
SPOC to MOOC, Extending Local Training to the HPC Community

Julia Mullen and Lauren Milechin

September 25, 2019



**Massachusetts
Institute of
Technology**



- **Challenges in HPC Education and Training**
- **Introduction to MOOCS**
- **Case Study: Understanding HPC Workflows and How to Exploit Them**
 - The initial one-on-one training
 - The first online course
 - The future refactored courses
- **Lessons Learned**



HPC Education Challenges

Audience

- **Learners want**
 - Formal training with certificates
 - Informal training to complete work task
- **Diverse learner background**
 - Range of ages
 - Range of computer literacy
 - Multi-cultural, multi-lingual

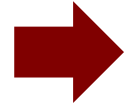
Content Selection

- **Workflows vary across domains requiring different solution techniques**
- **Learners have a range of background and skills**
- **Content needs include**
 - Basic Unix skills
 - Basic HPC concepts
 - HPC software development and troubleshooting

Delivery Mode

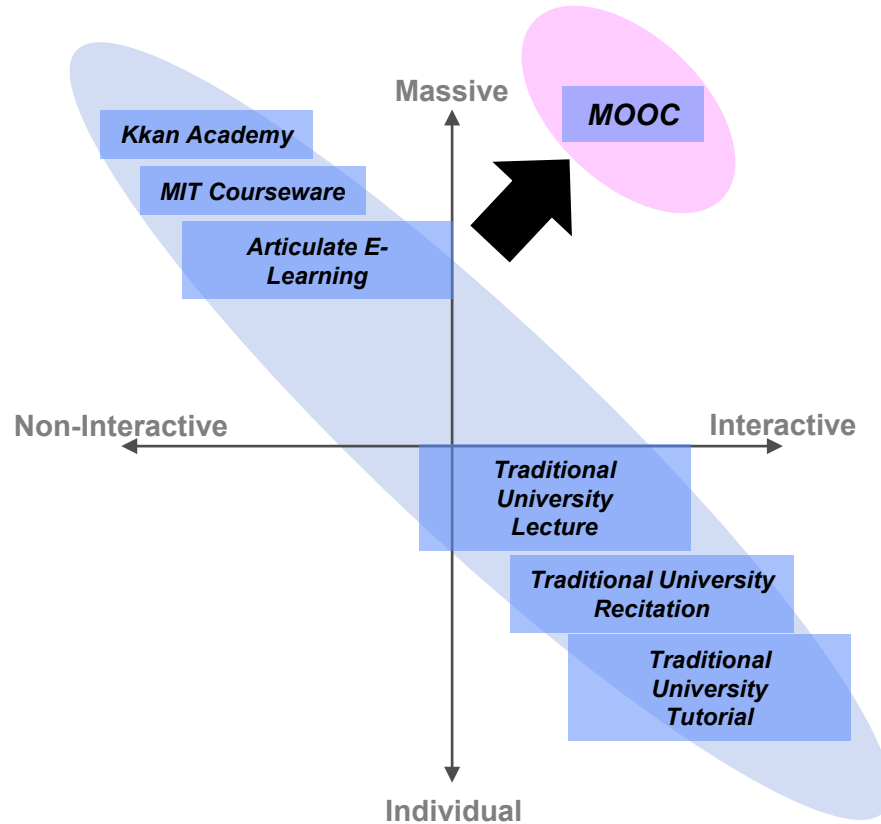
- **In person workshops**
 - Limited pool of expert trainers
 - HPC system access limited
- **Web resources**
 - Predominantly text
 - Tutorials for a given technology, e.g. MPI, OpenMP
 - Simplified examples, often tightly coupled to specific systems

Expanding HPC education and creating personalized “Just In Time” education and training is necessary but hard.



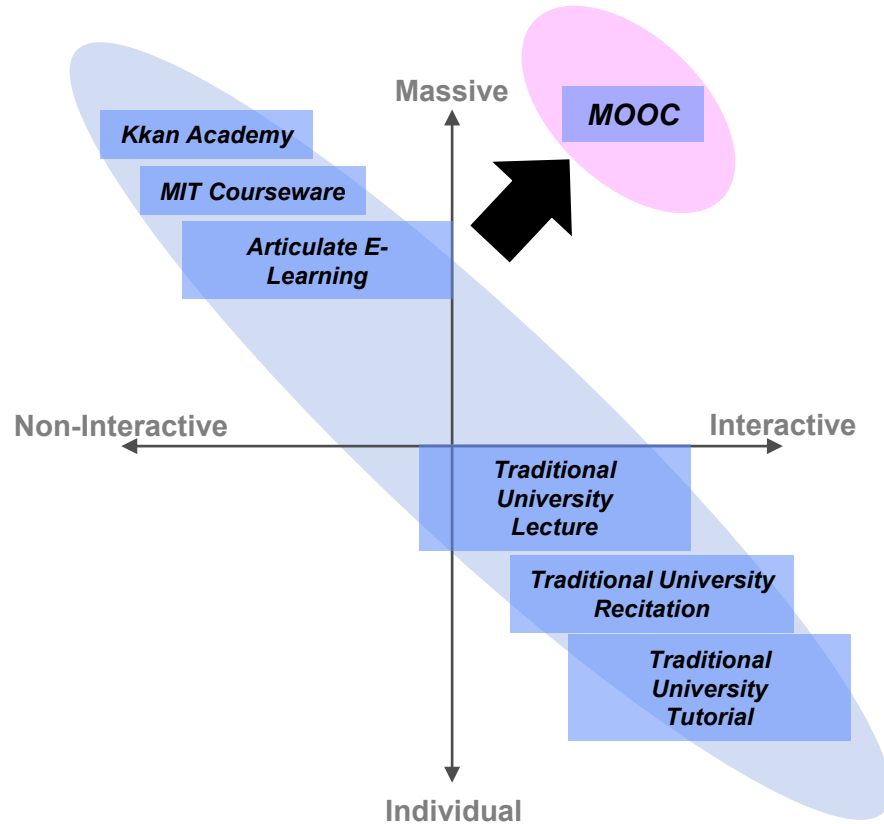
- **Challenges in HPC Education and Training**
- **Introduction to MOOCS**
- **Case Study: Understanding HPC Workflows and How to Exploit Them**
 - The initial one-on-one training
 - The first online course
 - The future refactored courses
- **Lessons Learned**

Why MOOCs?



Scaling

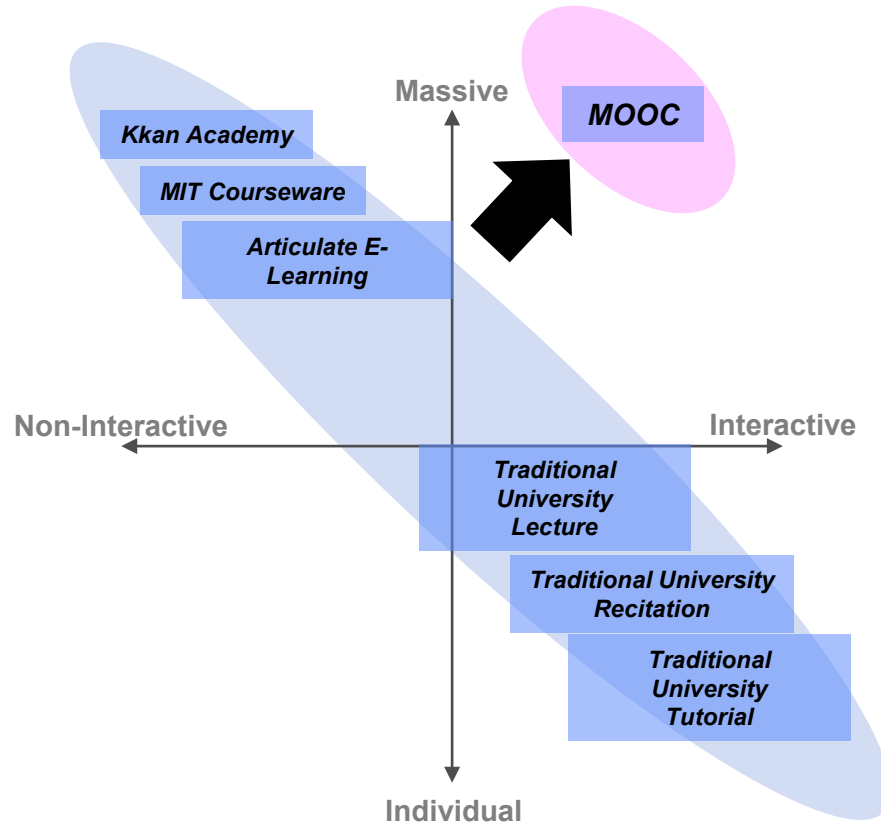
- 81 million learners across major providers
- Additional 13 million across independent Open edX sites
- 9+ thousand courses
- 25 languages (primarily Open edX)
- 33 providers worldwide



Pedagogy/Andragogy

- **Open**
 - No pre-requisites
 - Range of experience
- **Online learning**
 - Asynchronous
 - Self-paced
 - Instructor paced
- **Social learning – interactions among diverse learner groups**
- **Built to support theory and practice**

Why MOOCs?

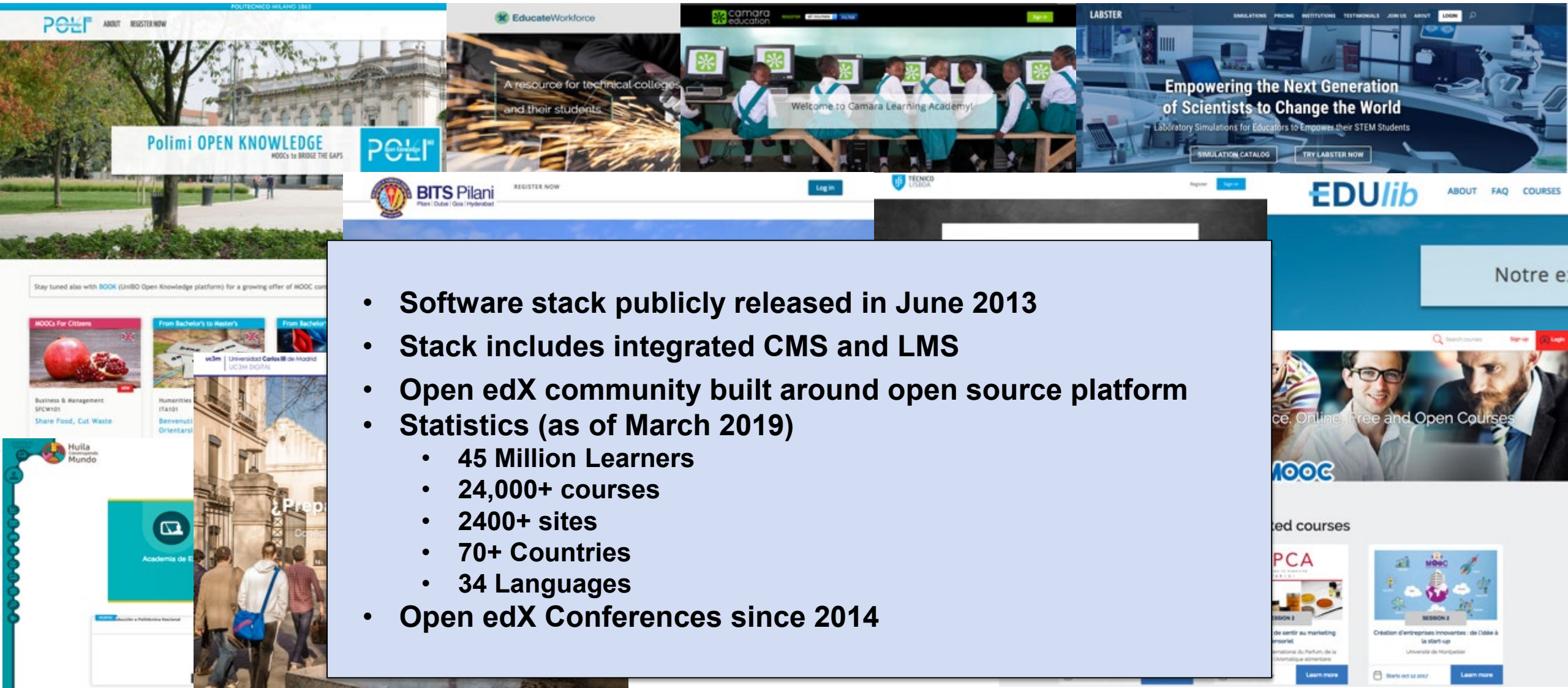


Metrics

- Basic demographics
- Engagement with content
- Exercises and grades
- Surveys & feedback comments
- Learning paths
- Data informs course updates



Open edX Platform



- Software stack publicly released in June 2013
- Stack includes integrated CMS and LMS
- Open edX community built around open source platform
- Statistics (as of March 2019)
 - 45 Million Learners
 - 24,000+ courses
 - 2400+ sites
 - 70+ Countries
 - 34 Languages
- Open edX Conferences since 2014



MOOC Design Considerations

- **Content Selection**
 - Partition material into easily absorbable segments
 - Segments must be self-contained, progression not always linear
 - Content must be clear and simple without unnecessary simplifications
 - Remove all redundant material
- **Delivery**
 - Vary delivery modes used to present the content, e.g. video, text, simulation
 - Select most suitable medium for content
 - Course structure must be transparent and easy to navigate
 - Hands-on exercises reinforce theory
- **Learning experience**
 - Provide optional activities and reference to additional information
 - Enable and encourage interactions between the learners
 - Provide learners with a variety of assessments to test their understanding
 - Little-to-no overhead in setting up hands-on exercises

2:18 YOU'VE COMPLETED 1 STEP IN WEEK 2



[View transcript](#)

[Download video: standard or HD](#)

Having watched the above video, how would you modify it to make it more accurate? **Share your ideas in the comments section!**

The four circles could be grouped together to indicate a blade.

Like 2 Reply Bookmark



David Henty LEAD EDUCATOR

Follow 18 FEB

That's a very good point - on ARCHER the nodes are packaged so that there are four on a physical "blade". This means that these four nodes can actually communicate with each much more quickly than with nodes on a different blade.

Like 1 Reply Bookmark

Supercomputing, FutureLearn (EPCC/PRACE)

Home Course Discussion Wiki Progress Instructor

Bookmarks Search

Use Case 1: Throughput Computing > Submitting and Monitoring Job Arrays > Job Array Submission

Job Array Submission [VIEW UNIT IN STUDIO](#)

Example Job Array Submission

```

[studentx@login-1-1 ~]$ cd examples/
[studentx@login-1-1 examples]$ ls
JobArrays LLGrid_MapReduce pMatlab_examples serial_test_code
[studentx@login-1-1 examples]$ cd JobArrays/cplusplus_code/
[studentx@login-1-1 cplusplus_code]$ ls
fib_batch inputFile_10 inputFile_200 submit_fib.sh
fib_batch.cpp inputFile_inputFile_100 submit_fib0_tasks.sh
[studentx@login-1-1 cplusplus_code]$ less inputFile_10
[studentx@login-1-1 cplusplus_code]$ vi submit_fib.sh
[studentx@login-1-1 cplusplus_code]$ LLGrid_status

LLGrid: tx2500

Online processors: 912
Claimed processors: 264
Claimed processors for exclusive jobs: 0
Active jobs (running/suspended): 36 (36/0)
Pending jobs : 21

-----
Available processors: 648

[studentx@login-1-1 cplusplus_code]$ qsub -t 1-10 ./submit_fib.sh
Your job-array 1814210.1-10:1 ("submit_fib.sh") has been submitted
[studentx@login-1-1 cplusplus_code]$
    
```

0:58 / 2:40 Speed 1.50x

Using MIT Supercloud, LLx (Open edX) (MIT/LLSC)

2:18 YOU'VE COMPLETED 1 STEP IN WEEK 2



[View transcript](#)

[Download video: standard or HD](#)

Having watched the above video, how would you modify it to make it more accurate? **Share your ideas in the comments section!**

The four circles could be grouped together to indicate a blade.

Like 2 Reply Bookmark



David Henty LEAD EDUCATOR

Follow 18 FEB

That's a very good point - on ARCHER the nodes are packaged so that there are four on a physical "blade". This means that these four nodes can actually communicate with each much more quickly than with nodes on a different blade.

Like 1 Reply Bookmark

Supercomputing, FutureLearn (EPCC/PRACE)

Home Course Discussion Wiki Progress Instructor

Bookmarks Search

Use Case 1: Throughput Computing > Submitting and Monitoring Job Arrays > Job Array Submission

Previous [Video Icon] [Document Icon] [Print Icon] [Share Icon] [Download Icon] Next

Job Array Submission

[Bookmark this page](#) [VIEW UNIT IN STUDIO](#)

Example Job Array Submission

```

[studentx@login-1-1 ~]$ cd examples/
[studentx@login-1-1 examples]$ ls
JobArrays LLGrid_MapReduce pMatlab_examples serial_test_code
[studentx@login-1-1 examples]$ cd JobArrays/cplusplus_code/
[studentx@login-1-1 cplusplus_code]$ ls
fib_batch input2 inputFile_10 inputFile_200 submit_fib.sh
fib_batch.cpp inputFile inputFile_100 submit_fib0_tasks.sh
[studentx@login-1-1 cplusplus_code]$ less inputFile_10
[studentx@login-1-1 cplusplus_code]$ vi submit_fib.sh
[studentx@login-1-1 cplusplus_code]$ LLGrid_status

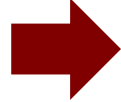
LLGrid: tx2500

Online processors: 912
Claimed processors: 264
Claimed processors for exclusive jobs: 0
Active jobs (running/suspended): 36 (36/0)
Pending jobs : 21
-----
Available processors: 648

[studentx@login-1-1 cplusplus_code]$ qsub -t 1-10 ./submit_fib.sh
Your job-array 1814210.1-10:1 ("submit_fib.sh") has been submitted
[studentx@login-1-1 cplusplus_code]$
    
```

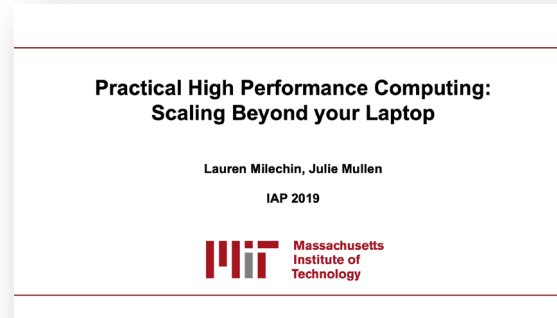
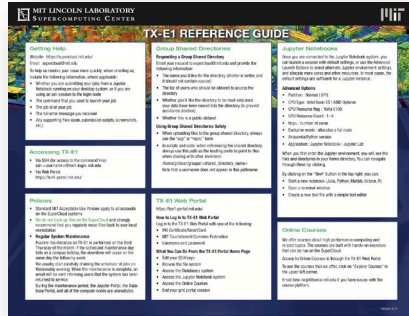
0:58 / 2:40 Speed 1.50x

Using MIT Supercloud, LLx (Open edX) (MIT/LLSC)

- **Challenges in HPC Education and Training**
- **Introduction to MOOCS**
-  • **Case Study: Understanding HPC Workflows and How to Exploit Them**
 - **The initial one-on-one training**
 - **The first online course**
 - **The future refactored courses**
- **Lessons Learned**



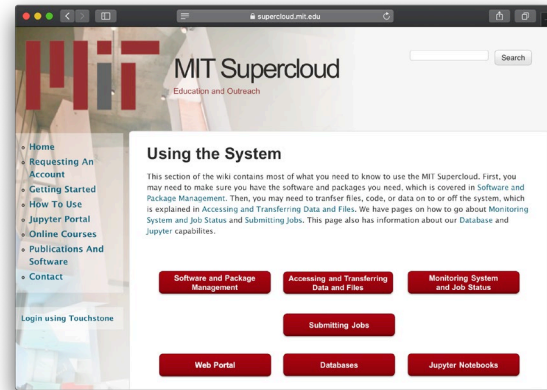
Pre-MOOC Training



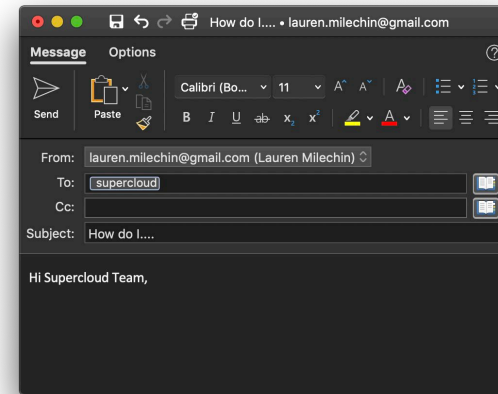
Reference Guides

IAP/Special Workshops

One-on-one Tutorials



Web Pages



Email



Scaling Step 1: Small Private Online Course (SPOC)

Goal: Develop a course for professionals and researchers that teaches strategies for building HPC workflows

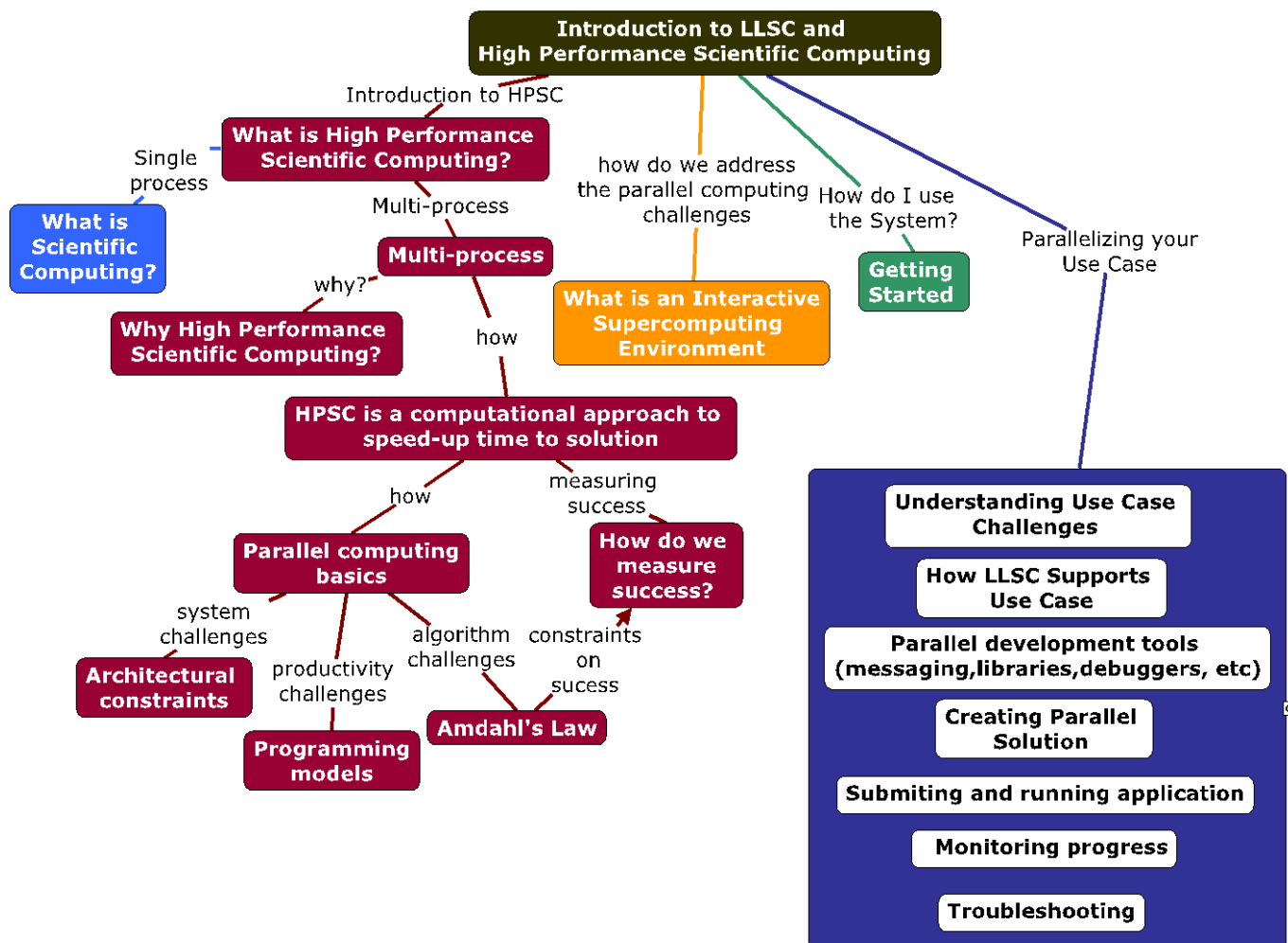
Audience

- Professional engineers and scientists who need to scale scientific workflows
- Diverse Learner background
 - Range of domains
 - Range of problem types
 - Range of computer literacy

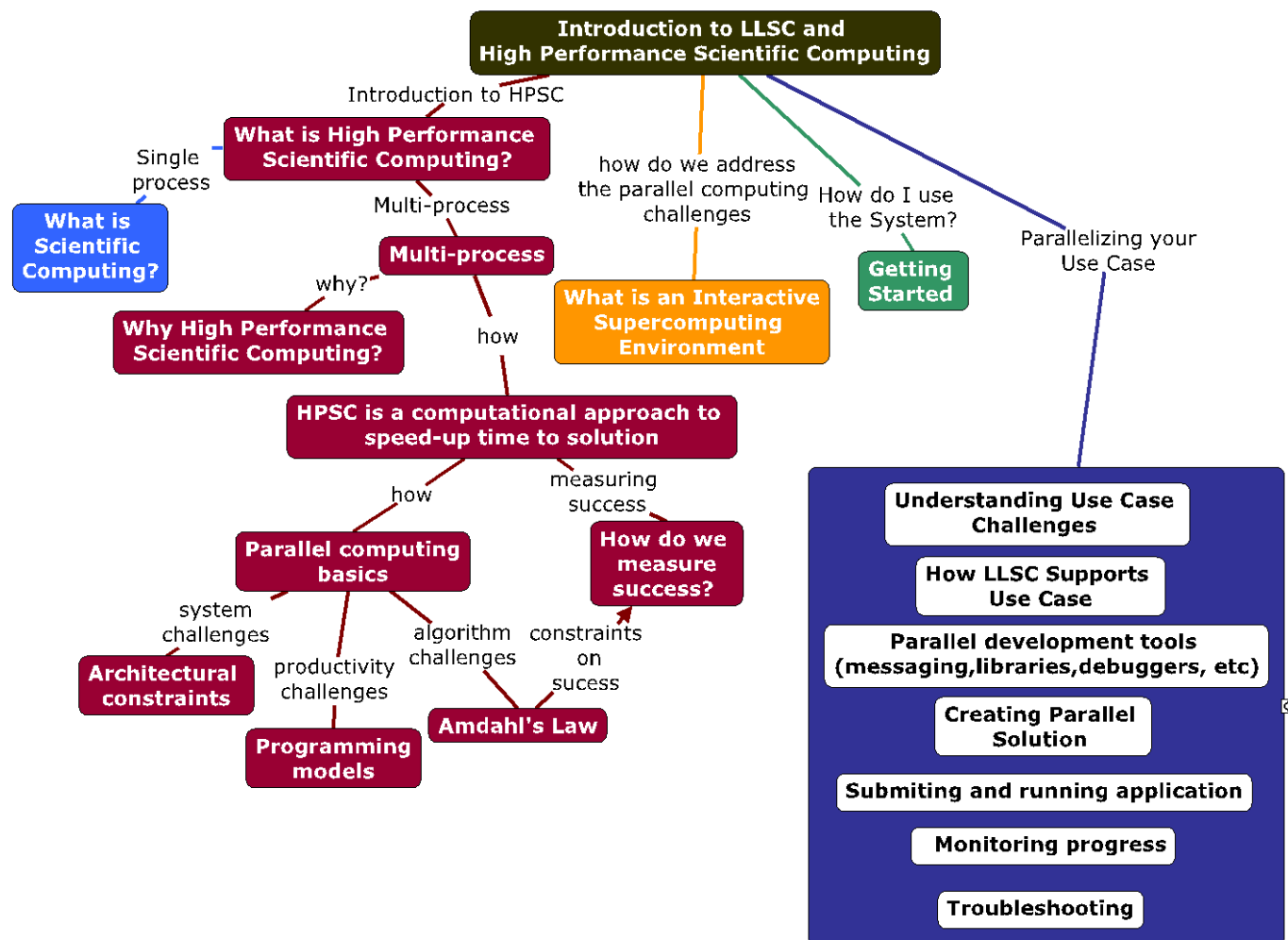
Delivery

- A mixture of videos, text, programming exercises and quizzes
- Focus on learning through theory and practice
- Hands-on practice using HPC system
- Self-paced

Design of Open Online Course

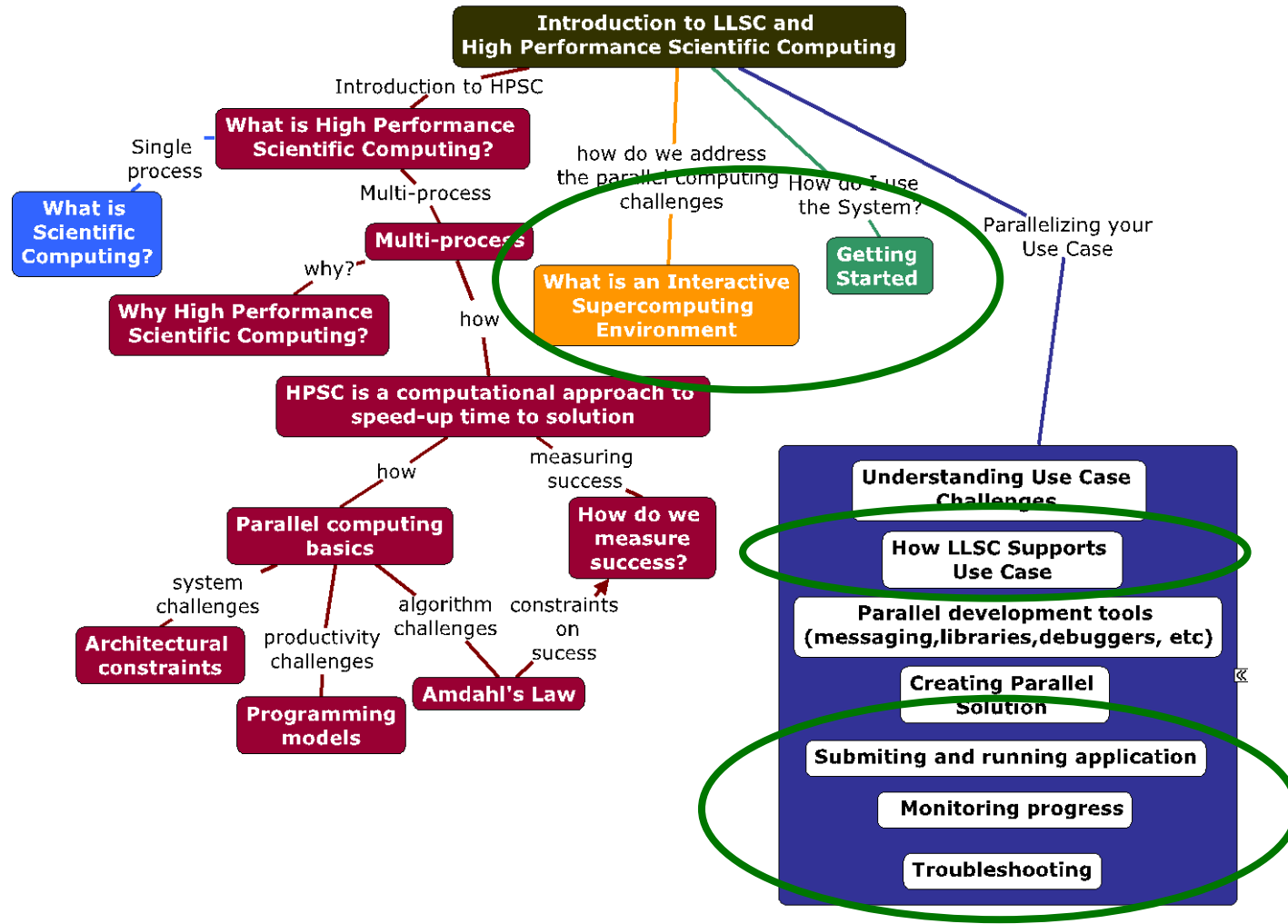


- **Andragogical Principles**
 - Interleave Theory and Practice
 - Present content in self-contained chunks
 - Highlight links between concepts
 - Path enables learners to “build their own adventure”
- **Concept Map Design Tool**
 - Leaf nodes form content units
 - Design exposes course components
 - Related units form sections
 - Related sections form modules
 - Links are bi-directional between related concepts
 - Concepts with no links are removed



- **Design supports learning paths**
 - **Learners can select content sections relevant to their immediate application**
 - **Increased likelihood of on the job practice leads to increased retention**
 - **Supports adult need to match learning to problem or question**

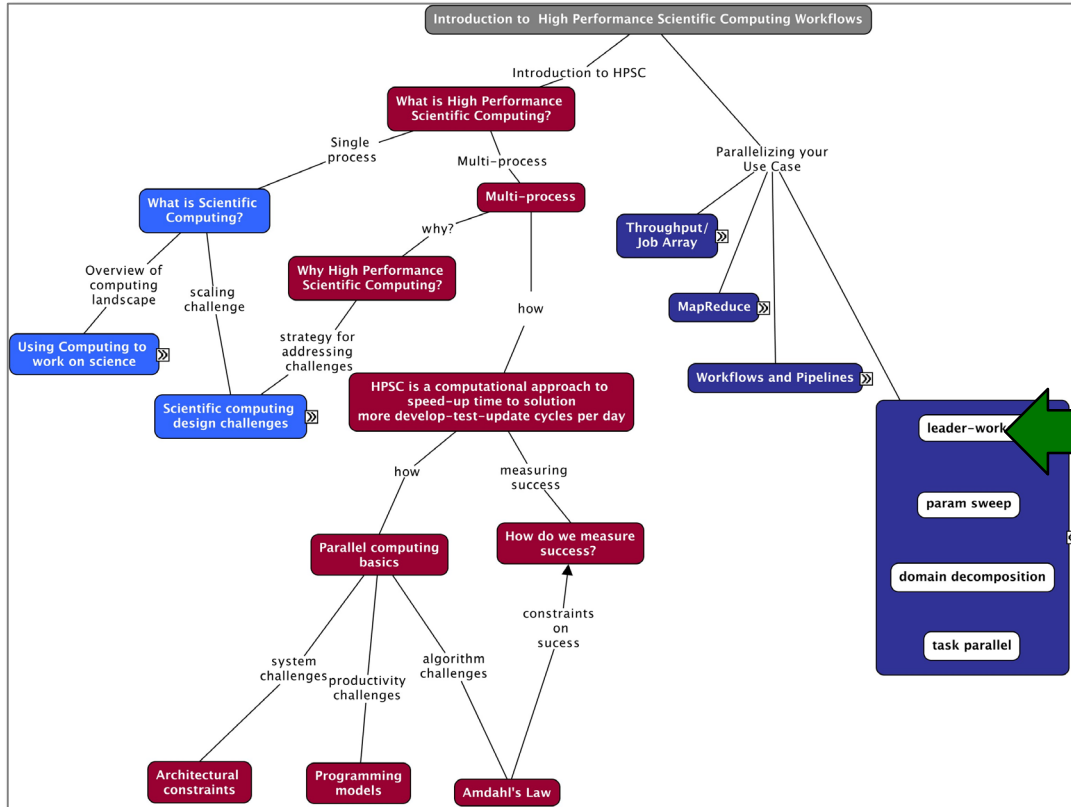
- **Interleave Theory and Practice**
 - **Learners gain experience with their application on target system**
 - **Immediate feedback to assessments minimizes misconceptions**



- Design supports learning paths
 - Learners can select content sections relevant to their immediate application
 - Increased likelihood of on the job practice leads to increased retention
 - Supports adult need to match learning to problem or question
- Interleave Theory and Practice
 - Learners gain experience with their application on target system
 - Immediate feedback to assessments minimizes misconceptions
- Not reproducible
 - Highlighted areas contain material specific to one supercomputing center

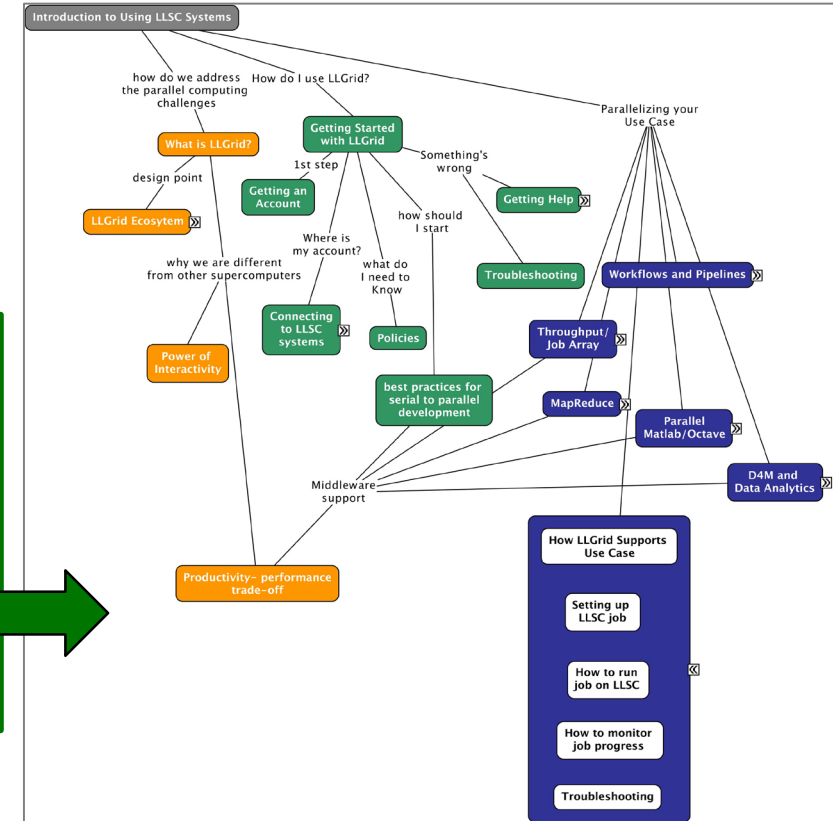


Scaling Step 2: Redesign as a MOOC and SPOC



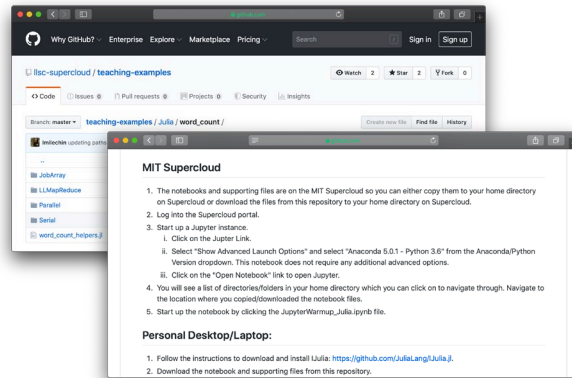
Introduction to High Performance Scientific Workflows (MOOC)

- Split course into 2 short courses
 - Introduction to HPC Strategies
 - Using the MIT Supercloud
- Convert programming components to demos and thought exercises
- Maintain hands-on component for center members
- SPOC designed to be easy to edit and adapt as center evolves and changes

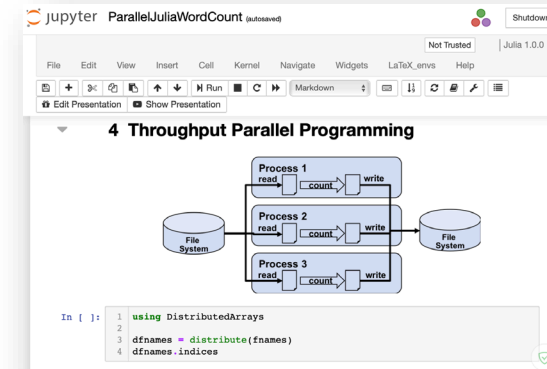


Using the MIT Supercloud System (SPOC)

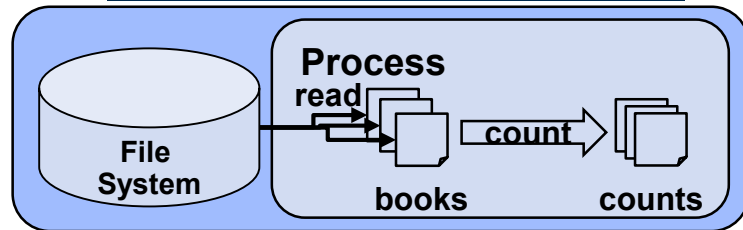
Github Repository



Jupyter Interface



Thought Problems



Questions to Consider:

- Where is the independence?
- What data access patterns do you expect?

Demonstrations

```
Lauren — StudentX@login-1:~ — ssh studentx@txe1-login.mit.edu — 80x24
[StudentX@login-1 ~]$ module load julia-1.0
[StudentX@login-1 ~]$ julia

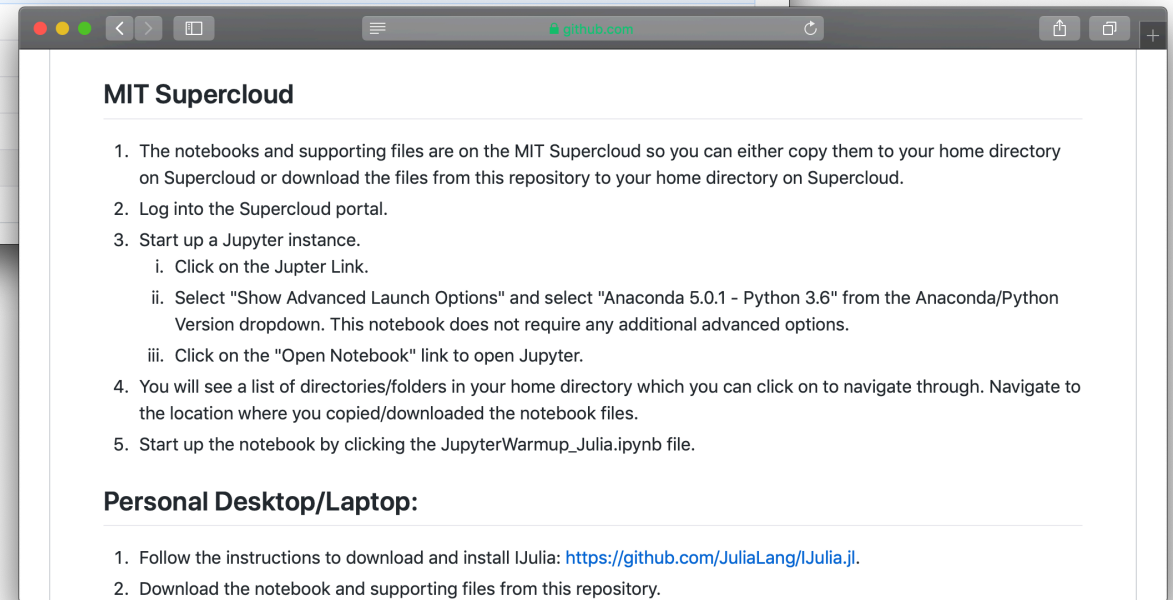
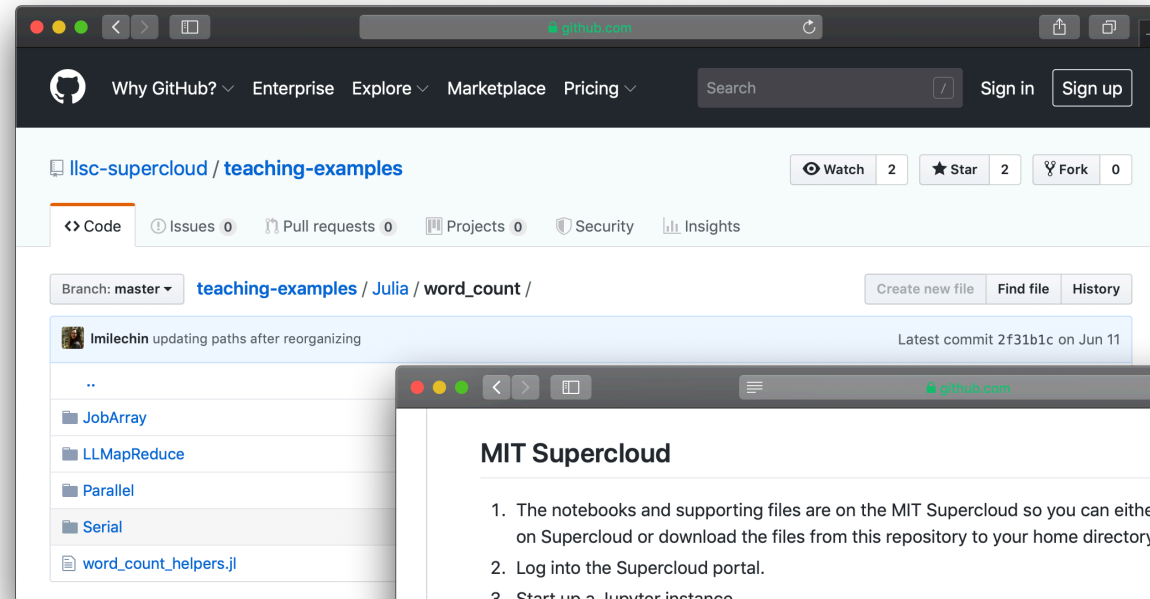
Documentation: https://docs.julialang.org
Type "?" for help, "]"? for Pkg help.
Version 1.0.0 (2018-08-08)
Official https://julialang.org/ release

[julia> using Pkg

[julia> Pkg.status()
Status `~/state/partition1/llgrid/pkg/julia-1.0.0/local/share/julia/enviro
nments/v1.0/Project.toml`
 [621f4979] AbstractFFTs v0.3.2
 [1520ce14] AbstractTrees v0.2.1
 [79e6a3ab] Adapt v0.4.2
 [6e4b80f9] BenchmarkTools v0.4.2
 [9e28174c] BinDeps v0.8.10
 [b99e7846] BinaryProvider v0.5.3
 [a74b3585] Blosc v0.5.1
 [e1450e63] BufferedStreams v1.0.0
```

Github Repository of Examples

- Hands-on examples provided in Github repository
- Instructions for setting up and running problems
 - Locally
 - On Supercloud
- Contain or point to sample data where used





<https://github.com/llsc-supercloud/teaching-examples>

jupyter ParallelJuliaWordCount (autosaved) Shutdown

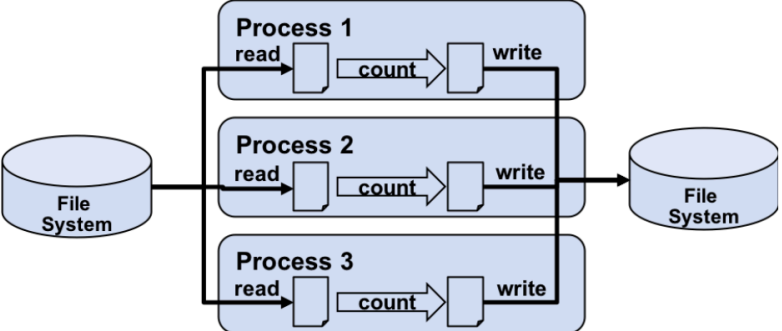
Not Trusted | Julia 1.0.0

File Edit View Insert Cell Kernel Navigate Widgets LaTeX_envs Help





4 Throughput Parallel Programming

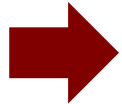


```
In [ ]:
1 using DistributedArrays
2
3 dfnames = distribute(fnames)
4 dfnames.indices
```

- **In-browser programming environment**
 - Notebooks contain images, plots, text, executable code blocks
 - Text editor with syntax highlighting
 - Command line
- **Support variety of programming languages**
 - Notebooks: Scripting languages
 - Text editor/terminal: Scripting and Compiled languages
- **Provide easy to use alternative to command line**

Outline

- **Challenges in HPC Education and Training**
- **Introduction to MOOCS**
- **Case Study: Understanding HPC Workflows and How to Exploit Them**
 - The initial one-on-one training
 - The first online course
 - The future refactored courses
- **Lessons Learned**





Lessons Learned

- **HPC Education and Training can benefit by leveraging MOOCs**
 - Can reach thousands of students
 - Students can
 - Self-select to create their own “Just In Time” experience
 - Learn at their own pace
 - Revisit material for review or deeper understanding
 - Online targeted lectures simplify learning and review for native and non-native speakers
 - Can track student activity to
 - Capture learning interests
 - Content gaps
- **MOOCs aren't a drop in replacement for existing training materials**
 - Materials need to be refactored with design emphasis on creating stand alone content
 - Diversity of student skills and experience requires inclusion of supporting material
 - Instructor led courses require facilitation
 - Difficult to provide HPC System access to thousands of students for hands-on practice



Acknowledgements

- **MIT Supercloud Team**
- **MIT Lincoln Laboratory Supercomputing Center**
- **Weronika Filinger, EPCC/PRACE**



Backup/Notes
