ANSIBLE

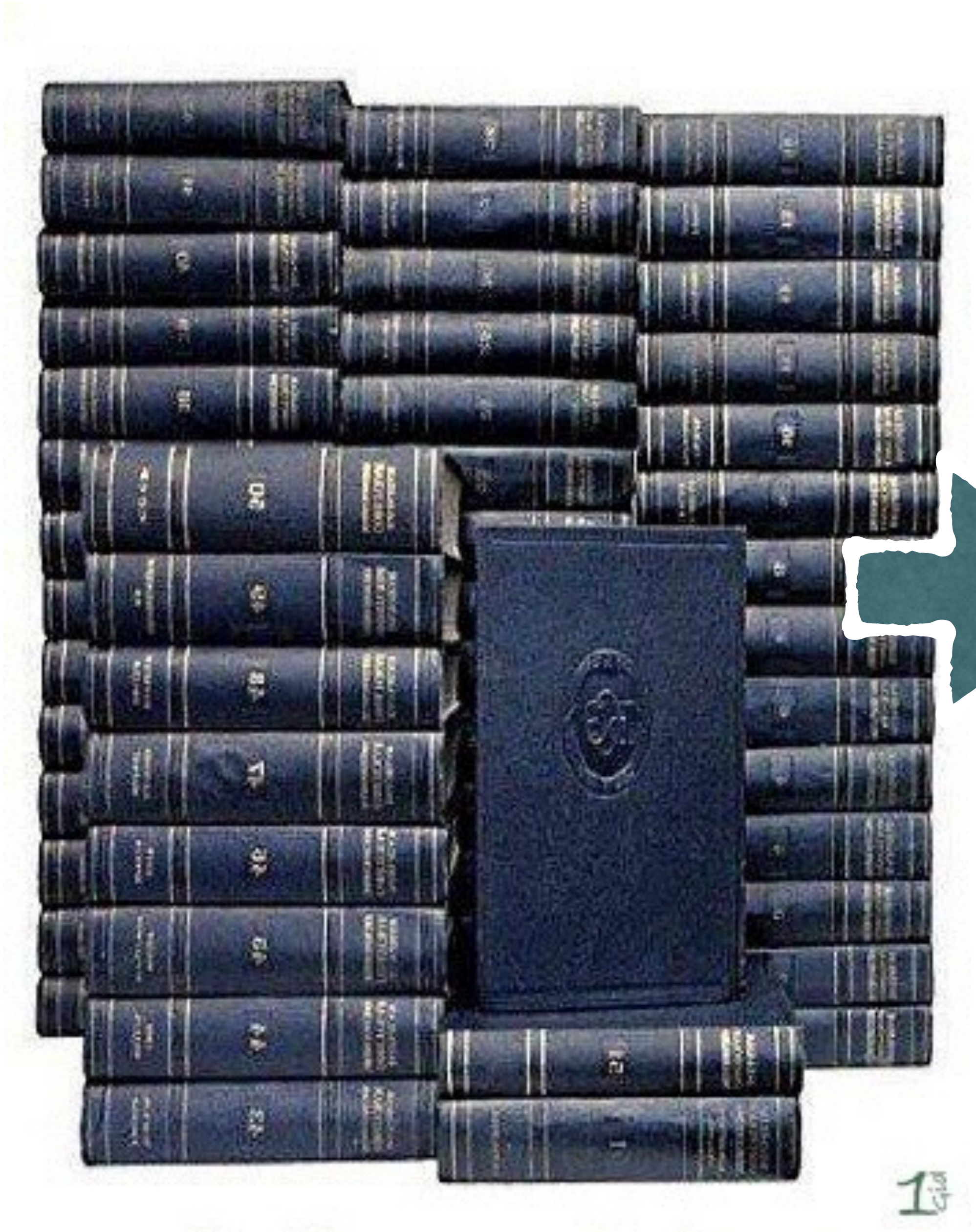
WHO AM I?

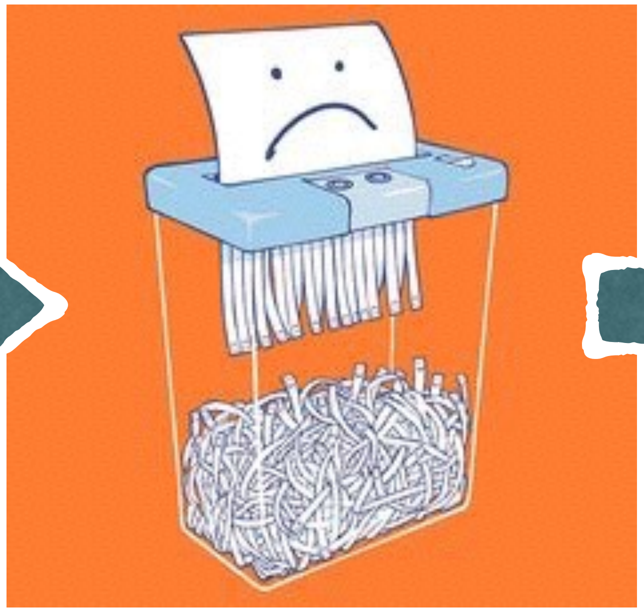
- Tetiana Khotiaintseva `@quit_ka`
- I am a python developer at Ardigen S.A.
- I create software for bioinformatics.

WHAT IS BIOINFORMATICS?

Bioinformatics develops methods and software tools for understanding biological data.

WHAT'S GENOME SEQUENCING?





WHAT IS BIOINFORMATICS?

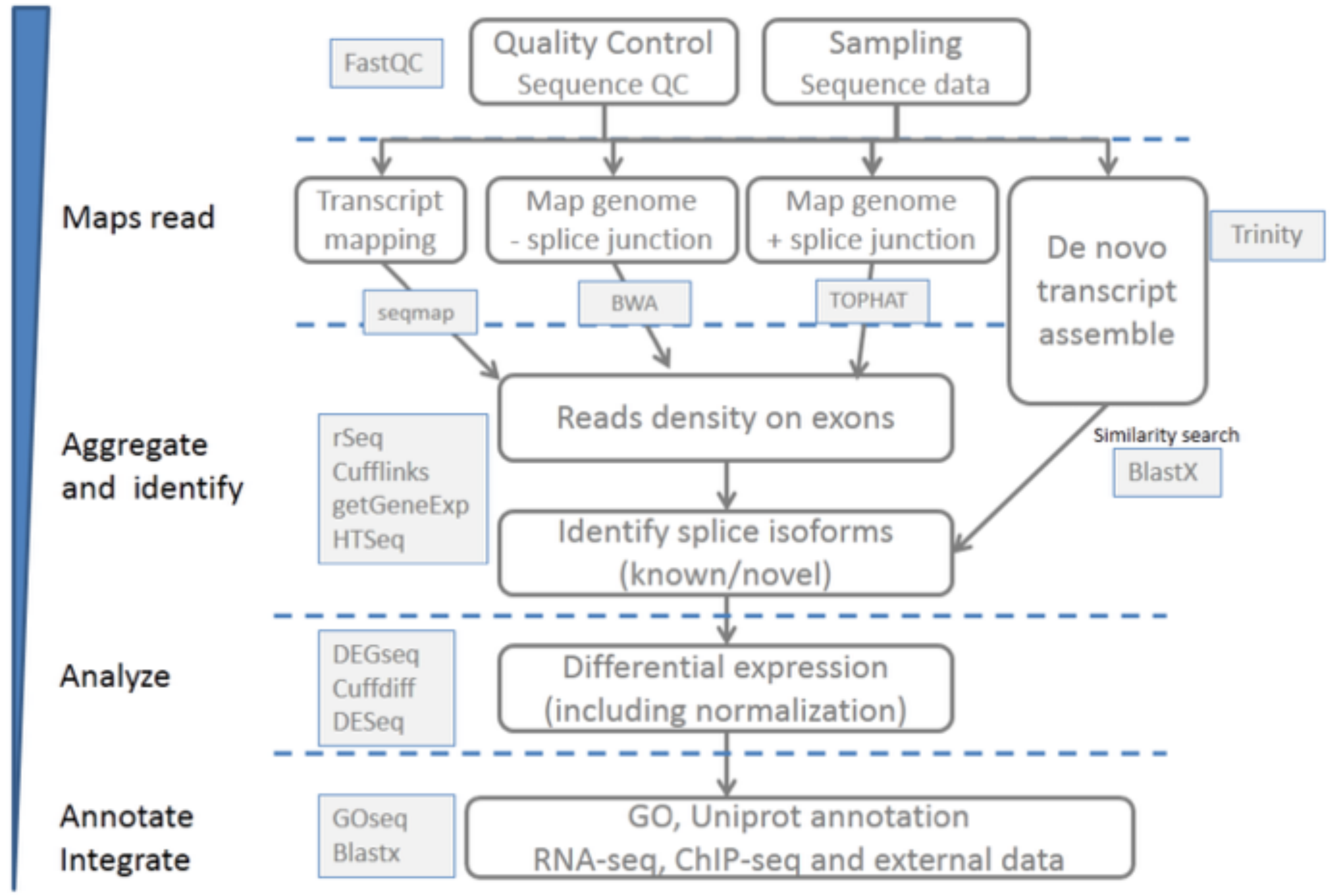
Bioinformatics is coming out of academia into the commercial world, bringing personalised and preventive medicine.

HOW IS BIOINFORMATICS SOFTWARE DIFFERENT?

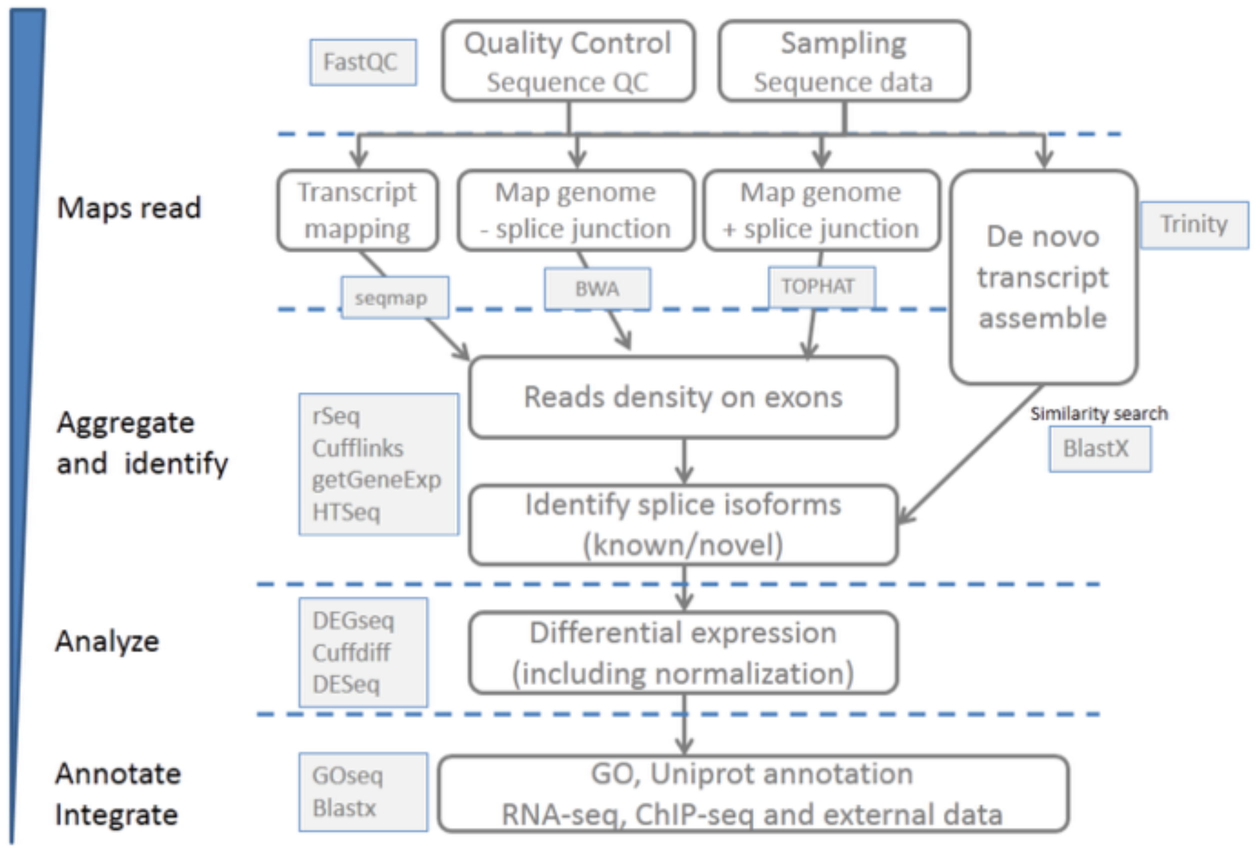
- A lot of data (single sample ~100 Gb)
- Complex, specialised algorithms
- New tools are being developed rapidly
- Developed and used by researchers
- Developed with HPC setup in mind

HPC	CLOUD
➤ Bare metal	➤ Virtualisation
➤ Specialised hardware	➤ Commodity hardware
➤ Fine-tuned software	➤ Containers
➤ InfiniBand	➤ Internet
➤ MPI	➤ REST APIs, AMQP, …
➤ Centralised data storage	➤ Data locality

DATA ANALYSIS PIPELINES


FastQC
Quality Control Sequence QC
Sampling Sequence data
Maps read
Transcript mapping
Map genome - splice junction
Map genome + splice junction
De novo transcript assemble
Trinity
seqmap
BWA
TOPHAT
Aggregate and identify
rSeq Cufflinks getGeneExp HTSeq
Reads density on exons
Similarity search
BlastX
Identify splice isoforms (known/novel)
Analyze
DEGseq Cuffdiff DESeq
Differential expression (including normalization)
Annotate Integrate
GOseq Blastx
GO, Uniprot annotation RNA-seq, ChIP-seq and external data

DATA ANALYSIS PIPELINES



- Dozens of tools in a pipeline
- Different software stacks
- Often not packaged
- Non-standard, arcane, or just-plain-weird build processes
- No ops

DATA ANALYSIS PIPELINES

- Dozens of tools in a pipeline
- Different software stacks
- Often not packaged
- Non-standard, arcane, or just-plain-weird build processes
- No ops

DATA ANALYSIS PIPELINES

- Dozens of tools in a pipeline
- Different software stacks
- Often not packaged
- Non-standard, arcane, or just-plain-weird build processes
- No ops
- **Irreproducible results**

REPRODUCIBILITY IS A CORNERSTONE OF SCIENCE

“Science moves forward by corroboration – when researchers verify others’ results. Science advances faster when people waste less time pursuing false leads. No research paper can ever be considered to be the final word … ***There is growing alarm about results that cannot be reproduced.”***

[Source http://www.nature.com/news/reproducibility-1.17552]

CONFIGURATION MANAGEMENT FTW!

#!/bin/bash

Perl

ANSIBLE

ANSIBLE OVERVIEW

- Provisioning, configuration management, application deployment
- Agentless
- Communication via ssh
- Task execution in parallel
- YAML syntax
- bash++

INVENTORY FILE

```
/etc/ansible/hosts

[app]
192.168.60.4
192.168.60.5

[db]
192.168.60.6
```

AD-HOC COMMANDS

```
$ ansible all -a “date”
192.168.60.5 | SUCCESS | rc=0 >>
Sun Nov 27 00:25:10 UTC 2016

192.168.60.4 | SUCCESS | rc=0 >>
Sun Nov 27 00:25:10 UTC 2016

192.168.60.6 | SUCCESS | rc=0 >>
Sun Nov 27 00:25:10 UTC 2016

$ ansible db -m ping
192.168.60.6 | SUCCESS | rc=0 >>
Sun Nov 27 00:28:55 UTC 2016
```

MODULES

- Wrappers around common operations
- Provide idempotence
- Declarative style

```
$ ansible all –b –m yum –a “name=git state=present”
192.168.60.5 | SUCCESS => {
    "changed": true,
    …
$ ansible all –b –m yum –a “name=git state=present”
192.168.60.5 | SUCCESS => {
    "changed": false,
    …
```

PLAYBOOKS

```yaml
- hosts: app
  become: true
  tasks:
    - name: Update apt
      apt: update_cache=yes

    - name: Install Apache
      apt: name=apache2 state=latest

    - name: Create custom document root
      file:
        path: /var/www/example
        state: directory
        owner: www-data
        group: www-data
```

CONDITIONALS

```yaml
- hosts: all
  gather_facts: yes
  remote_user: craun
  serial: "50%"
  become: yes
  tasks:
    - name: Update Shellshock (Debian)
      apt: name=bash
           state=latest
           update_cache=yes
      when: ansible_os_family == "Debian"

    - name: Update Shellshock (RedHat)
      yum: name=bash
           state=latest
           update_cache=yes
      when: ansible_os_family == "RedHat"
```

Source: https://blog.serverdensity.com/what-ive-learnt-from-using-ansible-exclusively-for-2-years/

HANDLERS

```
    - name: Set up Apache virtual host file
      template: src=vhost.tpl dest=/etc/apache2/sites-available/
000-default.conf
      notify: restart apache

  handlers:
    - name: restart apache
      service: name=apache2 state=restarted
```

- Handler is a task that can be triggered by another task
- Run once, at the end of a play
- Won’t be run if a play was stopped due to an error

VARIABLES

```
- hosts: all
  become: true
  tasks:
    - name: Install Apache, MySQL, and other dependencies.
      yum: name="{{ item }}" state=present
      with_items:
        - apache2
        - python-mysqldb
        - mysql-server
```

VARIABLES

```yaml
- hosts: all
  become: true
  vars:
    apache_depts:
      - apache2
      - python-mysqldb
      - mysql-server
  tasks:
    - name: Install Apache, MySQL, and other dependencies.
      yum: name="{{ item }}" state=present
      with_items: apache_depts
```

VARIABLES

```
- hosts: all
  become: true
  vars_files:
    - vars.yml
  tasks:
    - name: Install Apache, MySQL, and other dependencies.
      yum: name="{{ item }}" state=present
      with_items: apache_depts


./vars.yml
apache_depts:
  - apache2
  - python-mysqldb
  - mysql-server
```

INCLUDES

```yaml
- hosts: all
  pre_tasks:
    - name: Update cache if needed.
      apt: update_cache=yes cache_valid_time=3600

  handlers:
    - include: handlers.yml

  vars_files:
    - vars.yml

  tasks:
    - include: common.yml
    - include: load_balancers.yml
    - include: webservers.yml
    - include: dbservers.yml
```

PASS VARIABLES TO PLAYBOOKS

```
tasks:
  - include: user.yml
    vars:
        username: timmy
        ssh_keys:
          - { src: path/to/timmy/key1, dest: id_rsa }
          - { src: path/to/timmy/key2, dest: id_rsa_2 }
  - include: user.yml
    vars:
        username: jane
        ssh_keys:
          - { src: path/to/jane/key, dest: id_rsa }
```

RUN A PLAYBOOK

```
# run a playbook
 $ ansible-playbook playbook.yml
# check which changes will be made (dry run)
 $ ansible-playbook playbook.yml --check
# check which hosts will be affected
 $ ansible-playbook playbook.yml --list-hosts
# execute with a different inventory file
 $ ansible-playbook playbook.yml -i ./inventory_file
# specify number of parallel processes to use (defaults to 5)
 $ ansible-playbook playbook.yml --forks 20
# specify connection type
 $ ansible-playbook playbook.yml --connection=local
```

ROLES

- A way to **package** reusable playbooks
- To customise, provide variables to override the defaults

```
- hosts: mail_servers
  vars_files:
    - vars.yml
  roles:
    - postfix
- hosts: databases
  roles:
    - mysql
```

ANSIBLE GALAXY

Galaxy is your hub for finding, reusing and sharing the best Ansible content

Log Into Galaxy with GitHub
Use an existing account not associated with GitHub

```
# install a role
 $ ansible-galaxy install [role]

# list installed roles
 $ ansible-galaxy list

# Remove a role
 $ ansible-galaxy remove [role]

# Create a role template
 $ ansible-galaxy init
```

ROLES

```
/etc/ansible/roles/ANXS.fuse
├── .travis.yml
├── README.md
├── defaults
│   └── main.yml
├── meta
│   └── main.yml
├── tasks
│   ├── main.yml
│   ├── package.yml
│   └── source.yml
└── test.yml
```

```
# file: fuse/defaults/main.yml

fuse_install_method: "source"
fuse_version: "2.9.3"
```

```
 # file: fuse/tasks/main.yml

 - include: package.yml
   when: fuse_install_method ==
"package"
 - include: source.yml
   when: fuse_install_method ==
"source"
```

```
/etc/ansible/roles/mediapeers.nfs
├── README.md
├── defaults
│   └── main.yml
├── handlers
│   └── main.yml
├── meta
│   └── main.yml
├── tasks
│   ├── main.yml
│   ├── setup-Debian.yml
│   └── setup-RedHat.yml
├── templates
│   └── exports.j2
├── tests
│   ├── inventory
│   └── test.yml
└── vars
    ├── Debian.yml
    └── RedHat.yml
```

BEST PRACTICES

- Give `name` to your tasks and playbooks
- Split up your tasks into groups and `include` them
- Bundle configuration into reusable roles
- Use dedicated modules instead of `shell` and `command`

WHAT DOES ANSIBLE GIVE US?

- Automated environment setup
- Human-friendly, with a gentle learning curve
- Modular and flexible
- Working on different linux flavours
- Share reusable recipes

WHAT ELSE IS THERE?

WHAT ELSE IS THERE?

- Integration with Vagrant

WHAT ELSE IS THERE?

- Integration with Vagrant
- **Testing of playbooks and roles**

WHAT ELSE IS THERE?

- Integration with Vagrant
- Testing of playbooks and roles
- **Dynamic inventories**

WHAT ELSE IS THERE?

- Integration with Vagrant
- Testing of playbooks and roles
- Dynamic inventories
- **Integration with clouds**

WHAT ELSE IS THERE?

- Integration with Vagrant
- Testing of playbooks and roles
- Dynamic inventories
- Integration with clouds
- **Integration with docker**

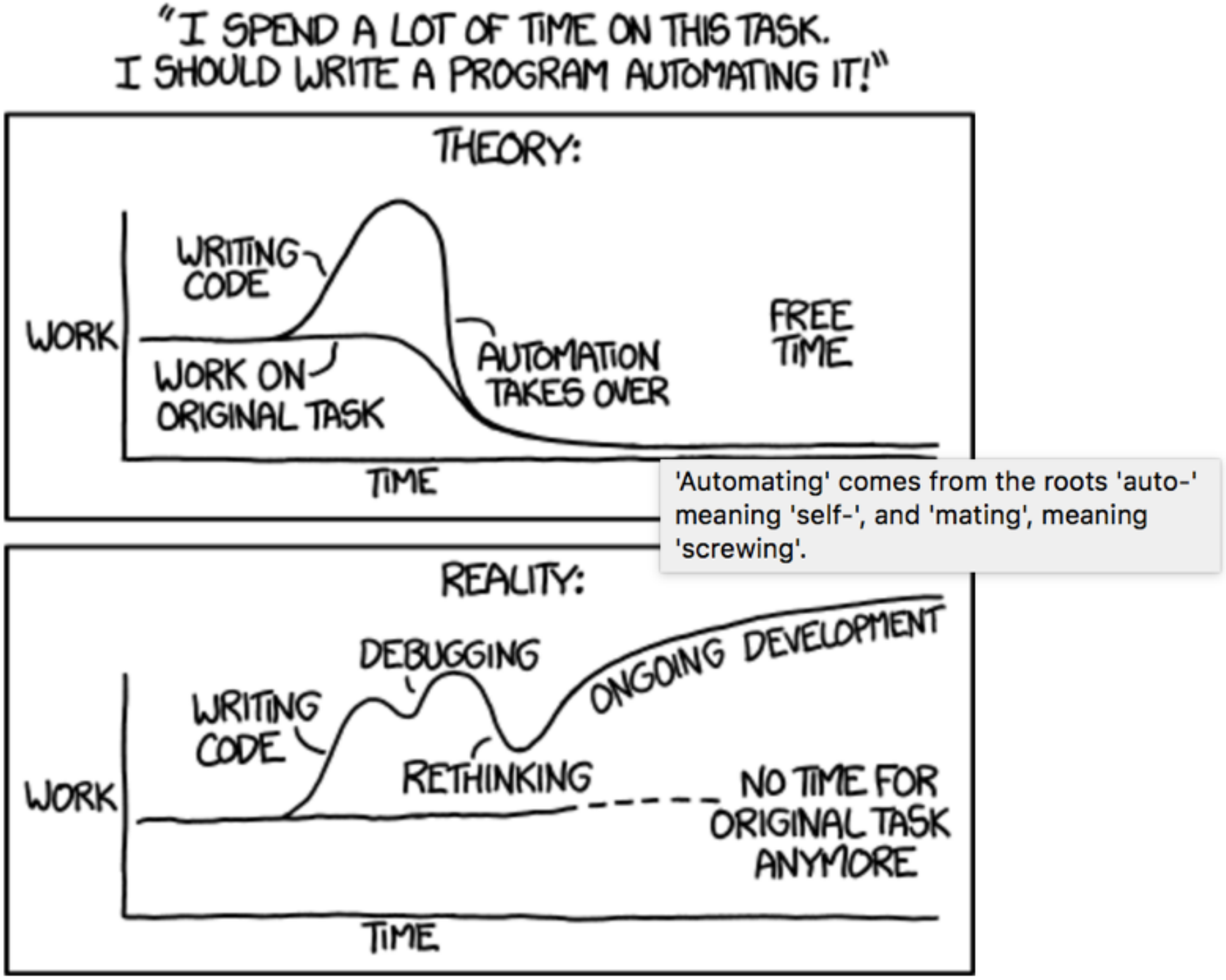
WHAT ELSE IS THERE?

- Integration with Vagrant
- Testing of playbooks and roles
- Dynamic inventories
- Integration with clouds
- Integration with docker
- **Application deployments**

WHAT ELSE IS THERE?

- Integration with Vagrant
- Testing of playbooks and roles
- Dynamic inventories
- Integration with clouds
- Integration with docker
- Application deployments
- **CI/CD**

THERE'S XKCD ABOUT THIS



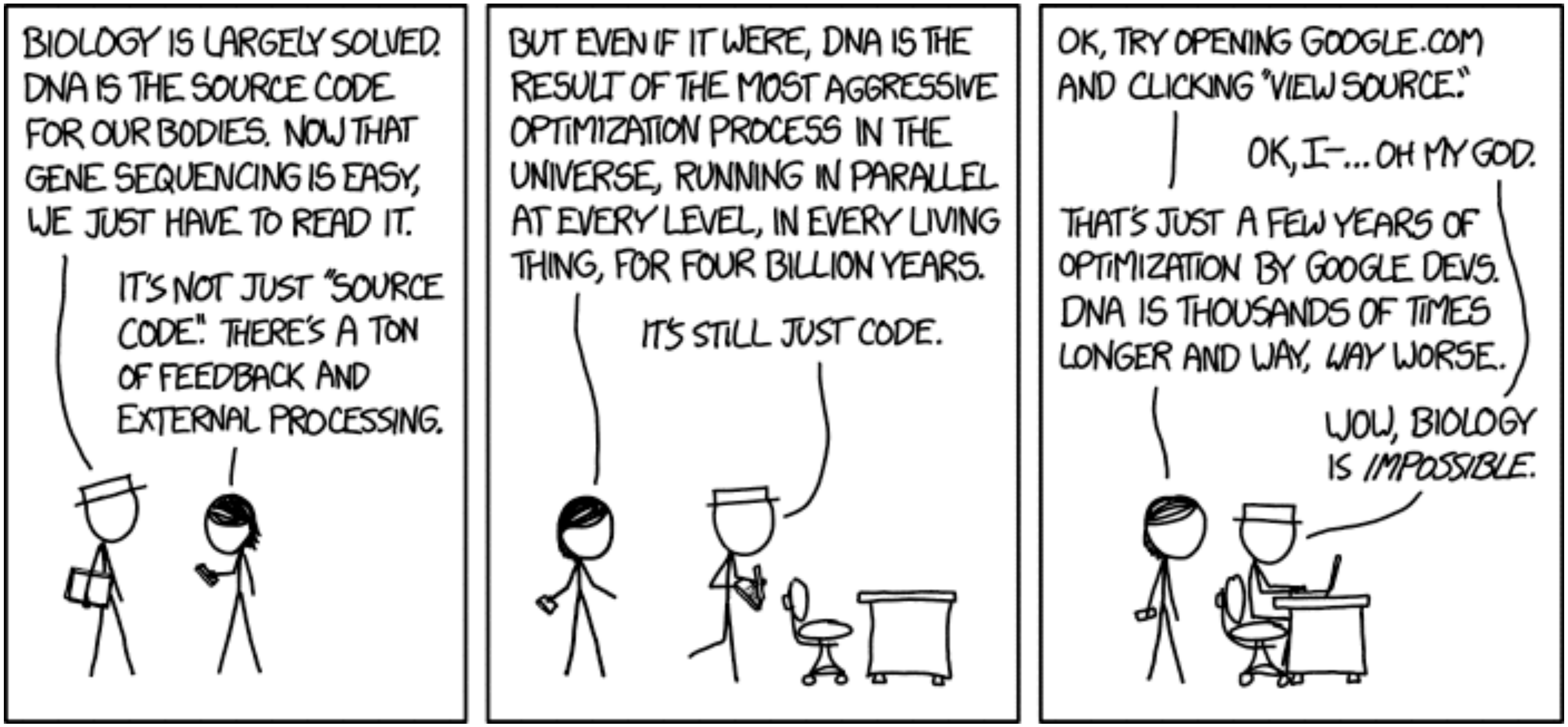
'Automating' comes from the roots 'auto-' meaning 'self-', and 'mating', meaning 'screwing'.

ardigen
BIOINFORMATICS

code
against
cancer

Join our team!

THERE'S XKCD ABOUT BIOINFORMATICS, TOO


BIOLOGY IS LARGELY SOLVED. DNA IS THE SOURCE CODE FOR OUR BODIES. NOW THAT GENE SEQUENCING IS EASY, WE JUST HAVE TO READ IT.
IT'S NOT JUST "SOURCE CODE." THERE'S A TON OF FEEDBACK AND EXTERNAL PROCESSING.
BUT EVEN IF IT WERE, DNA IS THE RESULT OF THE MOST AGGRESSIVE OPTIMIZATION PROCESS IN THE UNIVERSE, RUNNING IN PARALLEL AT EVERY LEVEL, IN EVERY LIVING THING, FOR FOUR BILLION YEARS.
IT'S STILL JUST CODE.
OK, TRY OPENING GOOGLE.COM AND CLICKING "VIEW SOURCE."
OK, I–... OH MY GOD.
THAT'S JUST A FEW YEARS OF OPTIMIZATION BY GOOGLE DEVS. DNA IS THOUSANDS OF TIMES LONGER AND WAY, WAY WORSE.
WOW, BIOLOGY IS *IMPOSSIBLE*.