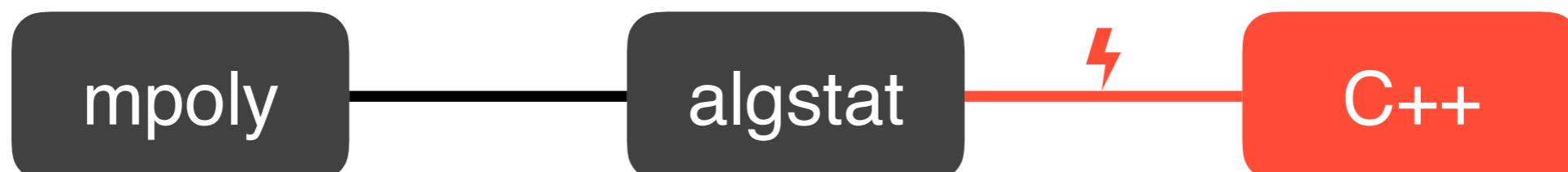
2016 : **m2r** – Macaulay2

2017 : **tropical** – Tropical geometry

2018 : **bertini** – Bertini

2011 : **mpoly** – data structures and methods for multivariate polynomials

2014 : **algstat** – algebraic statistical data analysis



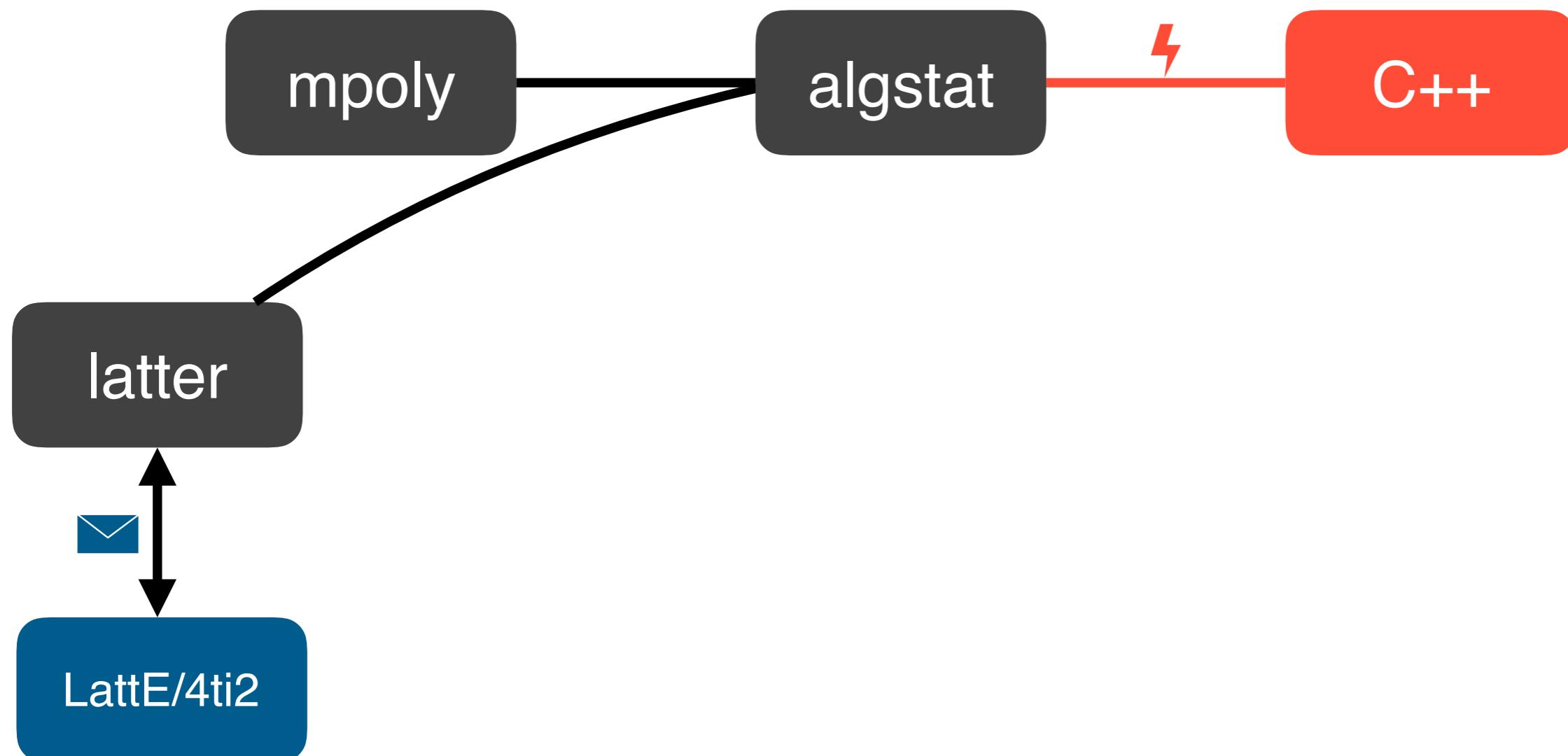
2011 : `mpoly` – data structures and methods for multivariate polynomials

2014 : `algstat` – algebraic statistical data analysis



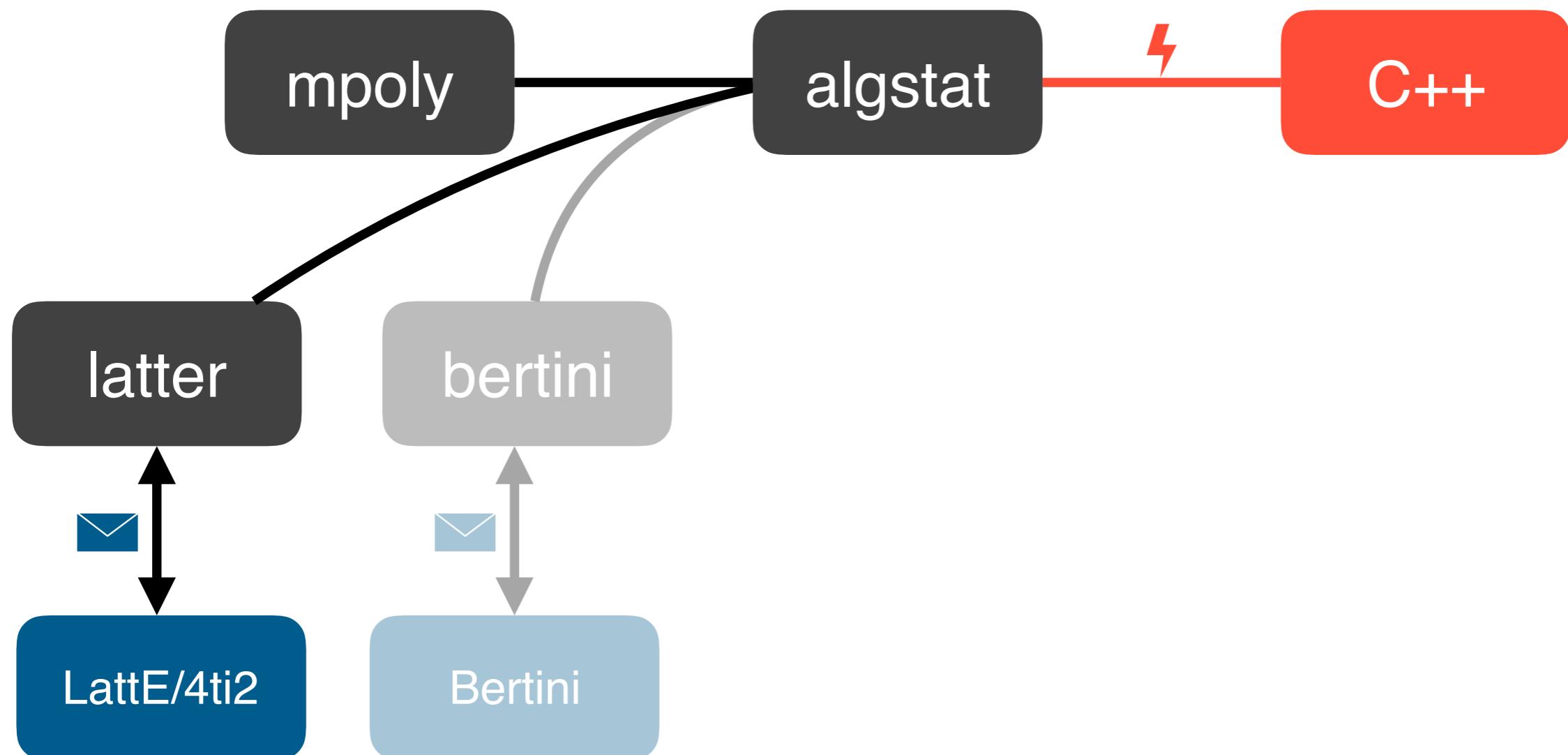
2011 : `mpoly` – data structures and methods for multivariate polynomials

2014 : `algstat` – algebraic statistical data analysis



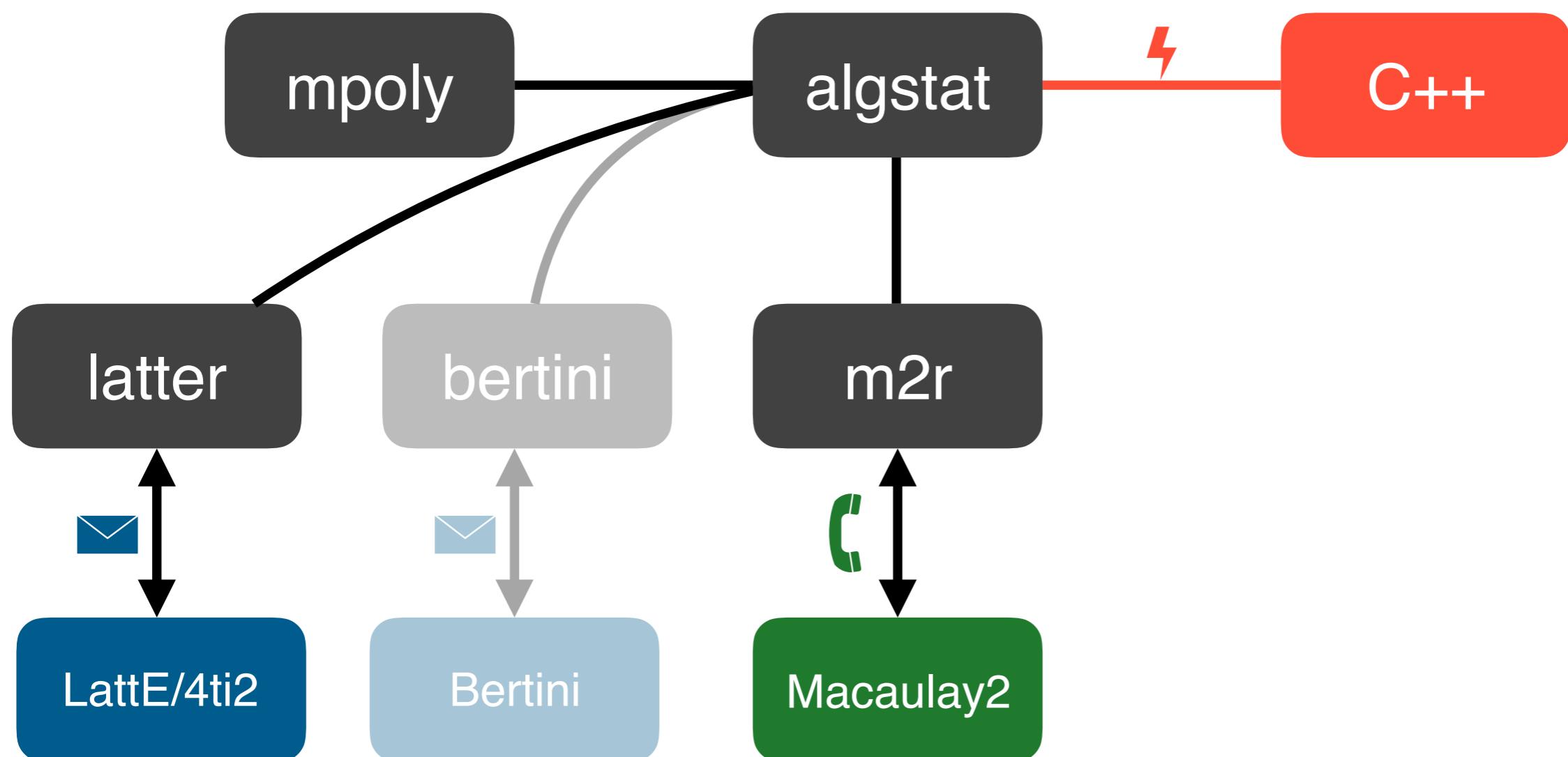
2011 : `mpoly` – data structures and methods for multivariate polynomials

2014 : `algstat` – algebraic statistical data analysis



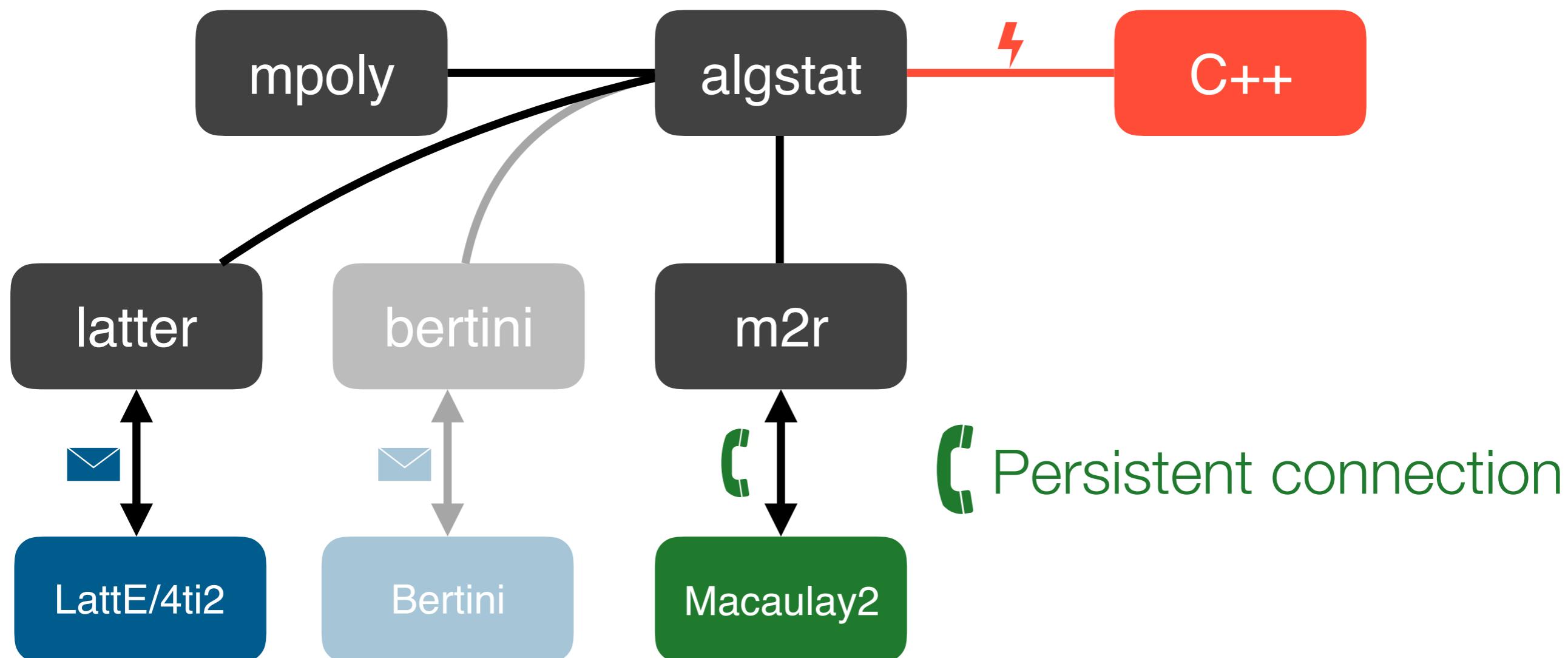
2011 : mpoly – data structures and methods for multivariate polynomials

2014 : algstat – algebraic statistical data analysis



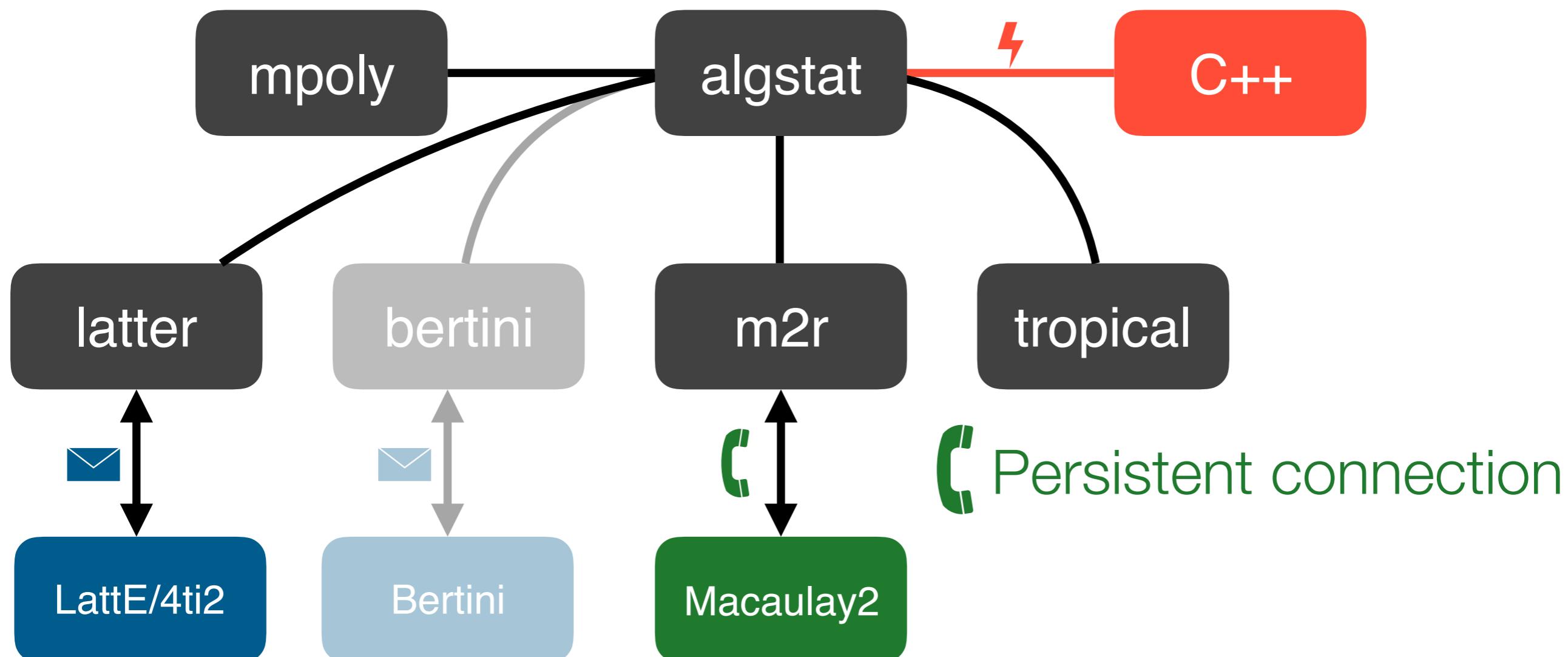
2011 : mpoly – data structures and methods for multivariate polynomials

2014 : algstat – algebraic statistical data analysis



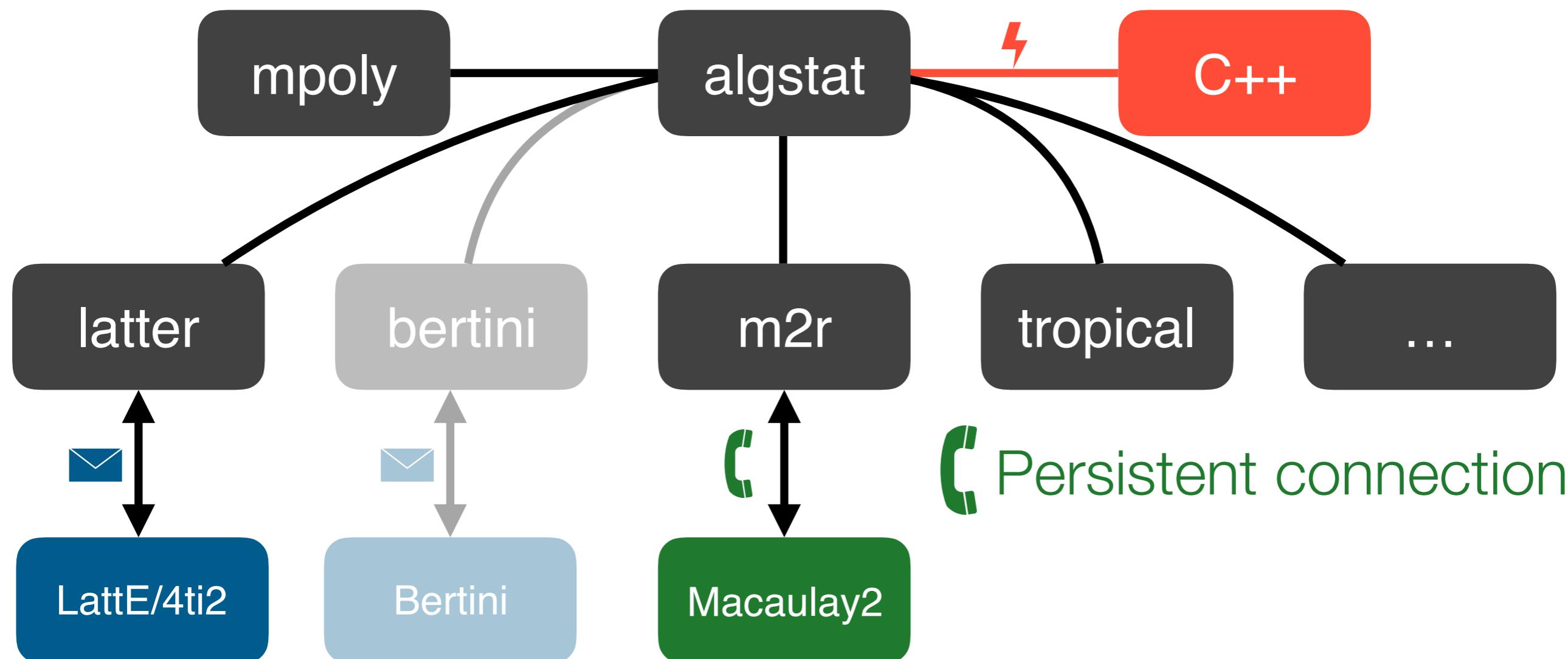
2011 : mpoly – data structures and methods for multivariate polynomials

2014 : algstat – algebraic statistical data analysis



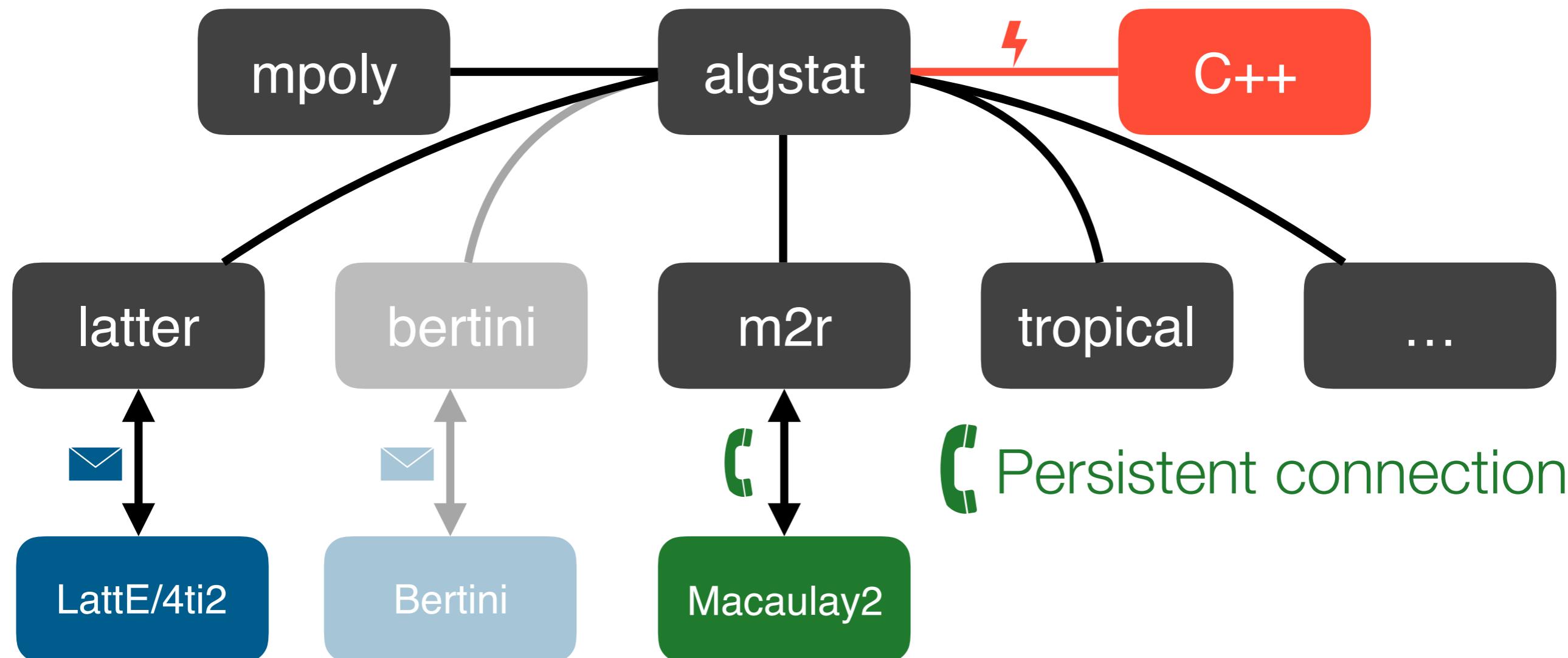
2011 : mpoly – data structures and methods for multivariate polynomials

2014 : algstat – algebraic statistical data analysis



2011 : mpoly – data structures and methods for multivariate polynomials

2014 : algstat – algebraic statistical data analysis



* Many other interconnections not shown

demo

Installing the algstat ecosystem



To install the packages in this talk, copy/paste the following code into R

```
if(!requireNamespace("devtools"))
  install.packages("devtools")

library(devtools)
install_github("dkahle/mpoly", ref = "670645f")
install_github("coneill-math/m2r", ref = "3cf5e94d")
```

System of polynomial equations

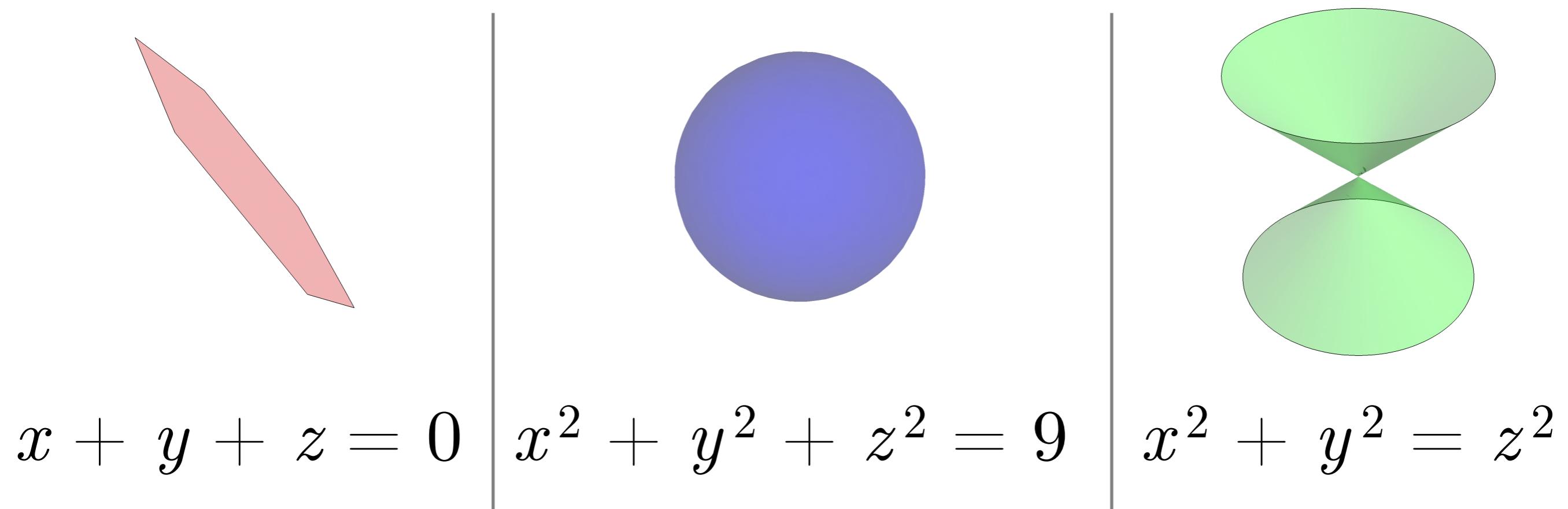
Solve the following system :

$$x + y + z = 0$$

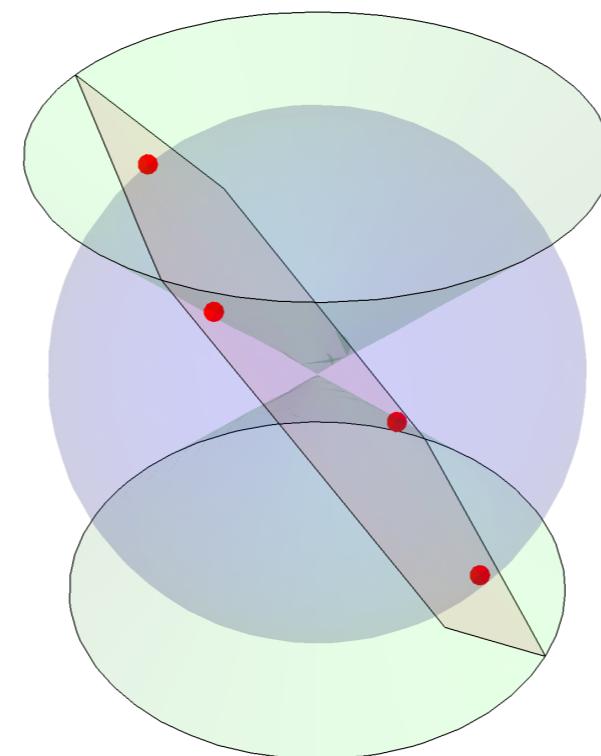
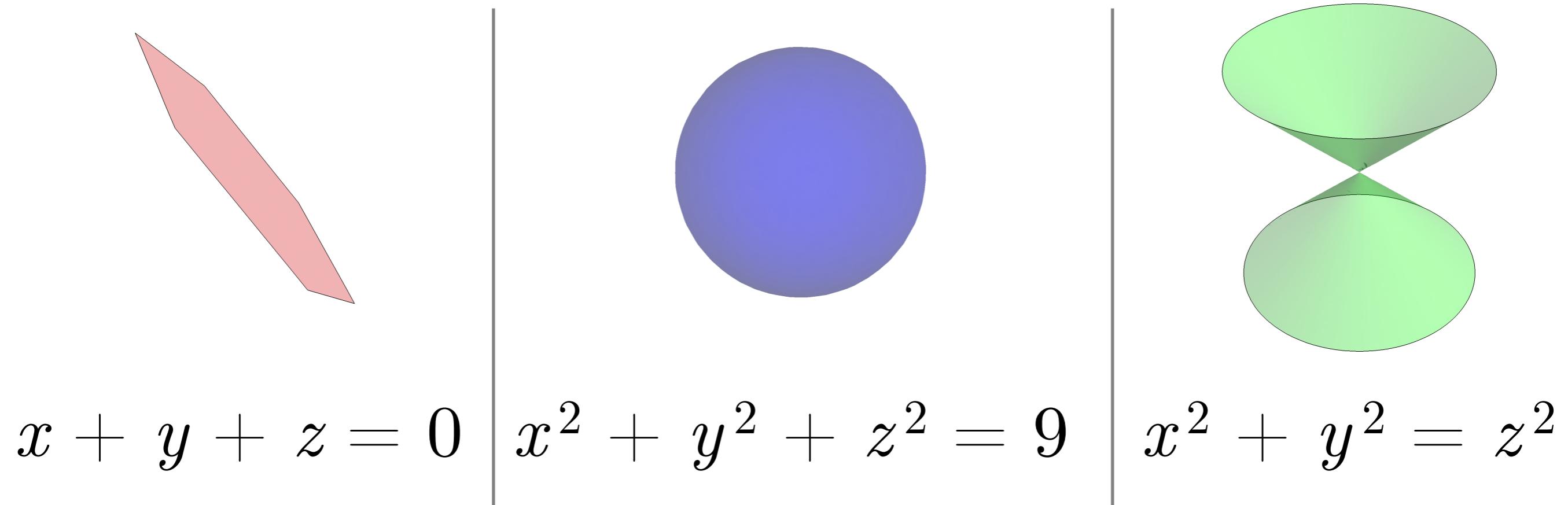
$$x^2 + y^2 + z^2 = 9$$

$$x^2 + y^2 = z^2$$

System of polynomial equations



System of polynomial equations



System of polynomial equations

Grobner bases result in the system

$$x + y + z = 0$$

$$x^2 + y^2 + z^2 = 9$$

$$x^2 + y^2 = z^2$$

System of polynomial equations

Grobner bases result in the system

$$x + y + z = 0$$

$$x^2 + y^2 + z^2 = 9$$

$$x^2 + y^2 = z^2$$



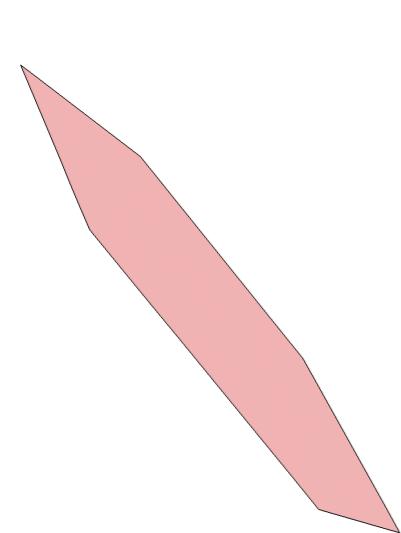
$$x + y + z = 0$$

$$y^2 + yz = 0$$

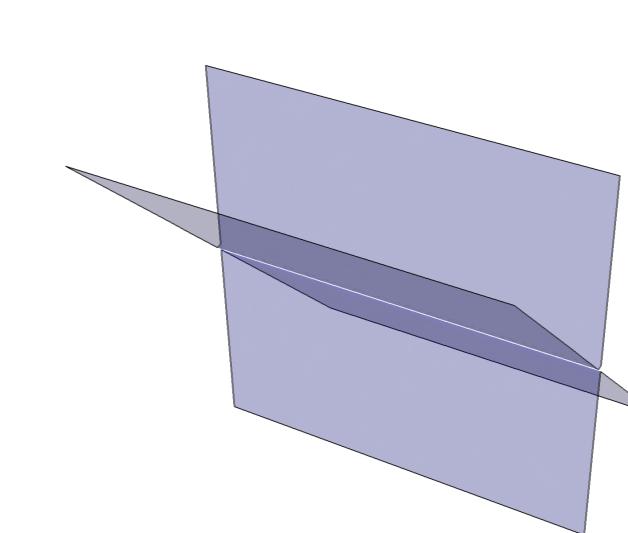
$$2z^2 - 9 = 0$$

System of polynomial equations

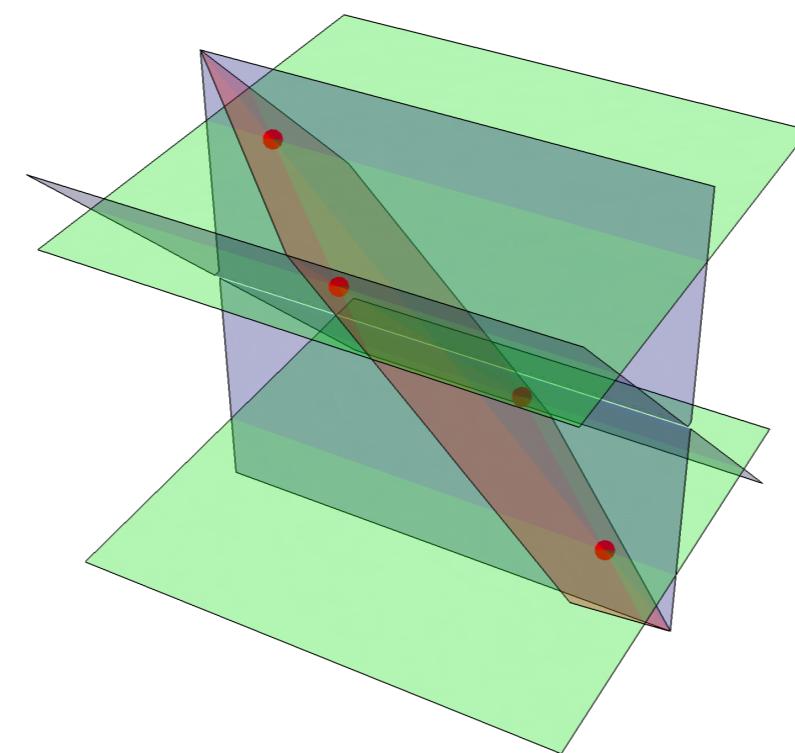
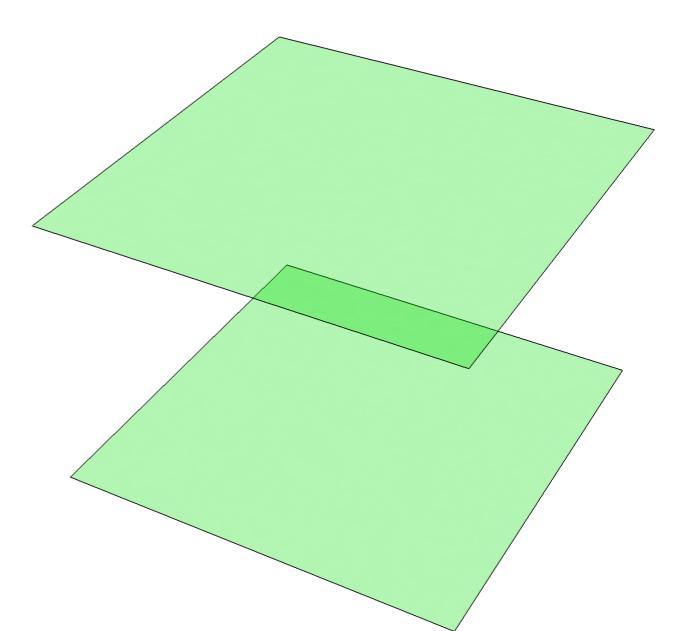
$$x + y + z = 0$$



$$y^2 + yz = 0$$

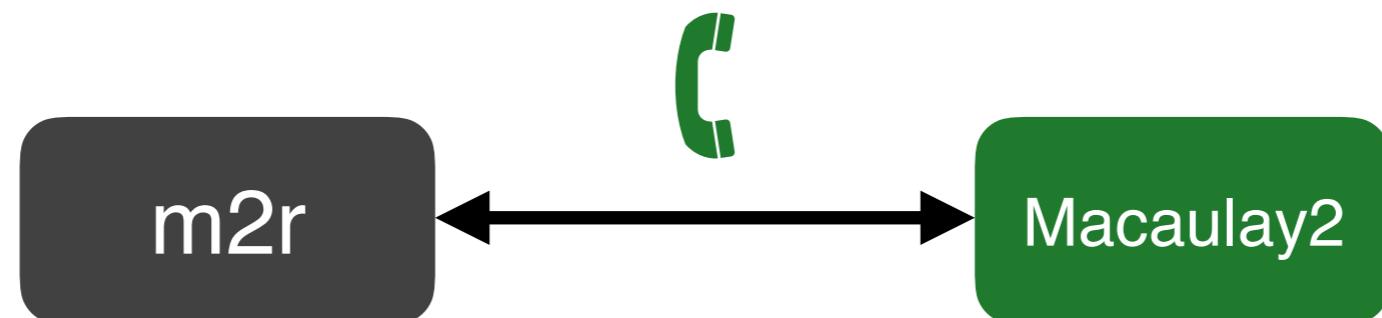


$$2z^2 - 9 = 0$$

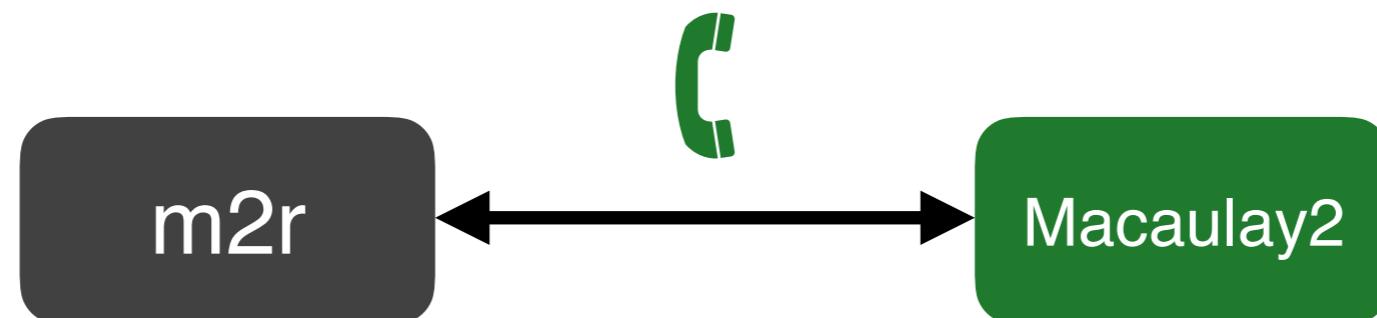


internals

Communication between R and Macaulay2 requires several steps



Communication between R and Macaulay2 requires several steps



I won't talk about initializing the connection between R and Macaulay2

Communication between R and Macaulay2 requires several steps

m2r

Macaulay2

Communication between R and Macaulay2 requires several steps

m2r

Macaulay2

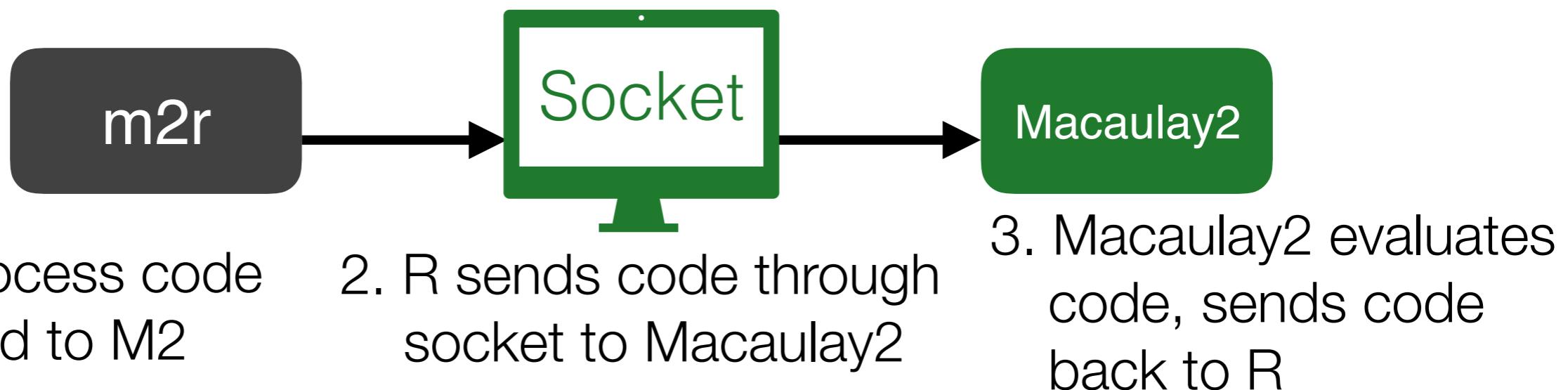
1. Preprocess code to send to M2

Communication between R and Macaulay2 requires several steps

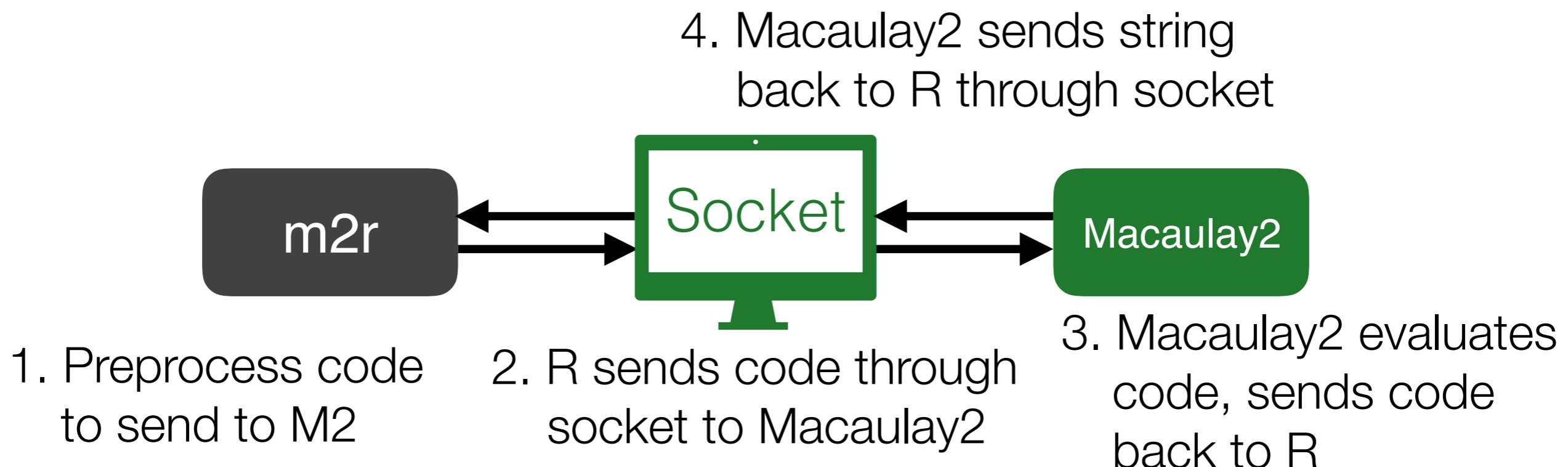


1. Preprocess code to send to M2
2. R sends code through socket to Macaulay2

Communication between R and Macaulay2 requires several steps



Communication between R and Macaulay2 requires several steps



Communication between R and Macaulay2 requires several steps

5. R parses Macaulay2 output into R structures

4. Macaulay2 sends string back to R through socket

1. Preprocess code to send to M2

2. R sends code through socket to Macaulay2

3. Macaulay2 evaluates code, sends code back to R



Communication between R and Macaulay2 requires several steps

This part is expensive

5. R parses Macaulay2 output into R structures

4. Macaulay2 sends string back to R through socket

1. Preprocess code to send to M2

2. R sends code through socket to Macaulay2

3. Macaulay2 evaluates code, sends code back to R



Communication between R and Macaulay2 requires several steps

This part is expensive

5. R parses Macaulay2 output into R structures

4. Macaulay2 sends string back to R through socket



1. Preprocess code to send to M2

2. R sends code through socket to Macaulay2

3. Macaulay2 evaluates code, sends code back to R

For large computations, we prefer to leave objects in Macaulay2 and only keep a “handle” on the R side

Most m2r functions have reference versions that return pointers, skipping the expensive parsing step 5.

Demo

connecting to Macaulay2

Instead of running Macaulay2 locally, m2r can spin instances of Macaulay2 in the cloud

Instead of running Macaulay2 locally, m2r can spin instances of Macaulay2 in the cloud



Instead of running Macaulay2 locally, m2r can spin instances of Macaulay2 in the cloud



This uses Amazon Web Services (AWS) Elastic Compute Cloud (EC2) instances

Instead of running Macaulay2 locally, m2r can spin instances of Macaulay2 in the cloud



This uses Amazon Web Services (AWS) Elastic Compute Cloud (EC2) instances

This is done automatically on machines where Macaulay2 is not found

GitHub and contributing

To submit a feature request or report a bug:

- Go to <https://github.com/> and create a free account
- Go to <https://github.com/coneill-math/m2r>
- Click *Issues*
- Click *New Issue*

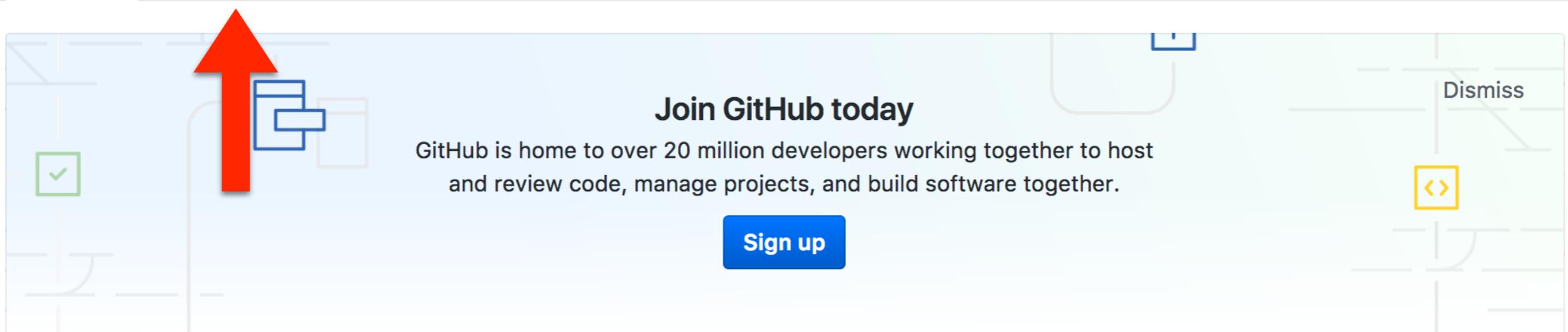
GitHub - coneill-math/m2r: Ma ×

GitHub, Inc. [US] | https://github.com/coneill-math/m2r

Features Business Explore Marketplace Pricing This repository Search Sign in or Sign up

coneill-math / m2r Watch 4 Star 0 Fork 4

Code Issues 19 Pull requests 0 Projects 0 Insights ▾



Macaulay2 interface for R

464 commits 1 branch 0 releases 4 contributors

Branch: master New pull request Find file Clone or download

Christopher O'Neill Ignore .DS_Store files Latest commit 3cf5e94 4 days ago

R fix tests for EC2 testing 3 months ago

inst/server update m2 server to use wait 28 days ago

man fix tests for EC2 testing 3 months ago



Features

Business

Explore

Marketplace

Pricing

This repository

Search

Sign in or Sign up

coneill-math / m2r

Watch

4

Star

0

Fork

4

Code

Issues 19

Pull requests 0

Projects 0

Insights ▾

is:issue is:open

Labels

Milestones

New issue

19 Open ✓ 55 Closed

Author ▾

Labels ▾

Projects ▾

Milestones ▾

Assignee ▾



Cloud session resume feature request

#197 opened 25 days ago by coneill-math

Support naming rings and ideals on creation bug

#196 opened on Jul 31 by coneill-math

Add support for maps feature request

#195 opened on Jul 31 by coneill-math



Vectorize Macaulay2 functions



#177 opened on Jun 7 by dkahle

1

Implement arguments to key functions enhancement



#157 opened on May 30 by dkahle

1

Fix numRows and numCols functions to use ncol and nrow bug



#156 opened on May 30 by coneill-math

1

Code print option increments ring counter unnecessarily

#153 opened on May 30 by coneill-math



1

Parse fractions properly and implement GMP bug



14

To join the fray, submit a pull request (PR)!

- Go to <https://github.com/> and create a free account
- Go to <https://github.com/coneill-math/m2r>
- Click *Fork* to make your own copy of the repository
- In RStudio...
 - File > New Project... > Version Control > Git
 - Enter the GitHub URL, <https://github.com/dkahle/algstat.git>
 - Add/change code and commit, see [tutorial here](#)
 - Push changes to GitHub
- On GitHub, click *Submit a Pull Request*

Thank you!!

www.kahle.io

<https://github.com/dkahle/2017-AMS-Sectional-Talk>

This material is based upon work supported by the National Science Foundation under Grant Nos. [1321794](#), [1622449](#), and [1622369](#).