Basic Computing Skills

MLGEO Lecture 2

Marine Denolle @ UW mdenolle@uw.edu

Goals of class

1. Literature review Guidelines & Goals
2. Textbook and other resources
    a. Our class working textbook
    b. Borrowed materials from EarthDataScience.
3. CyberInfrastructure
4. Get to know your cohort and brainstorm on cool data sets. Check MLGEO-data set and suggest a contribution.

5. Login to the Hub: https://jupyter.rttl.uw.edu/2024-autumn-ess-469-a
6. Basic Bash: example in jupyterhub
7. Basic Jupyter Environment: getting_started_guide by the UW IT
8. Python Environments: exercise, make your own environment and test in python.

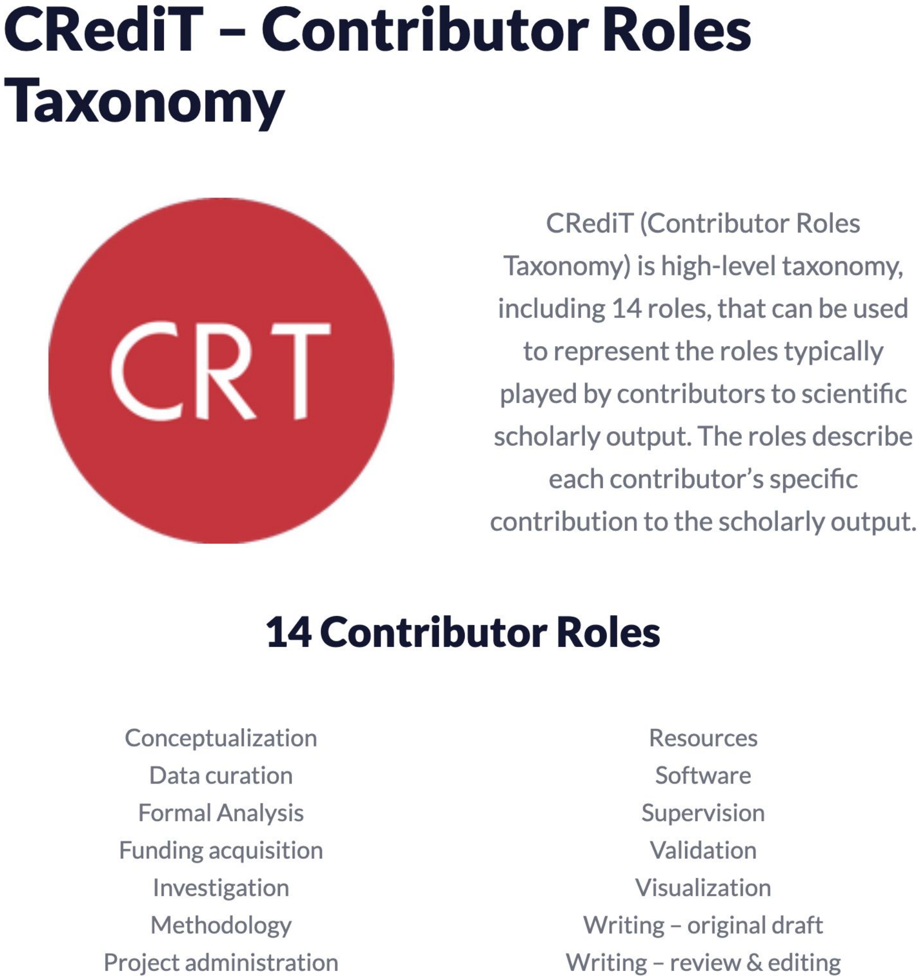
Literature Review

List of relevant papers

1. Full citation (including doi):
2. Scientific Motivation:
3. Data Source, Type, and modality:
4. Method [*]: if they use ML, what is the baseline model? What model architecture did they use? How did they train their model?
5. Key points of the research findings:
6. [If applicable] Data: does the study use open-access data (check for public repositories)? What is the guidance on how to access the data?
7. [If applicable] Is the workflow described, and could it be reproduced? (describe data and computational workflow
8. [If applicable] Does the manuscript provide a link (or zip file) to a code or notebook that can reproduce the work?
9. Optional: What type of open access does this journal offer?

[*] for review papers, did they, and how many papers did they use to review? Did they have any systematic meta-analysis process?

Ethics in publishing



CRediT – Contributor Roles Taxonomy

CRediT (Contributor Roles Taxonomy) is high-level taxonomy, including 14 roles, that can be used to represent the roles typically played by contributors to scientific scholarly output. The roles describe each contributor's specific contribution to the scholarly output.

14 Contributor Roles

- Conceptualization
- Data curation
- Formal Analysis
- Funding acquisition
- Investigation
- Methodology
- Project administration
- Resources
- Software
- Supervision
- Validation
- Visualization
- Writing – original draft
- Writing – review & editing

Conceptualization – Ideas; formulation or evolution of overarching research goals and aims.
Data curation – Management activities to annotate (produce metadata), scrub data and maintain research data (including software code, where it is necessary for interpreting the data itself) for initial use and later re-use.
Formal analysis – Application of statistical, mathematical, computational, or other formal techniques to analyze or synthesize study data.
Funding acquisition - Acquisition of the financial support for the project leading to this publication.
Investigation – Conducting a research and investigation process, specifically performing the experiments, or data/evidence collection.
Methodology – Development or design of methodology; creation of models.
Project administration – Management and coordination responsibility for the research activity planning and execution.
Resources – Provision of study materials, reagents, materials, patients, laboratory samples, animals, instrumentation, computing resources, or other analysis tools.
Software – Programming, software development; designing computer programs; implementation of the computer code and supporting algorithms; testing of existing code components.
Supervision – Oversight and leadership responsibility for the research activity planning and execution, including mentorship external to the core team.
Validation – Verification, whether as a part of the activity or separate, of the overall replication/reproducibility of results/experiments and other research outputs.
Visualization – Preparation, creation and/or presentation of the published work, specifically visualization/data presentation.
Writing – original draft – Preparation, creation and/or presentation of the published work, specifically writing the initial draft (including substantive translation).
Writing – review & editing – Preparation, creation and/or presentation of the published work by those from the original research group, specifically critical review, commentary or revision – including pre- or post-publication stages.

CyberInfrastructure - CI

1. Hardware
   a. Local hardware, on-prem (laptop, workstations, towers, ...)
   b. Clusters: large computers dedicated to large scale simulation. Hyak.
   c. Cloud Computing (next)
2. Software
   a. Programming languages (python)
   b. Operating System: OSX, Linux, Windows
   c. Scripted code (e.g., *.py)
   d. Containers are lightweight, portable, self-contained environment that include applications and all dependencies necessary to run across different computing environment. Docker is a tool to manage containers.

What is a cloud?

Huge buildings filled with servers owned by Microsoft, Amazon, Google, etc.

Marine Denolle
mdenolle@uw.edu
Zoe Krauss
zkrauss@uw.edu
Naomi Alterman
naomila@uw.edu

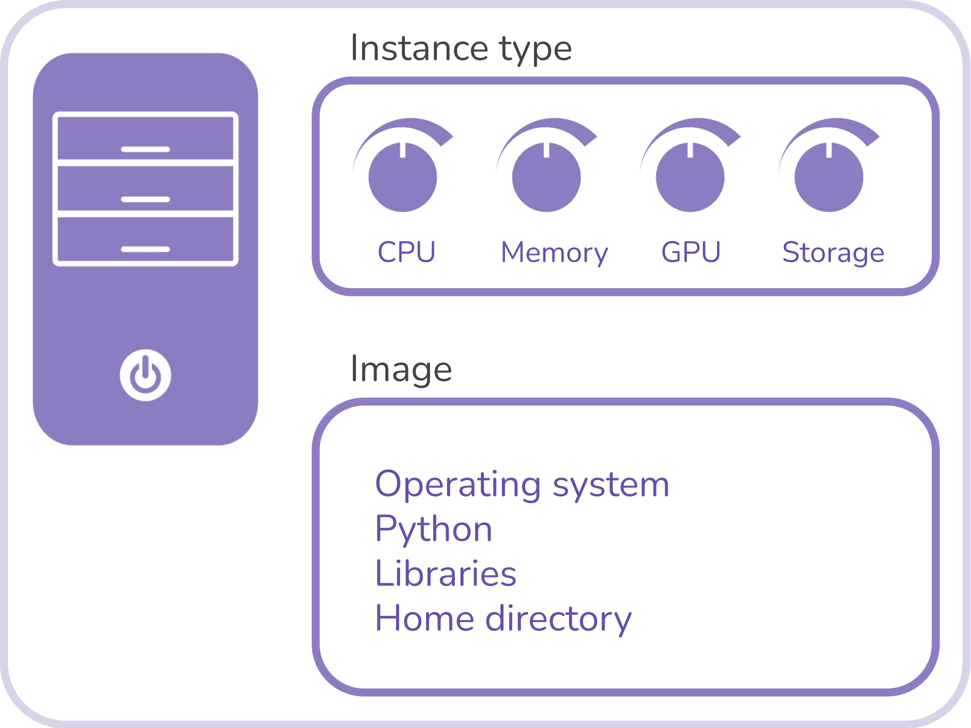
Compute

"Virtual machines" (VMs)

Storage

File stores, "Object storage"

Virtual machines: emulators of computer system



Instance type

CPU Memory GPU Storage

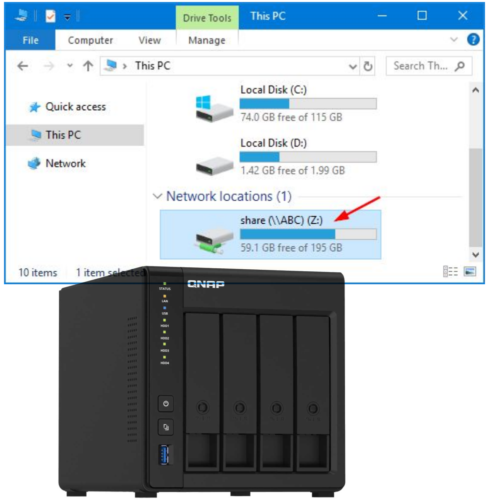
Image

Operating system
Python
Libraries
Home directory

Marine Denolle mdenolle@uw.edu
Naomi Alterman naomila@uw.edu
Rob Fatland rob5@uw.edu

Marine Denolle mdenolle@uw.edu
Naomi Alterman naomila@uw.edu
Rob Fatland rob5@uw.edu

Storage

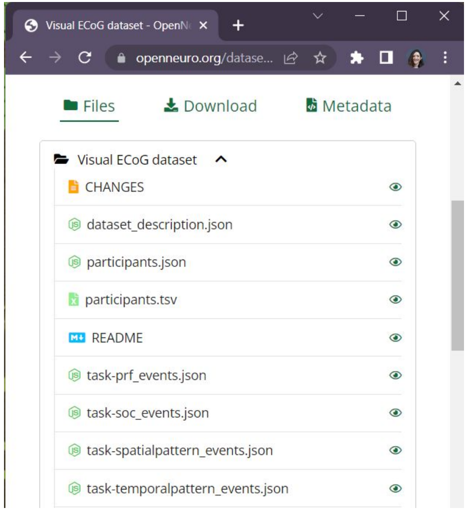


Local Disk (C:)
74.0 GB free of 115 GB

Local Disk (D:)
1.42 GB free of 1.99 GB

Network locations (1)

share (\\ABC) (Z:)
59.1 GB free of 195 GB



Visual ECoG dataset
- CHANGES
- dataset_description.json
- participants.json
- participants.tsv
- README
- task-prf_events.json
- task-soc_events.json
- task-spatialpattern_events.json
- task-temporalpattern_events.json

Block Storage

Fastest, usually one per-VM,
Usually 4 GB - 512 GB
$$$ ~$0.16 / GB

File Storage

Most flexible
< 100 TB
$$ ~$0.06 / GB

Object Storage

Most cost-effective
Unlimited
$ ~$0.01 / GB

How to access data?

Same region
0.1 - 10 Gbs/s

Scedc data ~ 100TB
Poro tomo DAS data ~ 50TB
(soon) **Earthscope/IRIS** data

NASA
SAR

Blob Storage

Planetary Computer

Sentinel
Landsat
NOAA

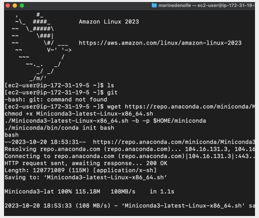
Earth Engine

70 PBs of geospatial data
Landast
Sentinel
DEMs
Surface water

How to access compute?

Single VM

Exploration & ML training



```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-172-31-19-5 ~]$ ls
[ec2-user@ip-172-31-19-5 ~]$ git
-bash: git: command not found
[ec2-user@ip-172-31-19-5 ~]$ wget https://repo.anaconda.com/miniconda/M
chmod +x Miniconda3-latest-Linux-x86_64.sh
./Miniconda3-latest-Linux-x86_64.sh -b -p $HOME/miniconda
./miniconda/bin/conda init bash
bash
--2023-10-20 18:53:31--  https://repo.anaconda.com/miniconda/Miniconda3
Resolving repo.anaconda.com (repo.anaconda.com)... 104.16.131.3, 104.16
Connecting to repo.anaconda.com (repo.anaconda.com)|104.16.131.3|:443..
HTTP request sent, awaiting response... 200 OK
Length: 120771089 (115M) [application/x-sh]
Saving to: ‘Miniconda3-latest-Linux-x86_64.sh’

Miniconda3-lat 100% 115.18M   108MB/s    in 1.1s

2023-10-20 18:53:33 (108 MB/s) - ‘Miniconda3-latest-Linux-x86_64.sh’ sa
```

- Ssh to Linux OS - install all packages or preload Machine Image (AMI)

- Use institution-hosted jupyterHub (at cost to institutions)

Many VMs

Deployment @ scale

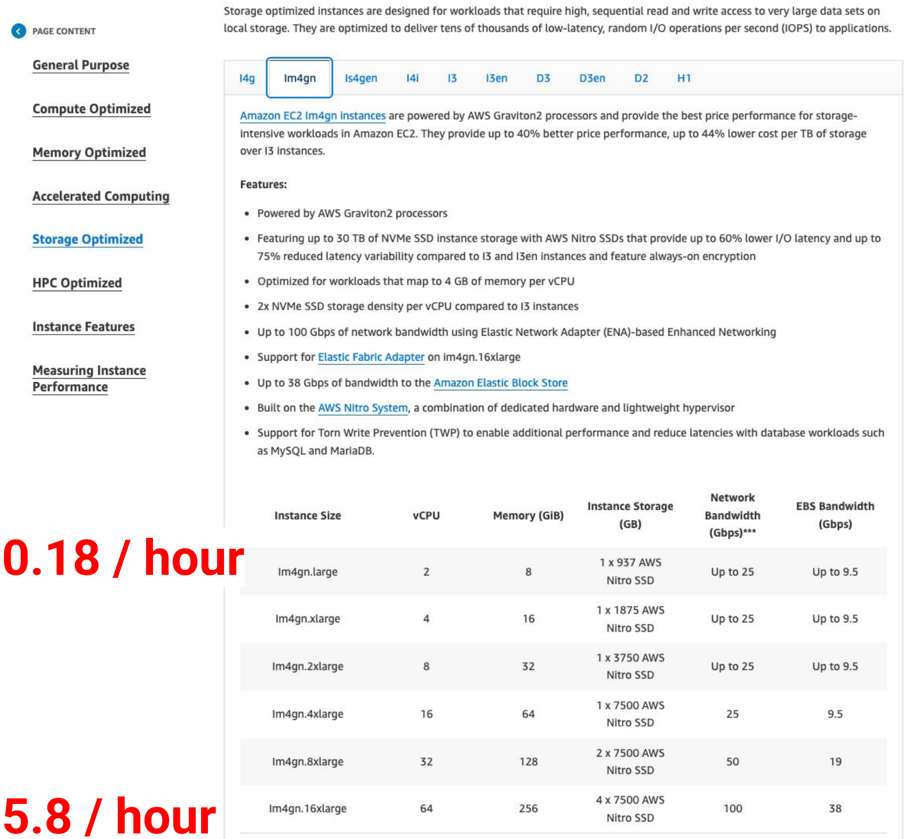
- Containerize software using Docker

- DIY cluster with kubernetes

- Batch services to ease cluster management

Cloud Concepts with AWS

EC2: Elastic Compute.



PAGE CONTENT

- General Purpose
- Compute Optimized
- Memory Optimized
- Accelerated Computing
- Storage Optimized
- HPC Optimized
- Instance Features
- Measuring Instance Performance

Storage optimized instances are designed for workloads that require high, sequential read and write access to very large data sets on local storage. They are optimized to deliver tens of thousands of low-latency, random I/O operations per second (IOPS) to applications.

Amazon EC2 Im4gn instances are powered by AWS Graviton2 processors and provide the best price performance for storage-intensive workloads in Amazon EC2. They provide up to 40% better price performance, up to 44% lower cost per TB of storage over I3 instances.

Features:

- Powered by AWS Graviton2 processors
- Featuring up to 30 TB of NVMe SSD instance storage with AWS Nitro SSDs that provide up to 60% lower I/O latency and up to 75% reduced latency variability compared to I3 and I3en instances and feature always-on encryption
- Optimized for workloads that map to 4 GB of memory per vCPU
- 2x NVMe SSD storage density per vCPU compared to I3 instances
- Up to 100 Gbps of network bandwidth using Elastic Network Adapter (ENA)-based Enhanced Networking
- Support for Elastic Fabric Adapter on im4gn.16xlarge
- Up to 38 Gbps of bandwidth to the Amazon Elastic Block Store
- Built on the AWS Nitro System, a combination of dedicated hardware and lightweight hypervisor
- Support for Torn Write Prevention (TWP) to enable additional performance and reduce latencies with database workloads such as MySQL and MariaDB.

Instance Size	vCPU	Memory (GiB)	Instance Storage (GB)	Network Bandwidth (Gbps)***	EBS Bandwidth (Gbps)
Im4gn.large	2	8	1 x 937 AWS Nitro SSD	Up to 25	Up to 9.5
Im4gn.xlarge	4	16	1 x 1875 AWS Nitro SSD	Up to 25	Up to 9.5
Im4gn.2xlarge	8	32	1 x 3750 AWS Nitro SSD	Up to 25	Up to 9.5
Im4gn.4xlarge	16	64	1 x 7500 AWS Nitro SSD	25	9.5
Im4gn.8xlarge	32	128	2 x 7500 AWS Nitro SSD	50	19
Im4gn.16xlarge	64	256	4 x 7500 AWS Nitro SSD	100	38

$ 0.18 / hour

$ 5.8 / hour

Compute

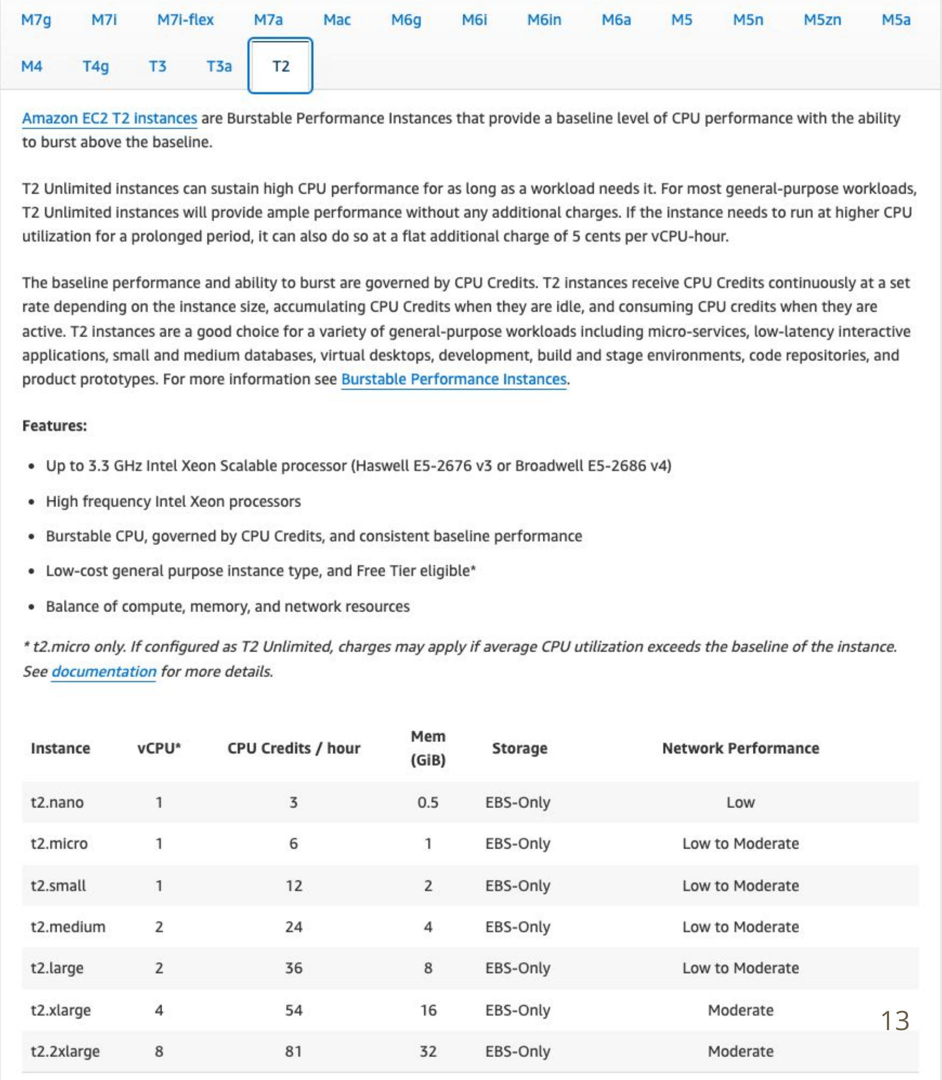
“Virtual machines” (VMs)

General purpose EC2

Small - testing - microservice

$0.0058 / hour

$0.3712 / hour



Amazon EC2 T2 instances are Burstable Performance Instances that provide a baseline level of CPU performance with the ability to burst above the baseline.

T2 Unlimited instances can sustain high CPU performance for as long as a workload needs it. For most general-purpose workloads, T2 Unlimited instances will provide ample performance without any additional charges. If the instance needs to run at higher CPU utilization for a prolonged period, it can also do so at a flat additional charge of 5 cents per vCPU-hour.

The baseline performance and ability to burst are governed by CPU Credits. T2 instances receive CPU Credits continuously at a set rate depending on the instance size, accumulating CPU Credits when they are idle, and consuming CPU credits when they are active. T2 instances are a good choice for a variety of general-purpose workloads including micro-services, low-latency interactive applications, small and medium databases, virtual desktops, development, build and stage environments, code repositories, and product prototypes. For more information see Burstable Performance Instances.

Features:

- Up to 3.3 GHz Intel Xeon Scalable processor (Haswell E5-2676 v3 or Broadwell E5-2686 v4)
- High frequency Intel Xeon processors
- Burstable CPU, governed by CPU Credits, and consistent baseline performance
- Low-cost general purpose instance type, and Free Tier eligible*
- Balance of compute, memory, and network resources

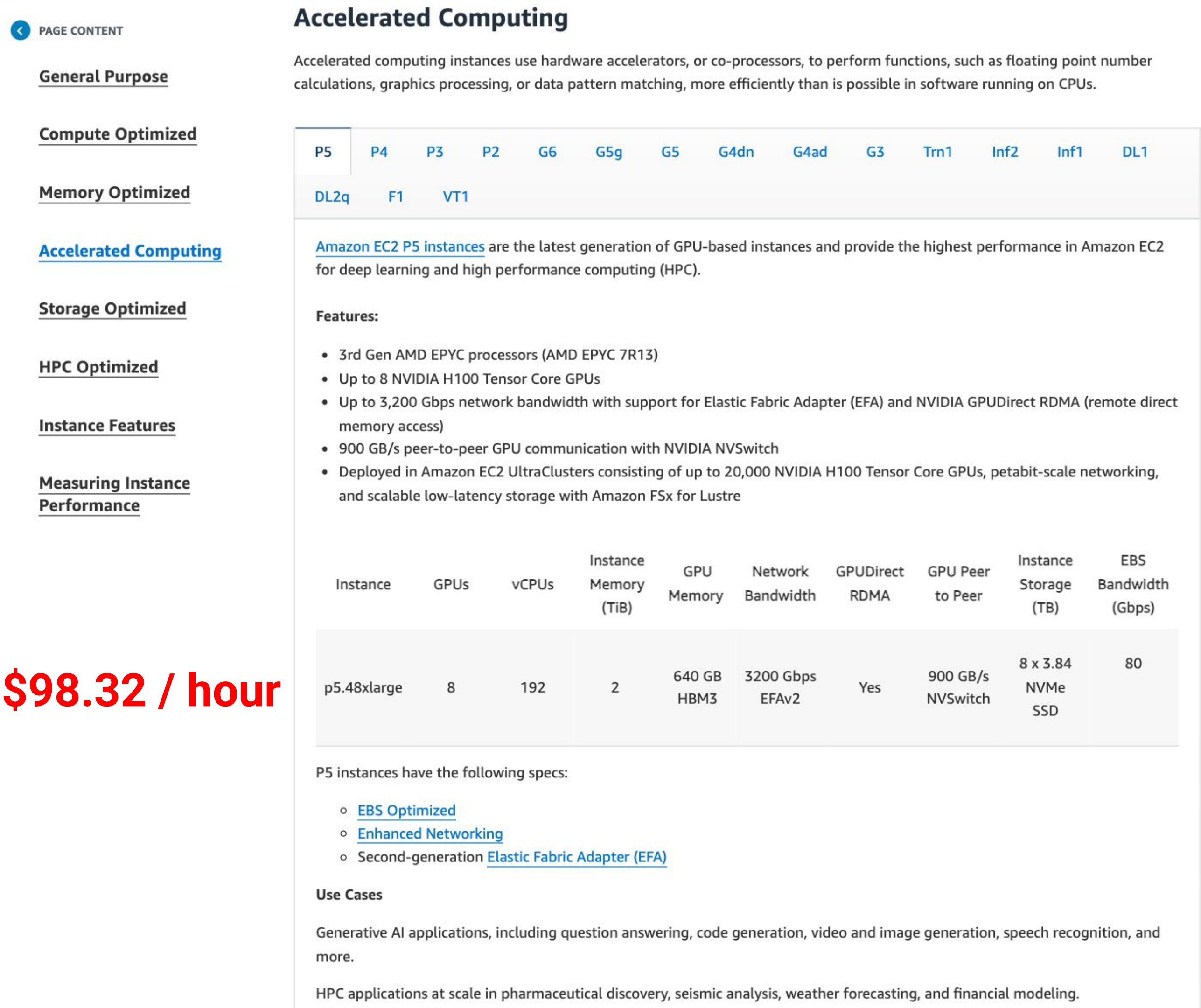
** t2.micro only. If configured as T2 Unlimited, charges may apply if average CPU utilization exceeds the baseline of the instance. See documentation for more details.*

Instance	vCPU*	CPU Credits / hour	Mem (GiB)	Storage	Network Performance
t2.nano	1	3	0.5	EBS-Only	Low
t2.micro	1	6	1	EBS-Only	Low to Moderate
t2.small	1	12	2	EBS-Only	Low to Moderate
t2.medium	2	24	4	EBS-Only	Low to Moderate
t2.large	2	36	8	EBS-Only	Low to Moderate
t2.xlarge	4	54	16	EBS-Only	Moderate
t2.2xlarge	8	81	32	EBS-Only	Moderate

Marine Denolle mdenolle@uw.edu

Heterogeneous computing EC2 instances.

Machine Learning training or big Xcorr-ready workflows



$98.32 / hour

Accelerated Computing

Accelerated computing instances use hardware accelerators, or co-processors, to perform functions, such as floating point number calculations, graphics processing, or data pattern matching, more efficiently than is possible in software running on CPUs.

Amazon EC2 P5 instances are the latest generation of GPU-based instances and provide the highest performance in Amazon EC2 for deep learning and high performance computing (HPC).

Features:

- 3rd Gen AMD EPYC processors (AMD EPYC 7R13)
- Up to 8 NVIDIA H100 Tensor Core GPUs
- Up to 3,200 Gbps network bandwidth with support for Elastic Fabric Adapter (EFA) and NVIDIA GPUDirect RDMA (remote direct memory access)
- 900 GB/s peer-to-peer GPU communication with NVIDIA NVSwitch
- Deployed in Amazon EC2 UltraClusters consisting of up to 20,000 NVIDIA H100 Tensor Core GPUs, petabit-scale networking, and scalable low-latency storage with Amazon FSx for Lustre

Instance	GPUs	vCPUs	Instance Memory (TiB)	GPU Memory	Network Bandwidth	GPUDirect RDMA	GPU Peer to Peer	Instance Storage (TB)	EBS Bandwidth (Gbps)
p5.48xlarge	8	192	2	640 GB HBM3	3200 Gbps EFAv2	Yes	900 GB/s NVSwitch	8 x 3.84 NVMe SSD	80

P5 instances have the following specs:

- EBS Optimized
- Enhanced Networking
- Second-generation Elastic Fabric Adapter (EFA)

Use Cases

Generative AI applications, including question answering, code generation, video and image generation, speech recognition, and more.

HPC applications at scale in pharmaceutical discovery, seismic analysis, weather forecasting, and financial modeling.

Cloud Concepts with AWS

EC2: Elastic Compute At Scale

Kubernetes

Kubernetes is a container orchestration platform that automates the deployment, scaling, and management of containerized applications.

AWS Batch

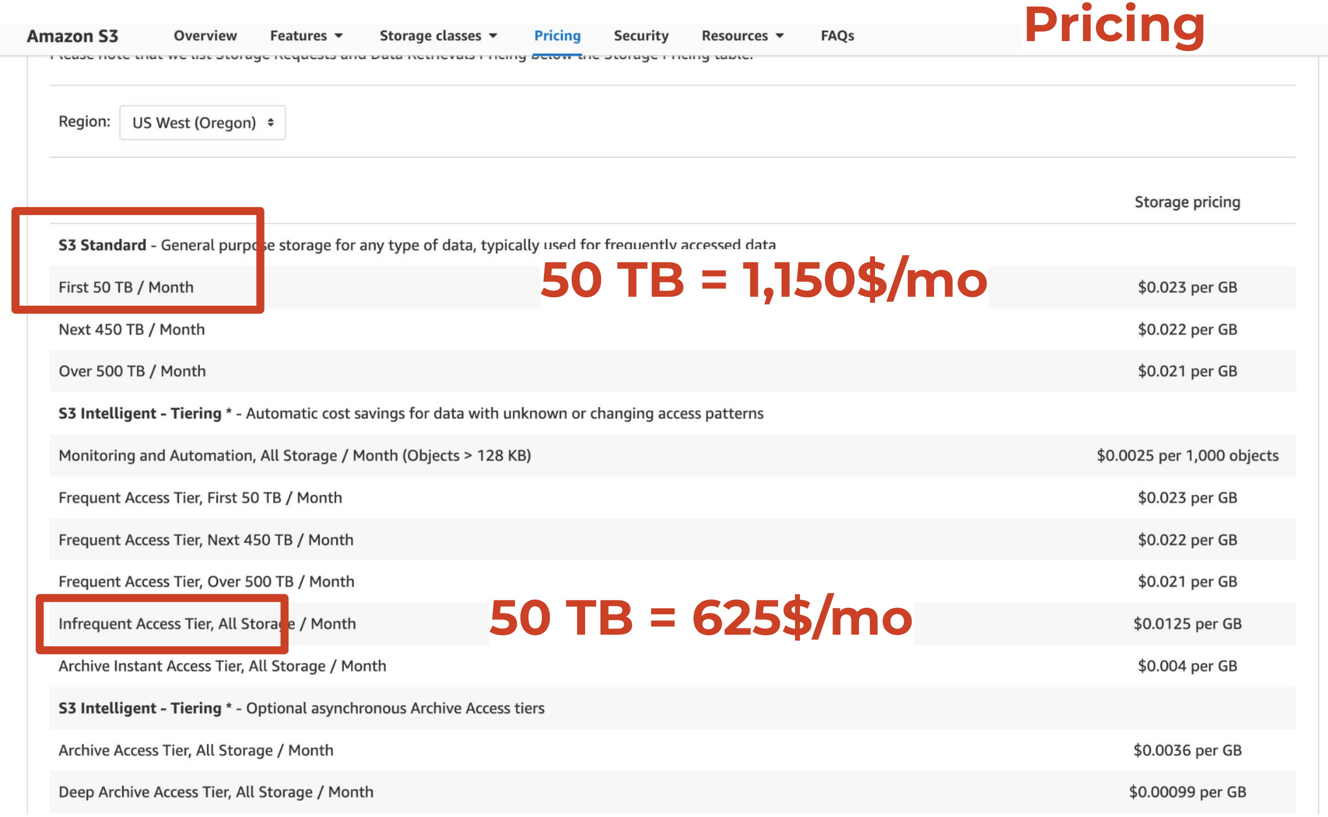
AWS Batch is a fully managed service that enables users to run batch computing jobs on the AWS cloud efficiently and at any scale.

Kubernetes is a versatile platform for containerized applications with complex orchestration needs, while *AWS Batch* is tailored for efficiently running and managing batch processing jobs with minimal overhead.

This course will show how to use AWS Batch

Cloud Concepts with AWS

Pricing



Amazon S3 | Overview | Features ▾ | Storage classes ▾ | Pricing | Security | Resources ▾ | FAQs

Please note that we list Storage Requests and Data Retrievals Pricing below the Storage Pricing table.

Region: US West (Oregon) ⬍

	Storage pricing
S3 Standard - General purpose storage for any type of data, typically used for frequently accessed data	
First 50 TB / Month	$0.023 per GB
Next 450 TB / Month	$0.022 per GB
Over 500 TB / Month	$0.021 per GB
S3 Intelligent - Tiering * - Automatic cost savings for data with unknown or changing access patterns	
Monitoring and Automation, All Storage / Month (Objects > 128 KB)	$0.0025 per 1,000 objects
Frequent Access Tier, First 50 TB / Month	$0.023 per GB
Frequent Access Tier, Next 450 TB / Month	$0.022 per GB
Frequent Access Tier, Over 500 TB / Month	$0.021 per GB
Infrequent Access Tier, All Storage / Month	$0.0125 per GB
Archive Instant Access Tier, All Storage / Month	$0.004 per GB
S3 Intelligent - Tiering * - Optional asynchronous Archive Access tiers	
Archive Access Tier, All Storage / Month	$0.0036 per GB
Deep Archive Access Tier, All Storage / Month	$0.00099 per GB

50 TB = 1,150$/mo

50 TB = 625$/mo

Storage

File stores, “Object storage”

Amazon S3

Marine Denolle mdenolle@uw.edu

Who Pays for Compute?

You. You own the account, you pay with a credit card.

CloudBank. An NSF project to support the adoption of cloud computing in NSF supported research and education. You may apply to cloudbank credits when submitting a proposal to specific RFPs.

Your institution. They may support cloud credit applications through partnerships between cloud providers and the institutions.

The national archives. It is unlikely that archives such as Earthscope Data Services will pay for large-scale computing, but they will offer basic jupyter Hubs for data explorations.

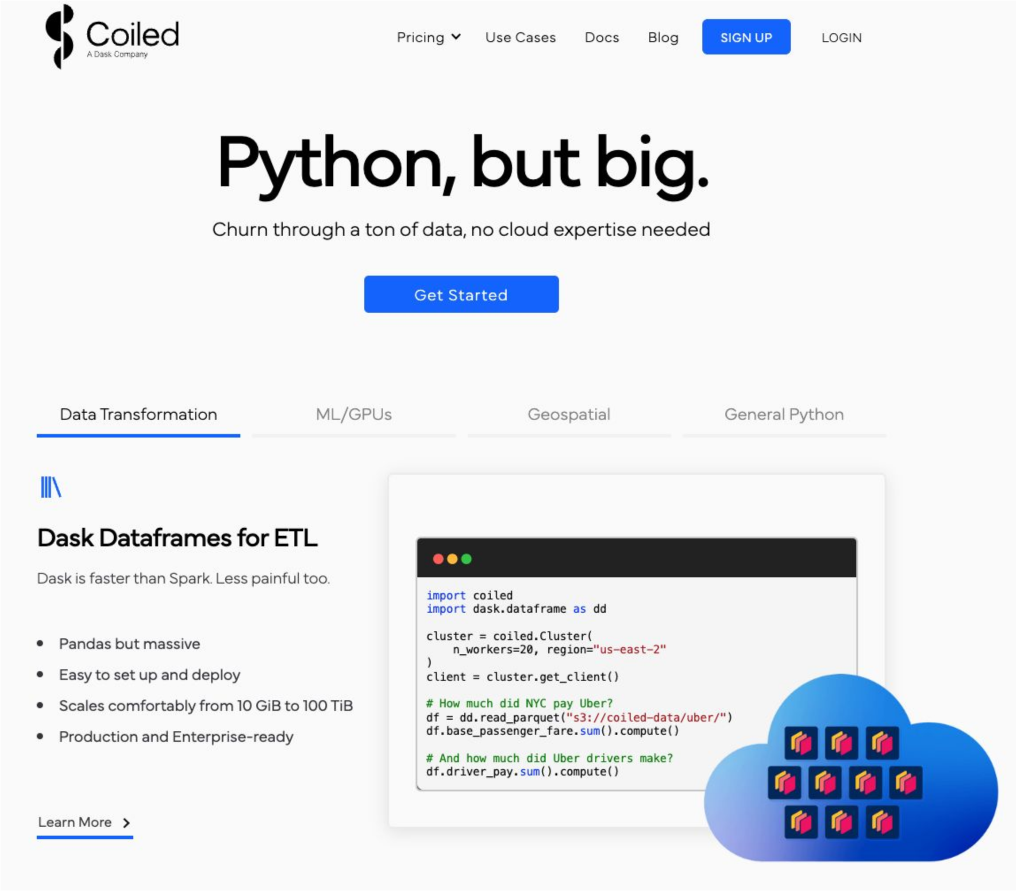
Abstracting the Cloud

Infrastructure-as-a-service (Iaas) is a way to abstract the compute and make the experience of running a job as *local* as possible.

Coiled is an example for Dask, Python users.

- It borrows from Dask.
- It maps your local environment and run it there (but the local environment has to be simple)

No additional fee for small compute but the Cloud costs.



Python, but big.

Churn through a ton of data, no cloud expertise needed

Dask Dataframes for ETL

Dask is faster than Spark. Less painful too.

- Pandas but massive
- Easy to set up and deploy
- Scales comfortably from 10 GiB to 100 TiB
- Production and Enterprise-ready

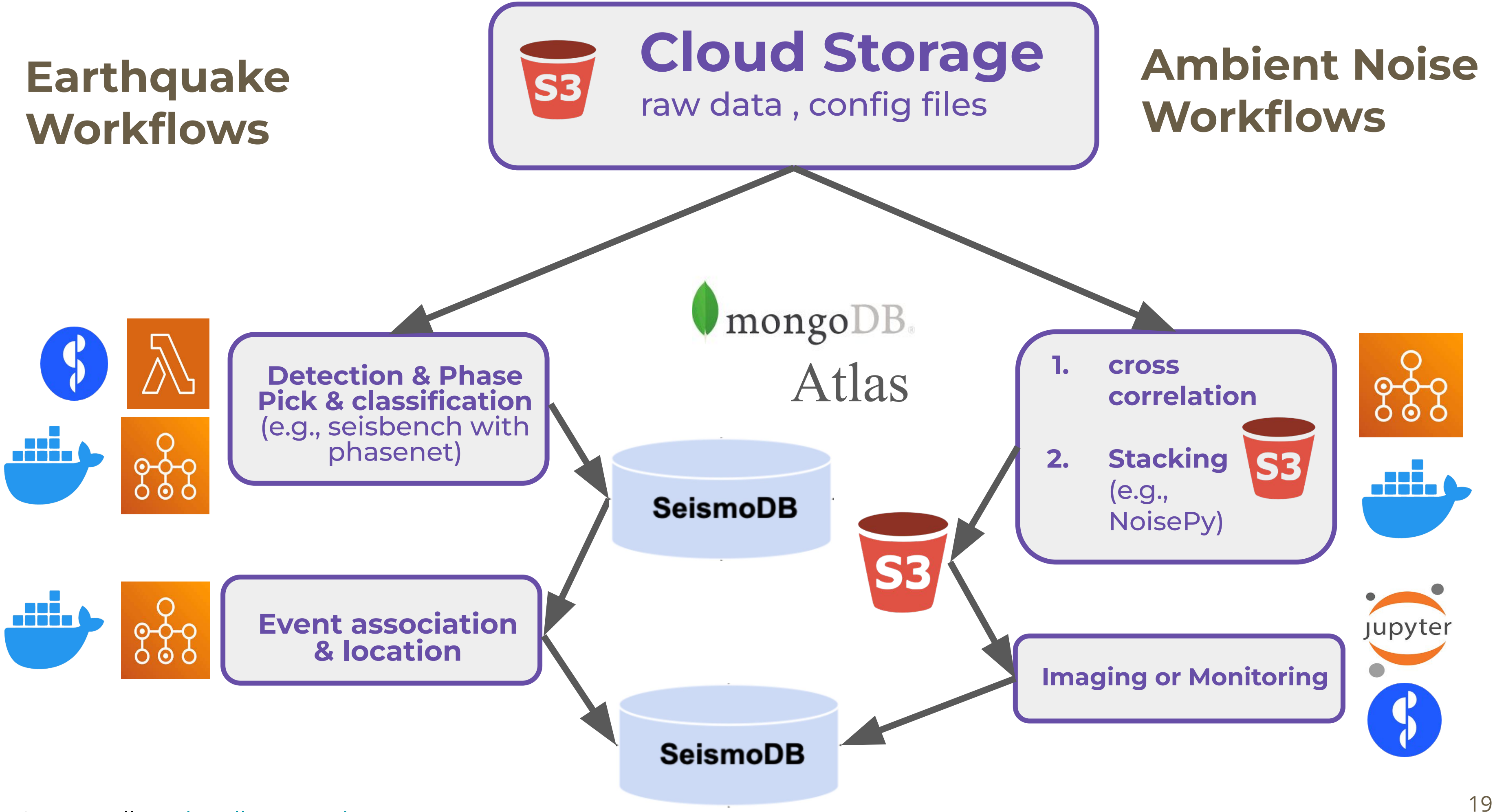
```
import coiled
import dask.dataframe as dd

cluster = coiled.Cluster(
    n_workers=20, region="us-east-2"
)
client = cluster.get_client()

# How much did NYC pay Uber?
df = dd.read_parquet("s3://coiled-data/uber/")
df.base_passenger_fare.sum().compute()

# And how much did Uber drivers make?
df.driver_pay.sum().compute()
```

Marine Denolle mdenolle@uw.edu



Cloud Storage

raw data , config files

Earthquake Workflows

Detection & Phase Pick & classification (e.g., seisbench with phasenet)

Event association & location

Ambient Noise Workflows

1. cross correlation
2. Stacking (e.g., NoisePy)

Imaging or Monitoring

What resources are available to me on the cloud?

Amazon Web Services

Microsoft Azure

Google Cloud

Ranges from:

Fully “built” systems with operating systems, programs like Jupyter, access to data, already installed

↓

Empty “from scratch” instances- you choose CPU, GPU, memory, etc. Infinitely scalable

Software

Programming language of the class: Python

Everybody opens up: Terminal, VsCode, of the JupyterHub.

We will create a conda environment and add the dependencies in the python environment.

Sign up for Github Education account if you can, or sign up for CoPilot

Get up to speed

Some useful resources

- https://www.earthdatascience.org/
- https://software-carpentry.org/lessons/

The tutorials and course materials will be done in Python.

Build Cohort - Chat about Data & cool problems

Get up to speed with computing

3. Basic shell

GUI: Graphical User Interface

Instructions from human to computer are done by clicking buttons on a mouse

Fun, easy, but not good if you want to repeat your task 1000s (watch out for carpal tunnel syndrome!)

CLI: Command Line Interface

Use terminal, rudimentary commands

Tutorials: EarthDataScience + software carpentry

```
pwd
ls
cd
mkdir
rmdir
cp
rm
```

4. Git, GitHub, GitLab

Git is an *open source version control tool*, **GitHub** and **GitLab** are companies that hosts Git repositories in the web and provides a web interface to interact with repos they host.

More in depth tutorials:

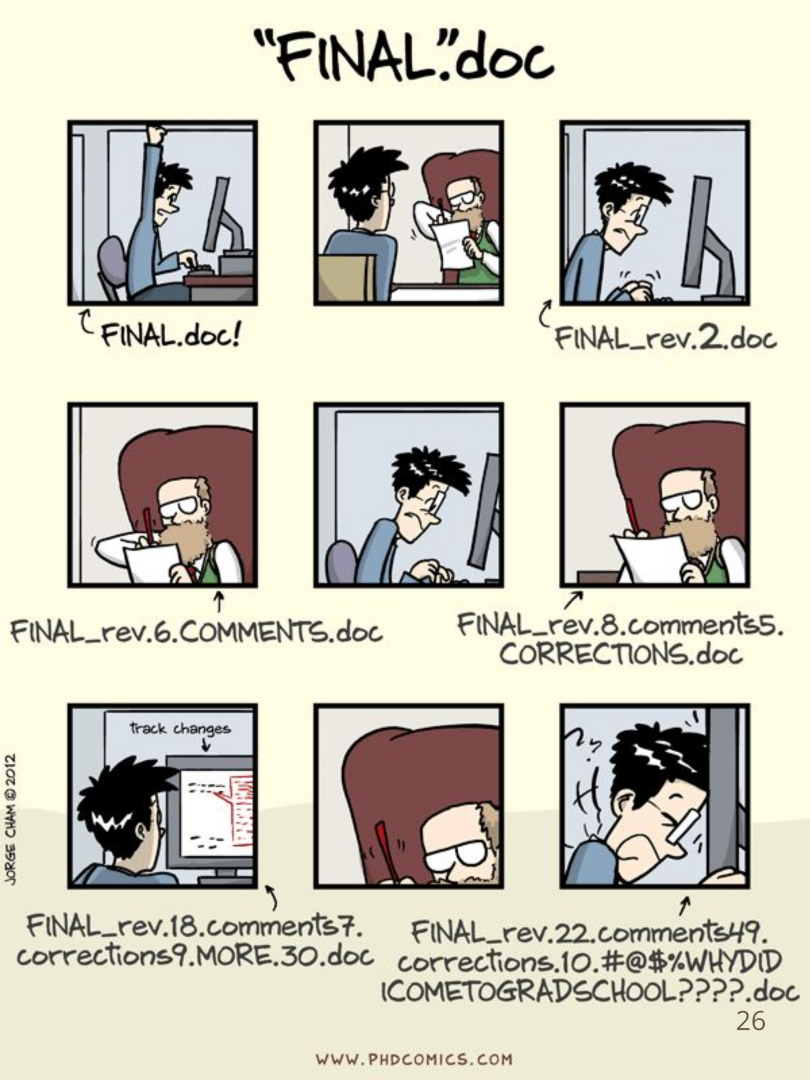
Software Carpentry

EarthDataScience

4. Version Control:

Iterations over docs
Collaborative comments and edits
How to keep track of changes?

Code Version Control is the scripting analog of Tracked changes


"FINAL".doc
FINAL.doc!
FINAL_rev.2.doc
FINAL_rev.6.COMMENTS.doc
FINAL_rev.8.comments5.
CORRECTIONS.doc
track changes
FINAL_rev.18.comments7.
corrections9.MORE.30.doc
FINAL_rev.22.comments49.
corrections.10.#@$%WHYDID
ICOMETOGRADSCHOOL????.doc
JORGE CHAM © 2012
WWW.PHDCOMICS.COM