https://doi.org/10.36315/2022v2end062
Education and New Developments 2022

EDUCATING FOR MODERN CLOUD TECHNOLOGIES IN A PLATFORM-AGNOSTIC FASHION

David Cutting, Andrew McDowell, Esha Barlaskar, Neil Anderson, Moira Watson, & Matthew Collins
School of Electronics, Electrical Engineering and Computer Science,
Queens University Belfast (United Kingdom)

Abstract

It is essential to provide computing students with hands-on exposure to modern techniques and technologies such as cloud computing. To this end we engaged with industry to co-design a curriculum for modern cloud computing to meet the industry needs for graduates. Vendors do provide access to cloud computing but their academic programs often leave much to be desired, with limited credits and features or a requirement to buy-in to their platform entirely. In the resource-constrained university sector simply buying vendor credits for students would be expensive but also wasteful as there is no capital investment. Having engaged with industrial partners to identify core theoretical skills and a subset of topics in which specific platform experience would be desirable we co-designed learning outcomes and assessments. To provide financially efficient platform-agnostic experience a private cloud was built using open-source tools which allowed a new large final year module. Having run this module for two full years and had excellent student feedback we reflect on the strengths and weaknesses of this approach and how industrial engagement cannot be a one-off, in a changing field it must be continuously performed to allow the curriculum to be refreshed and support emerging topics.

Keywords: *Cloud computing, co-design, curriculum, curriculum refresh.*

1. Introduction

The area of cloud computing is one that has seen significant growth and use in industry often categorized as “using someone else’s computer” i.e. rather than running code on your own equipment it is deployed to the cloud and runs on a vendors service (such as Amazon Web Services or Google Cloud Platform). This can be seen as a “utility computing” model i.e. you only pay for what you use rather than acquiring and maintaining expensive infrastructure. Use of cloud computing technologies allows businesses a quick route to market, much more efficient use of resources and agility in responding to changes in demand, making it a key factor in the growth of many organisations (DeStefano et al., 2020). Consequently, skills in cloud computing are much in demand by employers (Haranas, 2021) and so an area where Higher Education may choose to focus as industry alignment (Plice and Reinig, 2007; Benamati et al., 2010) is an important factor in delivering the workplace-ready graduates for the computing sector.

For us to deliver education in this area a curriculum review needed to be undertaken and new content devised. Given the drivers behind this requirement and the importance of curriculum alignment (Plice and Reinig, 2007; Benamati et al., 2010) a co-design approach was taken whereby industrial partners had direct input into how the learning outcomes, activities, and assessments would be defined as well as providing expertise and masterclasses as appropriate throughout the academic year. Early on it was also identified that given the domain and the difference in how students learn (Schmeck, 2013) with many favouring “learning by doing” i.e. kinesthetic learning (Ayala et al., 2013) there should be a substantial elements of hands on applied computing included.

In this paper we will focus on the co-design of the curriculum in section 2 and then how we provided access to cloud environments in section 3 and the student experience in section 4. Finally, we will discuss and reflect on how the process has worked after two full years of delivery (section 5) before reaching conclusions and identifying future work in section 6.

2. Co-design of curriculum

Co-design of a curriculum with expert parties is not in itself new and is widely adopted in many domains such as healthcare (Sbaiti et al., 2021) and can be defined as "a highly-facilitated, team-based process in which teachers, researchers, and developers work together in defined roles to design an educational innovation" (Penuel et al., 2007). Having identified both the need for cloud computing skills and a desire for strong industry engagement a series of events were held to co-design a final year computing module covering these topics. Working with 1 existing industrial contacts a series of small meetings were held starting with a blank page and beginning to draft possible topics and assessments. The intention of these, building on existing good relationships, was to define some draft structures which could be used to inform wider discussion, it being easier to get feedback on a plan than start with a blank piece of paper. The output from this stage was taken forward and a draft module outline with several options for assessments were generated by academics.

The process culminated in a large "coffee and cloud" event hosted at the University where a large number of employers in different fields were invited (these ranged from SMEs to multi-nationals) to help shape the curriculum. This event was held under "Chatham House Rules" and generated lively debate and engagement. Split into groups and each with a facilitator from the University a series of guided discussions were undertaken asking open questions around general issues and the proposed module draft in particular. Following each group discussion a wider inter-group discussion was held where the conclusions of each group were read out by the facilitator and other groups invited to comment or challenge.

The headline findings of the "coffee and cloud" event as the final consultation of the co-design were as follows:

- **Generic skills more important than platform experience** - employers would value generic hands on experience with any cloud techniques and technologies, especially for graduate roles, rather than expecting any particular technology or vendor although some experience of different vendors would be useful to see differences in offerings.
- **Distributed computing theory is very relevant** - focusing on the platforms and technologies must not be at the expense of key theoretical concepts. The challenges of distributed computing (Liu et al., 2012) are ever present in large cloud systems and should be thoroughly understood.
- **Processes and ops skills are important** - to fully utilise cloud computing efficiently understanding and exposure to processes such as continuous integration (CI) and operation skills such as shell scripting are needed.
- **End to end engineering** - students should be expected to understand the process of engineering a cloud system from end to end, including architecting a modern system and a good understanding of relevant software engineering principles to make effective use of technologies such as microservices.
- **Metrics and monitoring** - students should be aware of how and why services are monitored, how reliability can be measured and improved, and the type of modern metrics gathered from distributed cloud systems.

Ultimately two things occurred; a drive to implement certain concepts important to "cloud engineering" throughout the curriculum at different stages and a specialist new module was implemented with topics and learning outcomes based on the co-design output. Assessments for this module were finalised following discussion at "coffee and cloud" and then distributed for any further comment to the participants resulting in only positive comments.

3. Access to cloud environments

Recognising it is essential to provide hands on applied skills in this area it was necessary to consider how provision to cloud environments and infrastructure could be facilitated. While most cloud vendors do offer some form of educational grant or academic access we found this far from plain sailing, with students being rejected from academic programmes despite providing ID, or just running out of provided credit before completing their work. Of course it would be possible to ourselves utilise "utility computing" and simply pay for access to vendor systems but this was not a serious option in the resource constrained world of UK higher education, something some providers did not seem to understand (being highly US-centric in their education programs where financial resources are more readily available). Another restriction on buying service was the transient nature of the spend, at the end of the year the students would move on but no long-term investment would have been made in the university facilities and would require ongoing financial commitment year-on-year.

However, many of the technologies used are actually freely available, mostly based on free open-source software. The vendors may charge for use of their computing resources and storage but in many cases the actual technologies and stacks can be implemented on hardware by anyone. A plan was therefore formed to build a private internal cloud (called our “cloud in a cupboard”), a state-of-the-art cloud computing lab facilitating hands on learning but using resources which represent a long-term capital investment rather than an ongoing utility cost.

3.1. Solution design of the private cloud

To facilitate the delivery of a modern cloud syllabus the following functional requirements were defined for the private cloud:

- **Latest industry practice** – the latest industry practice in cloud deployment should be implemented as far as possible.
- **Open access architecture** – students should be free to interact with the resources directly in user interfaces or through APIs to allow open expansion and access from third-party tools such as Terraform.
- **Shell access** – students must be provided remote Unix shell access to allow for command-line use and execution of tools within the environment.
- **Pipelines** – students must have access to pipeline systems to facilitate CI.

Additionally, some non-functional requirements were defined representing the constraints within which any system must operate:

- **Secure** – the system should provide a minimal security risk to the internal network.
- **Immediate creation pending further investment** – given the funding and budgetary cycles of the university the system must be able to deliver a minimum viable product using only reused/repurposed hardware in the interim.
- **Robust** – as a critical component to a module the system must be robust enough to handle any minor issues without compromising performance.
- **Scalable** – the system must have the capacity to scale in future should this be required.
- **Evolvable** – the system should have the general design characteristics to allow the evolution of specific technology platforms deployed upon it.

The private cloud was then implemented during summer 2019 using repurposed hardware as an initial proof of concept before being scaled up with dedicated equipment procured in 2020 and 2021.

4. Student experience

The first student cohort took the module in 2019/20 and it has been in consistent high demand ever since, such that in 2020/21 it was run twice to satisfy all requests while still procuring hardware to scale. As of 2020/21 it is the most chosen optional module in our computing pathways and scores consistently highly in student satisfaction. Student comments at the end of the module have included the following:

- “Very relevant for industry.”
- “It is relevant to what I was doing on Placement, it’s getting exposure to modern technologies and it’s interesting.”
- “Teaching style, the practicals were very useful for the project, industry relevant and up to date.”
- “Variation in type of assessment allowed to use different skills.”

These comments, and scores, reflect positively on the impact the co-design process and open architecture provided have on student experience, allowing what are clearly relevant real-world skills to be taught and assessed as part of the HE curriculum. Returning graduates who are now working in industry have commented that this was “the most relevant module” and “should be essential for all future developers”.

5. Discussion and reflection

While the need to include cloud concepts, engineering and hands on experience was clear from industry input how we should approach this was initially uncertain. Our decision to choose co-design, involving industry right from the start in shaping the module rather than asking for an opinion post hoc, was novel in our school and required significant up-front effort to make contacts and arrange meetings or events. Once the group was setup however the input proved invaluable helping to remove some assumptions (for example we had assumed specific vendor experience would be an essential component while the industry input was the opposite) and build learning outcomes and assessments from the ground up based on real world projects, experience, and needs. This alignment was clearly felt and acknowledged by students, many of whom would have just returned from an industrial placement, giving them a very valuable educational experience.

The financial model, seeking capital investment to procure hardware to build a private cloud rather than paying for utility computing, was made through a business case which included alternative costing models. The expected take-up of the course showed that a hardware investment model would pay for itself within three years (i.e. by year three the cost of hardware would be less than the cost of utility cloud computing for the same period) and this was then supported by our faculty. In fact the demand for the module was higher than expected and the point of return on investment was within two years not three. The ability to use re-purposed hardware initially also meant that we could show such a system was possible and allow staff and administrators to upskill to support the private cloud before any significant investment was made. The ongoing success of the module, both in terms of numbers and student satisfaction, has allowed further business cases to be made to expand the hardware capabilities. This approach did however require significant amounts of staff time, support, and re-skilling which purchasing utility credits from a major vendor (or even using a vendor’s pre-packaged courses such as AWS Academy) would not. The emphasis of maintaining, updating, and managing the hardware and service also falls within the school which provides further risk and challenges that a utility model would avoid.

Although specific platform experience was identified as not needed it was felt some exposure to different vendors was important more to see the differences in offerings rather than any specific technology. To this end the academic programmes of major vendors were used to provide access but as this wasn’t essential to the assessments there was no risk with credit use or access changes, offering a best-of-both approach.

One other aspect of industry engagement was the use of guest speakers. Taking offers to talk gathered during the co-design events a range of speakers were invited in on weeks aligned to the topic they would be covering, adding extra depth and detail to the academic and practical content of that topic. Speakers in this context had a mixed reception with some students clearly preferring talks which were directly related to coursework rather than the wider context of the subject. This feedback, combined with the strictures of COVID-19 remote learning, meant that for 2020/21 guest “talks” actually were in the form of pre-recorded discussions between academic staff and industry experts which could then be optionally viewed in full by the students or used in part (for example a specific question and answer) as video content in a particular topic. This approach was very well received and will continue post COVID.

6. Conclusion and future work

In the time since the curriculum review and the new module it is clear that both have been positively received by both students and industry. The co-design process was an excellent way to engage more effectively with industrial partners while maintaining academic control and quality assurance, and a process we hope to repeat in other topic areas for the future.

Like any modern technology cloud computing is constantly evolving and this co-design and implementation cannot be seen as a one-off activity. Ongoing industrial engagement and regular (we are targeting three year cycles) root-to-branch reviews are important taking a frank and honest look at content to see if it is still relevant as well as identifying any new emerging topics which are now key. Making constant small changes combined with this review cycle should mean the content remains fresh and relevant and responses to emerging technologies can be agile, something it is perceived the University sector has failed to be in the past.

Our next steps with the module, beyond reviews mentioned above, is to look how we could further create industrially-aligned projects seeking to embed industrial partners in the setting and mentoring of project delivery. We will also continue to expand the private cloud capabilities adding both capacity and new features.

Acknowledgements

The authors wish to acknowledge the help and invaluable support of Neil Lowry, Keith Stewart, Debbie Britton, Sean McKeever, Maire Bowler, Martin Kinkead and Michael Garland in the ongoing work around cloud computing in the institution.

References

John H Benamati, Zafer D Ozdemir, and H Jeff Smith. 2010. Aligning undergraduate IS curricula with industry needs. Commun. ACM 53, 3 (2010), 152–156.

M Liu, A Tannenbaum, and M Van Steen. 2012. Distributed Systems: Concepts & Design. (2012).

Mariam Sbaiti, Mike J Streule, Mervat Alhaffar, Victoria Pilkington, Melanie Leis, Shyam Sundar Budhathoki, Hala Mkhallalati, Maryam Omar, Lillian Liu, Amelia Kataria Golestaneh, et al. 2021. Whose voices should shape global health education? Curriculum codesign and codelivery by people with direct expertise and lived experience. BMJ global health 6, 9 (2021), e006262.

Mark Haranas. 2021. The Most In-Demand Cloud Computing Jobs For 2021.

N´estor Adri´an Rodr´ıguez Ayala, Eduardo Gonz´alez Mend´ıvil, Patricia Salinas, and Horacio Rios. 2013. Kinesthetic learning applied to mathematics using kinect. Procedia Computer Science 25 (2013), 131–135.

Robert K Plice and Bruce A Reinig. 2007. Aligning the information systems curriculum with the needs of industry and graduates. Journal of Computer Information Systems 48, 1 (2007), 22–30.

Ronald R Schmeck. 2013. Learning strategies and learning styles. Springer Science & Business Media.

Timothy DeStefano, Richard Kneller, and Jonathan Timmis. 2020. Cloud computing and firm growth. (2020).

William R Penuel, Jeremy Roschelle, and Nicole Shechtman. 2007. Designing formative assessment software with teachers: An analysis of the co-design process. Research and practice in technology enhanced learning 2, 01 (2007), 51–74.