# The impact of MOOC methodology on the scalability, accessibility and development of HPC education and training

**Julia Mullen, Weronika Filinger, Lauren Milechin and David Henty**

**November 11, 2018**

# Outline

- Challenges in HPC Education and Training
- Introduction to MOOCS
  - MOOC overview (scaling, andragogy, metrics)
  - Common Best Practices for online course design
  - MOOC examples (FutureLearn screen shot, edX screenshot)
- Case Study 1: Supercomputing
- Case Study 2: Understanding HPC Workflows and How to Exploit Them
- Lessons Learned

# MOOC Overview

## Scaling

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

## Pedagogy/Andragogy

- Open
  - No pre-requites
  - Range of experience
- Online learning
  - Asynchronous
  - Self-paced
  - Instructor paced
- Social learning – massive, diverse students interacting
- Built to support practice and theory

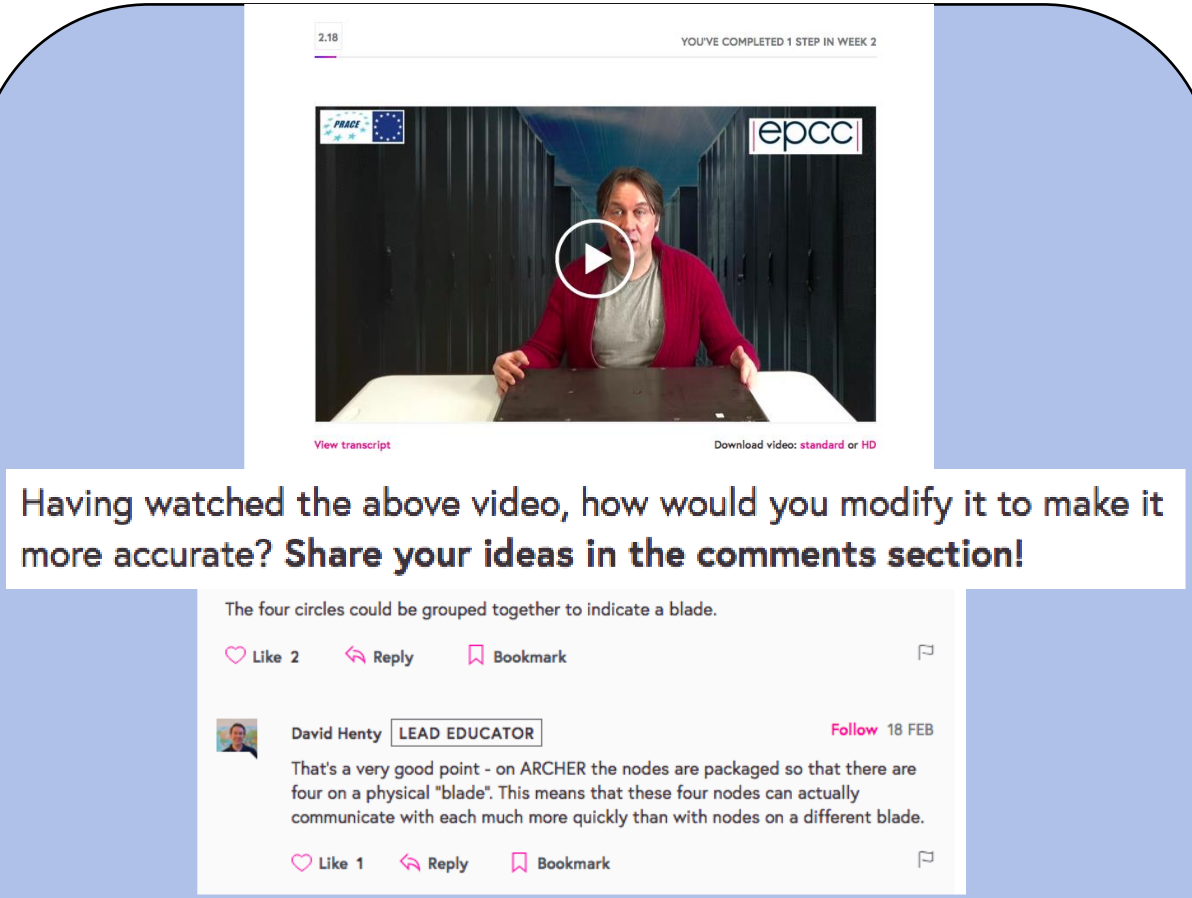
## Metrics

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

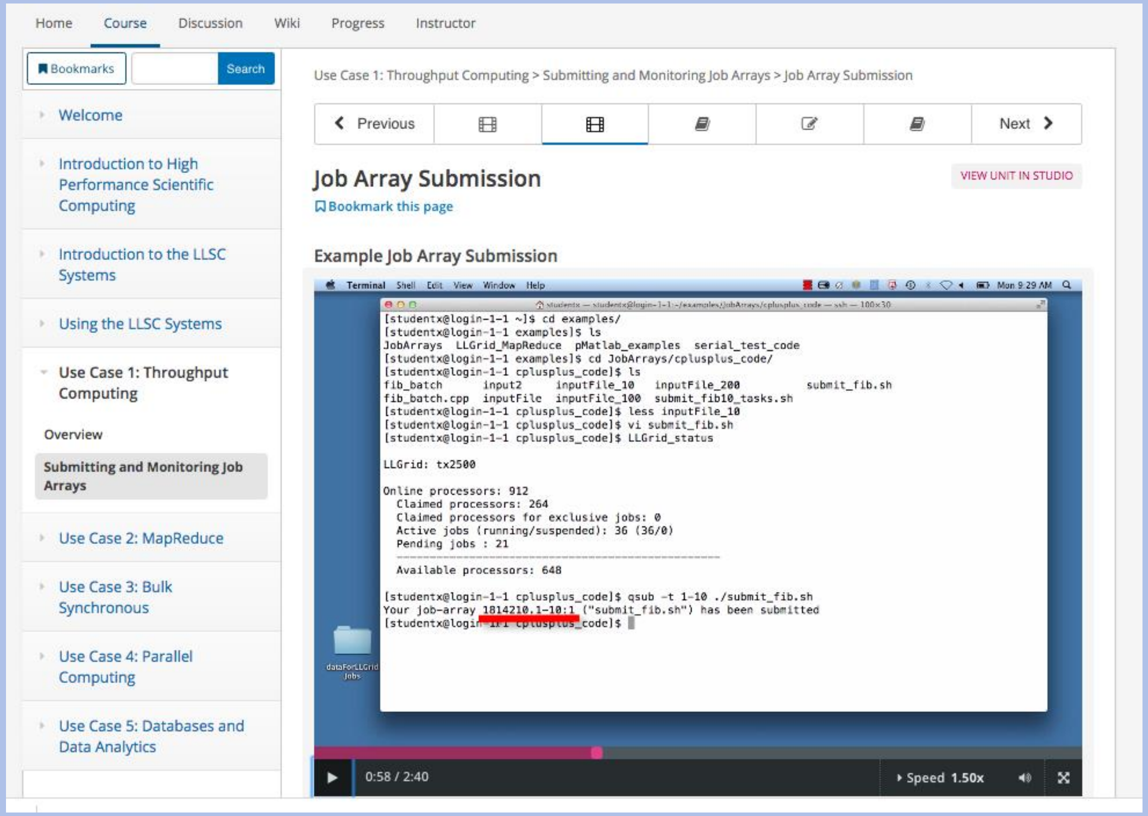
# MOOC Design Considerations

- 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
- Learning experience
  - Provide optional activities or information where to find more information
  - Enable and encourage interactions between the learners
  - Provide a variety of assessments for learners to test their understanding

# MOOC Examples




**FutureLearn: "telling stories, provoking conversation and celebrating progress".**




**BWedX (Open edX): Merging Theory and Practice**



Massachusetts Institute of Technology

THE UNIVERSITY of EDINBURGH

epcc

PRACE

# Case Study 1 – Supercomputing on FutureLearn

## Goal: create a general Introduction for Supercomputing

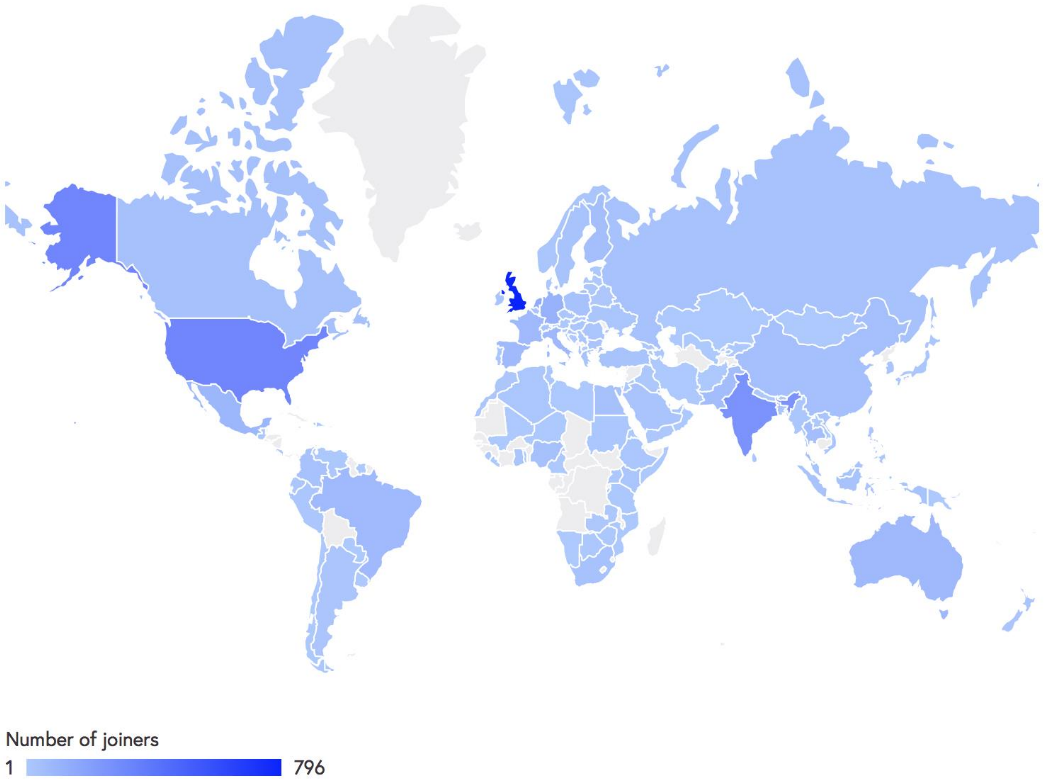
### Audience

- Anyone interested in Supercomputing
  - general public
  - entry level HPC practitioners
- Diverse Student background
  - Range of ages
  - Range of computer literacy
  - Multi-cultural, multi-lingual

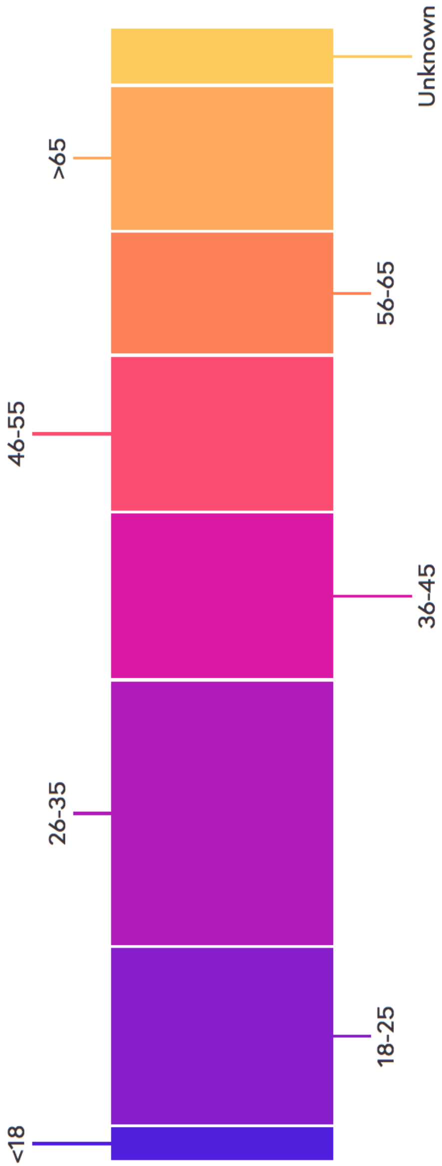
### Delivery

- A mixture of videos, articles, discussions, exercises, quizzes and tests (increase engagement)
- Focus on discussions (social learning)
- Learners primarily interact with each other
- Educator interaction important to sustain and steer discussions

# Supercomputing Stats




- 4 runs in 2 years
- Over 5400 learners
- 137 countries
- About 20% learners completed >50% of the course
- About 9% completed >90% of the course

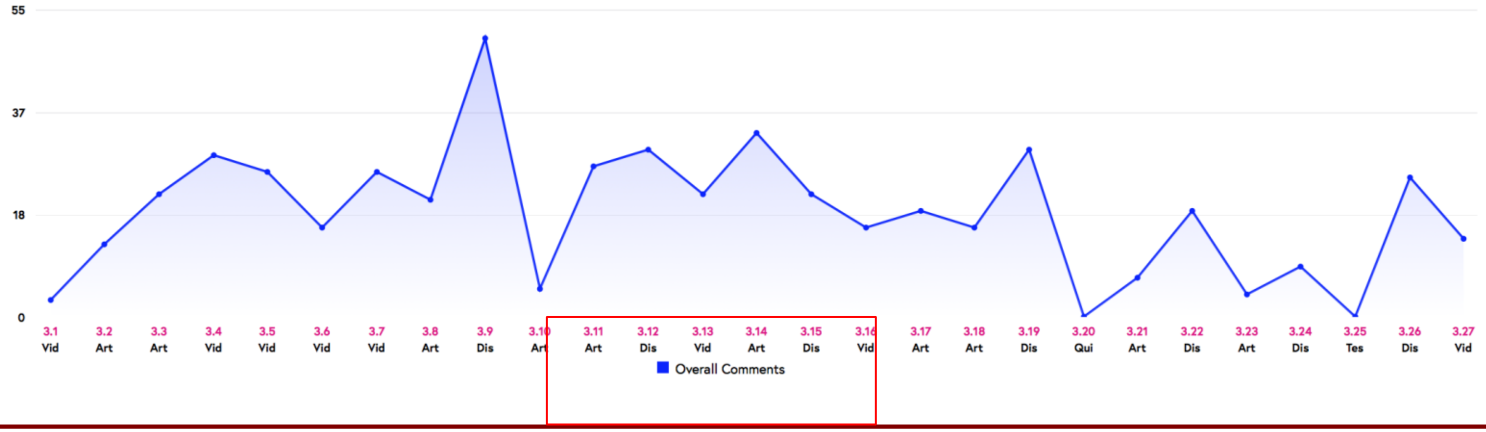




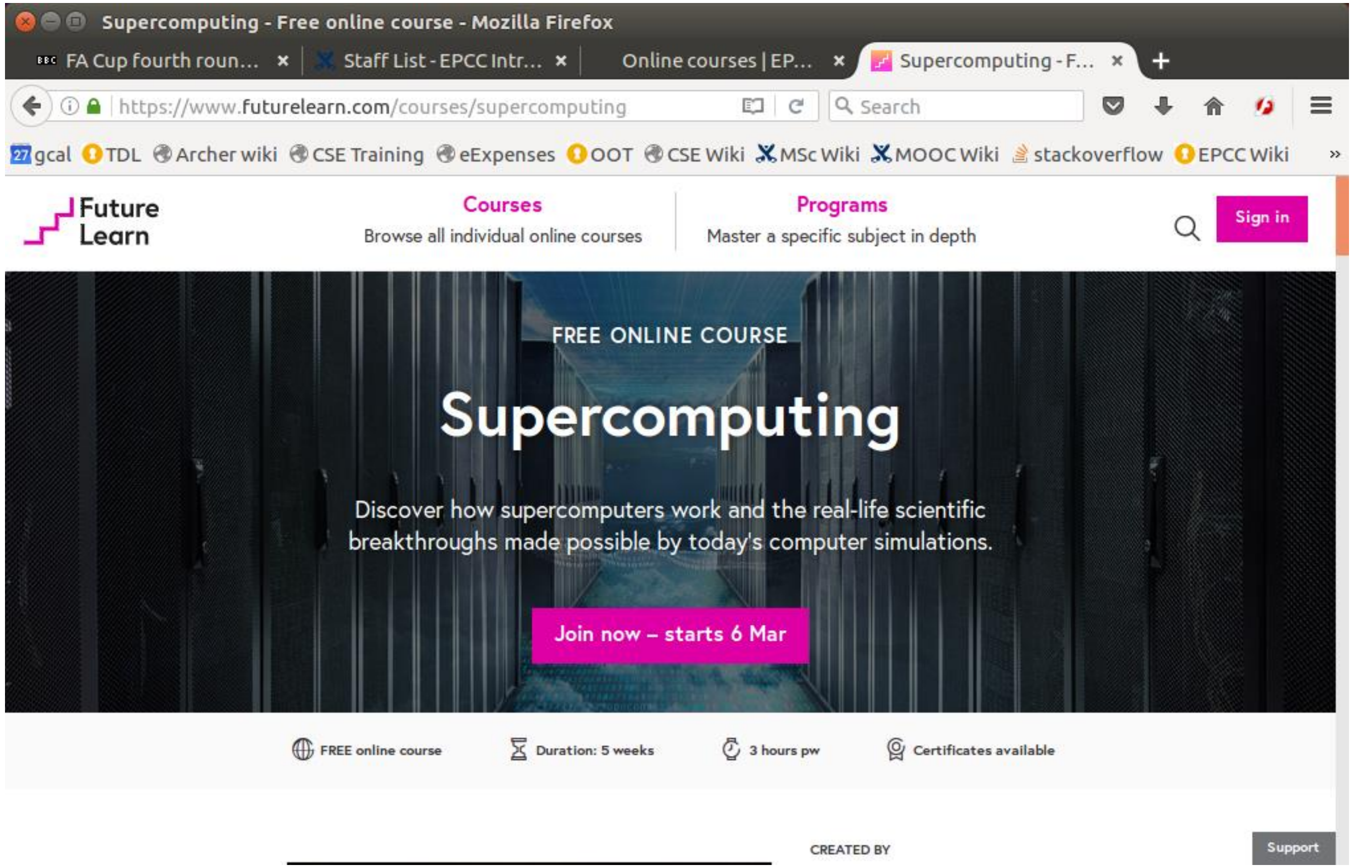

Massachusetts Institute of Technology
THE UNIVERSITY of EDINBURGH
epcc
PRACE

# Supercomputing - Lessons Learned

- **Demystifying ‘what supercomputing is not’ turned out to be a big part of the course**
- **The combination of different step types worked well – not too many videos!**
- **Content refactoring and making it presentable took a significant amount of effort**
- **Each activity should encourage discussion**
- **The course doesn’t quite run itself – input from educators is important**
- **Creating a programming based MOOC would be very challenging**



# futurelearn.com/courses/supercomputing



# Short promotional video


Download video: standard or HD

**CREATED BY**

PRACE

**SHARE**

f G+ in

**ABOUT THE COURSE**

Today's supercomputers are the most powerful calculating machines ever invented, capable of performing more than a thousand million million calculations every second. This gives scientists and engineers a powerful new tool to study the natural world – computer simulation.

**START DATES**

6 Mar 2017 — Join
UPCOMING

TBA — Register interest
DATE TO BE ANNOUNCED

Support

# General outline

Using supercomputers, we can now conduct virtual experiments that are impossible in the real world – from looking deep inside individual atoms, to studying the future climate of the earth and following the evolution of the entire universe from the big bang.

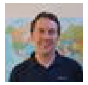
## Discover how supercomputers are powering scientific breakthroughs

This free online course will introduce you to what supercomputers are, how they are used and how we can exploit their full computational potential to make scientific breakthroughs.

Over five weeks, we'll look at:

- **supercomputers**: introducing supercomputing terminology and some of the largest machines in the world.
- **parallel computers**: how they are built from hundreds of thousands of CPUs, each similar to those in a desktop PC.
- **parallel computing**: using parallel processing to harness the power of all of those CPUs for a single calculation.
- **computer simulation**: how we can perform virtual experiments to make real-life predictions.
- **case studies**: how supercomputing is making scientific breakthroughs that were never possible before.

EDUCATORS



David Henty

IN ASSOCIATION WITH

THE UNIVERSITY of EDINBURGH

SURF SARA

# Typical activities and steps in week 2

## Connecting multiple computers

Here we explain how large supercomputers containing many thousands of CPU-cores are constructed from commodity building blocks.

- 2.6 DISTRIBUTED MEMORY ARCHITECTURE ARTICLE
- 2.7 SIMPLE PARALLEL CALCULATION DISCUSSION
- 2.8 CASE STUDY OF A REAL MACHINE ARTICLE
- 2.9 UNDERSTANDING PARALLEL COMPUTERS QUIZ

## Comparing the two approaches

The shared and distributed memory architectures are very different: each has its own pros and cons, with performance governed by different factors.

Support

# Typical article around 750 words max


© iStock.com/BrianAJackson

## Shared memory v.s. Distributed memory

**We've seen how individual CPU-cores can be put together to from large parallel machines in two fundamentally different ways: the shared and distributed memory architectures.**

In the shared-memory architecture all the CPU-cores can access the same memory, much like several workers in an office sharing the same whiteboard, and are all controlled by a single operating system. Modern processors are all multicore processors, with many CPU-cores manufactured together on the same physical silicon chip.

There are, however, limitations to the shared-memory approach due to all the CPU-cores competing for access to memory over a shared bus, much like the obvious issues in trying to cram too many workers into the same office. This can be alleviated to some extent by introducing memory caches or putting several processors together in a NUMA architecture, but there is no way we can reach the hundreds of thousands of CPU-cores we need for today's multi-petaflop supercomputers.

© iStock.com/Maxiphoto

## Simple Parallel Calculation

**Let's return to the income calculation example. This time we'll be a bit more ambitious and try and add up 8000 salaries rather than 800. This list of salaries fills 10 whiteboards (800 on each) all in separate offices.**

If we have one worker per office, think about how you could get them all to cooperate to add up all the salaries. Consider two cases:

- only one "boss" worker needs to know the final result;

- all the workers need to know the final result.

# Simple “blue-screen” videos