

Article

# Inspiring the Next Generation of HPC Engineers with Reconfigurable, Multi-Tenant Resources for Teaching and Research

Taha Al-Jody \* , Hamza Aagela and Violeta Holmes \* 

School of Computing and Engineering, University of Huddersfield, Queensgate, Huddersfield HD1 3DH, UK; H.Aagela2@hud.ac.uk

\* Correspondence: Taha.Al-Jody@hud.ac.uk (T.A.-J.); V.Holmes@hud.ac.uk (V.H.)

**Abstract:** There is a tradition at our university for teaching and research in High Performance Computing (HPC) systems engineering. With exascale computing on the horizon and a shortage of HPC talent, there is a need for new specialists to secure the future of research computing. Whilst many institutions provide research computing training for users within their particular domain, few offer HPC engineering and infrastructure-related courses, making it difficult for students to acquire these skills. This paper outlines how and why we are training students in HPC systems engineering, including the technologies used in delivering this goal. We demonstrate the potential for a multi-tenant HPC system for education and research, using novel container and cloud-based architecture. This work is supported by our previously published work that uses the latest open-source technologies to create sustainable, fast and flexible turn-key HPC environments with secure access via an HPC portal. The proposed multi-tenant HPC resources can be deployed on a “bare metal” infrastructure or in the cloud. An evaluation of our activities over the last five years is given in terms of recruitment metrics, skills audit feedback from students, and research outputs enabled by the multi-tenant usage of the resource.

**Keywords:** HPC course structure; HPC cluster and cloud resources for teaching; advanced HPC teaching materials; sustainable education



**Citation:** Al-Jody, T.; Aagela, H.; Holmes, V. Inspiring the Next Generation of HPC Engineers with Reconfigurable, Multi-Tenant Resources for Teaching and Research. *Sustainability* **2021**, *13*, 11782. <https://doi.org/10.3390/su132111782>

Academic Editors: Aleksandra Klačnja-Milićević and Mirjana Ivanović

Received: 25 June 2021

Accepted: 29 September 2021

Published: 25 October 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Fast computing is essential to modern science. It has become a fundamental tool underpinning research and innovation in an ever-increasing array of domains, including the modelling of physical phenomena, fluid dynamics, molecular interactions, astronomy, genomics, game design, social media and even music technology. The dependence on research computing capabilities will continue to grow, especially as new methods are developed that require the processing of large quantities of data. However, accessibility to courses that enable students to study High Performance Computing from an architecture, administration and technology platform perspective does not reflect this level of activity. In the current petascale computing and future exascale, the vast amount of computational power is a formidable challenge that poses unique problems from both application and administration perspectives. There is a distinct skills shortage in terms of how these kinds of systems will be constructed and supported that is not fulfilled by research software engineering or traditional DevOps practices [1].

UK academic institutions offer a diverse range of HPC related study, from short courses, summer schools, degree programmes and post-graduate opportunities for taught or independent research. However, we found that few of these courses offer hands-on experience building HPC systems from the ground up, running serial and parallel code, and benchmark their performance. We do not think that this is a good thing, which provides continued motivation for us to offer Parallel Computer Architecture (PCA) university

modules that fill this gap. We find that students have a high level of engagement with this approach and demonstrate a more acute understanding of exploiting the performance and efficiency gains from a computer cluster that they designed and built themselves. This provides an environment that stimulates discussion between peers and sparks new research projects in HPC infrastructures.

In this context, the main challenge in achieving educational and research goals is providing adequate computational resources to support experiential learning that represents the real world. Our approach has evolved over the last ten years. The work builds on the previous modes of delivery. In 2009, the course was delivered using laboratory hardware to build OSCAR clusters [2]. Further publication explored the use of HPC in education—SAI 2013. In 2015—ISC Frankfurt—presenting container technology that was used to build VCC cluster middleware [3] and VCC [4], followed by the cluster middleware development presented in SC2018 [5]. This work has facilitated the evolution of the cluster middleware used in the delivery of the course. It started by using open source OSCAR middleware, and it was discontinued due to its limitations and lack of continuous support for the software. Therefore, a new cluster middleware was designed and implemented as part of our research, that is using sustainable container technology and tools for the deployment of computer clusters.

In this paper, we describe the current iteration of the systems, as presented at SC2018, and the technology used to provide it in a way, which is sustainable, cost-effective and offers a low administrative burden by considering the teaching aspects in the context of other computational research requirements.

## 2. Landscape of HPC Courses in the UK

A short review of institutions around the United Kingdom offering courses including the High-Performance Computing topic is conducted by [3]. Since 2015, a considerable increase has been observed in HPC related courses provided by institutions around the United Kingdom, and we present a non-exhaustive review in this section, summarised in Table 1. The majority of these courses do not advertise the use of the cloud in their uses. However, some HEI does use the cloud to support their teaching of parallel programming [6,7].

**Table 1.** Overview of HPC related courses in the UK.

Institution	Type	Parallelization	Using HPC	Deploying HPC
University of Liverpool [7]	MSc	✓	✓	✓
University of Leeds [8]	MSc	✓	×	×
The University of Edinburgh [9]	MSc	✓	✓	×
The University of Edinburgh [10]	MSc	✓	✓	×
University of Stirling [11]	MSc	✓	×	×
The University of Dublin [12]	MSc	✓	×	×
University of Plymouth [13]	BSc	×	✓	×
Cardiff University [14]	BSc	×	✓	×
University of Bath [6]	BSc	✓	×	×
ARCHER [15]	Training	✓	✓	×
Warwick University [16]	Short Course	✓	×	×

### 2.1. Undergraduate Courses

In general, there are fewer opportunities available to undergraduate students to get involved in HPC subjects. Cardiff University and Plymouth University both offer BSc courses with *High Performance Computing* in the degree title [13,14]. The relevant modules cover the creation of applications for HPC systems and applying HPC to mathematical problem-solving. The University of Bath also offers two modules that may be chosen on undergraduate computing degrees that will give the student an experience of using and programming HPC systems: the first aims to provide a fundamental understanding of

parallel computing, in terms of types of parallelism and its implementation in software and hardware [6]. The second module aims to provide a comprehensive understanding of software-based solutions to scientific and engineering problems [17].

There are many more institutions offering undergraduate modules that can be selected by students on appropriate courses that involve HPC and research computing related topics in some way but are often difficult to find.

## 2.2. Postgraduate Courses

We found a higher number of institutions offering postgraduate degrees with a focus on HPC and research computing reflected in the title.

The Big Data and High-Performance Computing course at the University of Liverpool offers a wide range of materials for HPC; from designing HPC systems for Big Data processing to implementing the appropriate middleware, testing and optimization of the system [7].

Leeds University's High-Performance Graphics and Games Engineering focus mainly on designing and developing games and graphics through parallel programmes using multi-core HPC systems.

The EPCC at the University of Edinburgh, notable as the host of the ARCHER supercomputers [10], offer a wide range of HPC related modules accessible through the two MSc in HPC and Data Science [9]. These address HPC architecture, the ecosystem of HPC hardware and software in the real world, and programming skills for parallel environments, such as message passing and threading.

University of Stirling offers a Masters course focused on Big Data. It covers data processing with Python, Hadoop, Spark, etc, additionally using a Condor deployment in order to demonstrate distributed computing. This course is focused on the theory and software implementation of data science concepts, rather than traditional HPC applications [11].

Finally, Trinity College Dublin offers a taught MSc degree that provides a foundation in High Performance and technical computing. Students on this course have access to a range of production clusters [12]. Similarly, the available modules cover the programming and application side of HPC, with the opportunity to study data analytics or domain-specific modules in Physics.

## 2.3. Short Courses

A number of short courses related to High Performance Computing are available each year, provided by academic institutions and organizations collaborating with them. The ARCHER training calendar which varies yearly covers a hands-on introduction to HPC and software development through to domain-specific topics and system administration, delivered in person or through webinars [15].

In addition, universities also provide research computing courses available to students and the academic community. In 2016, a program of courses run by the HPC-SC covered HPC from the introduction and system design to advanced parallel programming topics [16]. UCL also runs a series of *Research IT* courses, one which covers High Performance Computing aimed at an introductory level [18]. Many more offer ad hoc training events organized as and when the demand is needed from particular user communities or scientific domains.

## 3. Outline of HPC Courses at the University of Huddersfield

It is clear that there is a gap to be filled in the currently available training from an HPC system design and administration perspective. The readily available courses are not sufficient to prepare a student wishing to specialize in the infrastructure and management side of HPC. Our university's Parallel Computer Architecture (PCA) course provides an introduction to HPC computer clusters, grid technologies, and program parallelization. Although parallel programming using MPI is introduced in the course, the focus is on the HPC infrastructure and management, through the deployment and management of

laboratory computer clusters. The students are equipped with a selection of computer hardware and various components in order to design and build an HPC system. They subsequently gain hands-on practical experience using this system while learning the theoretical aspect of the course.

The students are provided with lab instruction in order to:

1. Set up the hardware and the network configuration.
2. Install the operating system and required packages.
3. Build a batch scheduling cluster using an appropriate middleware from scratch.
4. Test the system by using simple MPI programs and demonstrate scaling using benchmarking software.

The PCA module is offered in two options for Computing and Engineering students with an intermediate level of programming experience, one of which is for undergraduate students that are being taught over two terms with a 1.5 h lab and 2 h lecture weekly. The students cover various topics about parallel computing including the main topic of deploying an HPC system and testing it, running jobs on the HTCondor cluster, and finally use the university's private cloud where they have access to the cloud dashboard and create instances within the cloud environment. The second option of the PCA module is taught to Postgraduate master students, for a shorter period over one term. The students are also provided with instructions and supervision in order to deploy an HPC system, running jobs on the system, and benchmarking the system with the HPC challenge benchmark.

In the latest delivery of the PCA modules on the undergraduate and postgraduate levels, we have deployed additional sustainable resources such as the public Microsoft Azure cloud. The students use these resources to experiment with the virtual computer cluster and cloud technologies. Similar to the hardware-based approach, the students are provided with lab instruction in order to:

1. Set up the Linux Virtual machines and the network configuration on Microsoft Azure Cloud.
2. Install the required packages and the middleware for a virtual computer cluster.
3. Test the system by writing simple MPI programs and demonstrate scaling using benchmarking software.

Students are recruited to the PCA modules from a variety of computing and engineering courses.

### 3.1. HPC Course Learning Outcomes

The course learning outcomes intend to equip the students with comprehensive knowledge about the practical aspects of designing and deploying parallel and distributed processing systems, from both development and usability perspectives:

- Understand the need for HPC system and evolution of computer clusters and cloud in the context of speedup process and data-intensive applications.
- Gain enough knowledge about the fundamental components of cluster environments, such as hardware for commodity components of clusters, network services and communication, Linux operating systems, cluster middleware.
- Gain skills in installing software middleware, bash scrips for job submission and resource management, and Public Key Infrastructure (PKI).
- Understand Message Passing Interface standards and implementations such as MPICH and OpenMPI, parallel programming concepts, data and task parallelism, and speedup—Amdahl's law.
- Gain enough knowledge about the fundamental components of cloud computing, such as Openstack dashboard, Azure dashboard, virtual machines and virtual networks.
- Recognize the scalability issues by using performance metrics and measures, to quantify speed up in parallel applications.

These are the basic key skills, knowledge and understanding required in order to administer and manage HPC systems in a small institution or company.

### 3.2. Delivery and Assessment

The underpinning theoretical concepts of HPC systems architecture and scalability, algorithms and programming models are presented through lectures, followed by practical group work during the laboratory-based sessions. This allows the development of understanding and enhancement of practical skills, whilst providing a mechanism for continuous formative assessment. During the course of the term, the students will build, program and profile a computer cluster. They will learn how to install and configure the operating system, middleware and toolchain in order to compile and run an MPI code. The postgraduate students additionally gain experience with the cloud through the deployment of a web server and load balancer on an OpenStack private cloud or Azure public cloud. The undergraduate students continue the cluster project into the second term, implementing grid technologies for authentication and authorization in order to share their cluster resources with other groups in the same classroom. Everything required is provided in the laboratory sessions: hardware is made available from the University's rolling replacement program and software is hosted on an internal repository. Support is provided by academic staff in the laboratory sessions for all aspects of the module.

The principal strategy for assessment is through project-based learning. Each group is tasked with benchmarking the constructed cluster using a suitable method. It is usually clear to students, through modifying code to include simple measures such as wall time and ping-pong latency, where the inefficiencies in their system lie. They must use this information to develop a critical evaluation of the performance of the designed system. In order to compare and contrast their small laboratory clusters with the university research clusters performance, the students are given access to the university computer clusters. They are expected to write basic bash scripts, submit, and run their jobs and analyse the outputs. In addition to project work, the final element of assessment is time-constrained examination.

## 4. New Approach in HPC Course Delivery

The inspiration for development of the HPC Computer Cluster courses was the work conducted by Rakjumar Buyya from the University of Melbourne, Australia in 2002 [19]. At our university, the initial HPC cluster computing courses were designed in 2008, and subsequent development and delivery of undergraduate and postgraduate courses, have seen significant development. In the past five years we utilized technologies such as Containers, OpenStack, cloud platforms and the hardware used to provision it. The requirements for the hardware and software resources in the HPC laboratory are outlined in this section.

### 4.1. Hardware Requirements

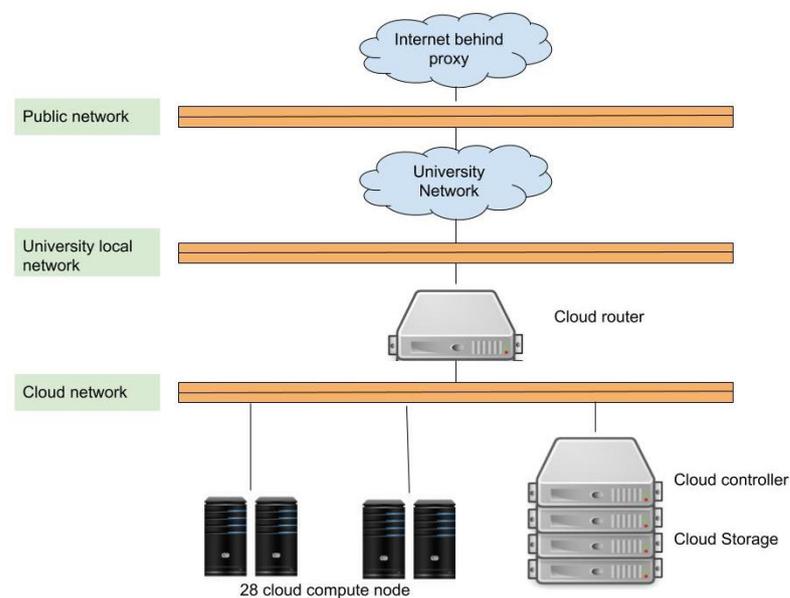
The HPC laboratory facility is mainly reliant on decommissioned PCs and workstations to provide the students with the hardware needed to build their own computer clusters. The laboratory usually houses 30 workstations and several switches to be distributed for each group of students. Clearly, this strategy is not easily implementable and scalable, in terms of required space and required resource [20]. It also requires that each group build relatively small scale clusters in order to conduct their laboratory work. While this affords a high degree of practical, hands-on experience, it compromises the theoretical aspect by limiting the extent to which they can observe effects such as parallel speed-up, for example. Furthermore, when hardware problems occur, in the first instance it is an interesting problem-solving process. However, when a problem persists over several weeks of the course, it becomes tedious to fix and detracts from the experience. Therefore, a balance must be found between these two goals.

One of the iterations of this solution in 2018, was to maintain the physical presence using workstations but aggregate this resource together, rather than being individualized per group. To support this aggregation using a cloud software stack, some additional hardware is required.

A controller node, in order to run essential management services for OpenStack, is provided. The specification for this system is modest, and biased towards memory and network I/O. In addition, two storage nodes with a capacity of 24TB each are provided for block storage.

#### 4.2. Software Requirements

We have found that hard drive failures on the reused computing systems are common [20], and providing a centralized storage service will resolve this limitation. Figure 1 shows the physical infrastructure of the OpenStack private cloud implemented. In order to allow connectivity to the University network in a secure manner, while mitigating the ability for the cloud to be exploited, a NAT router is used to connect the private cloud network with the university network. No public addressing is possible within this configuration.



**Figure 1.** Our University Openstack Cloud architecture.

The process of configuring cloud resources used to build the HPC cluster then becomes a software-defined process that closely resembles the process performed on the physical hardware—such as configuring nodes to a certain specification of CPU, memory and disk space, and creating the network switches and routers required to connect the nodes together.

OpenStack and Azure clouds are used as management platforms to provide this capability through Infrastructure as a Service (IaaS). The OpenStack cloud is built using DevStack, which is a sequence of scripts used to rapidly build a complete OpenStack development environment, based on the latest versions of OpenStack services [21]. This significantly reduces the administrative burden of implementing our solution and allows the environment to be quickly and easily configured and reconfigured at the beginning and end of each academic year, respectively.

The “bare metal” multi-node installation scatters the services of OpenStack amongst the available resources: cloud controller runs the management services, storage nodes run the block storage services Cinder, and worker nodes run the compute service Nova.

Similarly, the deployment of OpenStack cloud on the Azure cloud has the same facilities and services within the OpenStack and provides a larger pool of resources. This cloud solution provides the resources to overcome the limitations of finite laboratory hardware, which were identified in HPC courses delivery over the years, leading to better resource availability, scalability and sustainability.

Microsoft Azure platform is a public cloud provider, which offers IaaS that can be managed via a browser-accessible web portal. The Azure cloud facility is currently used by the students on the undergraduate and postgraduate courses and the university researchers. They can create their own virtual infrastructures at scale within the cloud environment, and run various applications, from hosting websites to deploying The Virtual Container Cluster (VCC) Figure 2.

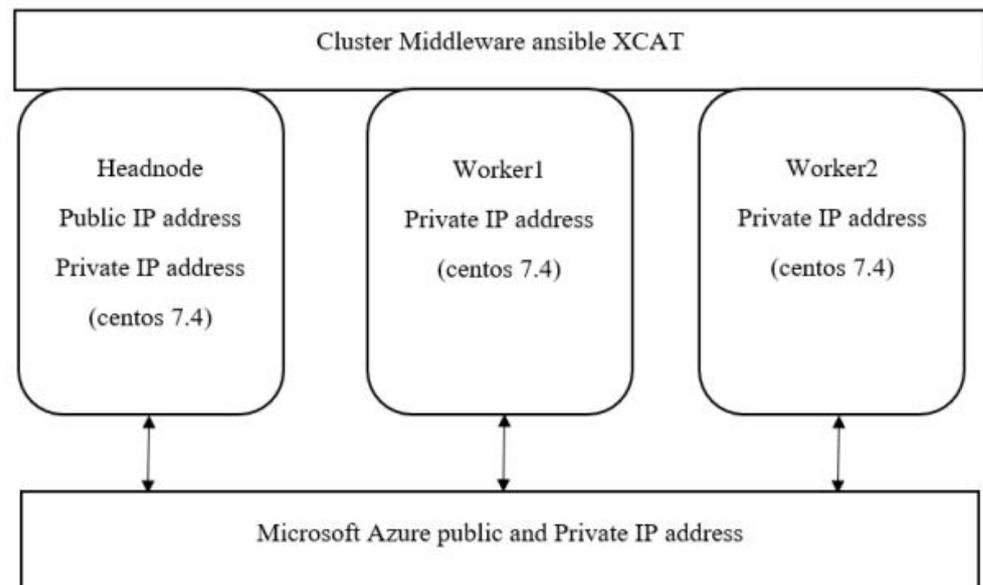


Figure 2. Virtual Container Cluster Architecture cluster architecture.

In order to establish a fully working HPC cluster, the students must install an appropriate middleware within their allocated instances “VM” in the cloud, in the same way, they would on the physical machines. To provide adequate performance, Linux Containers are used rather than traditional Virtual Machines. The user can select the location of the VM according to the user geographic location.

A published work done at the University of Huddersfield details a solution for provisioning, configuration and deploying HPC cluster environment on bare-metal or cloud instances using Ansible [5]. This methodology allows groups who wish to extend and scale their cluster to do so using available resources in the cloud, beyond those they would be able to create in the laboratory environment. In addition, this implementation does not differ from standard HPC implementation, thus is as flexible as extensible as other HPC systems. The software will integrate well with multiple package management systems. Furthermore, the Ansible playbook integration makes it easy for any other HEI to adopt this method of deployment on bare-metal machines or cloud systems.

This offers an improvement in the results that they are able to obtain from the clusters in terms of parallel and scaling performance. However, the process of hosting the cluster on the cloud virtual environment comes with its drawbacks, (a) impact on the cluster performance because of additional network delays and the use of Virtual machines, (b) increase the security vulnerability and privacy concerns, (c) cost management.

## 5. Virtual Cluster Set-Up

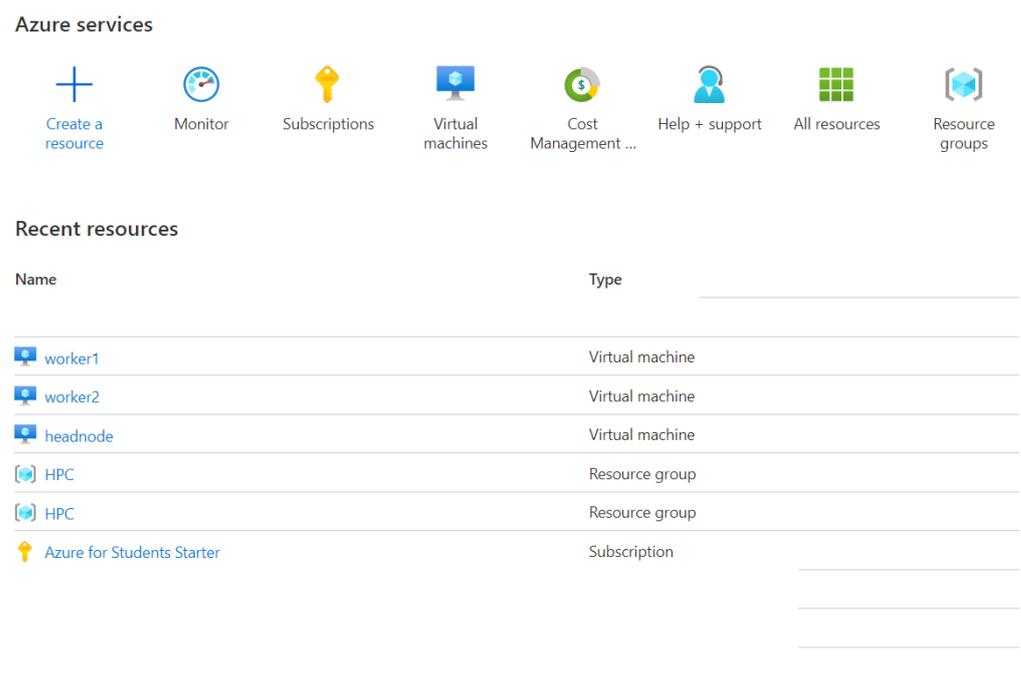
It has been shown that cloud computing topics can be delivered using traditional HPC resources [22]. The idea is to flip this approach, in order to replace the physical machine that was used in the past with a virtualized environment that is indistinguishable from the physical cluster environment.

The Virtual Container Cluster (VCC) toolkit is utilized by students in order to deploy cluster middleware within the container environment [4]. The container solution provides

an accessible approach that does not disguise the underlying configuration but offers some provisioning capability so that nodes do not have to be configured by hand.

The process of establishing a fully working computer cluster could be quite challenging for the students, especially when defining and installing cluster middleware software. Hence, to assist with the deployment of clusters, provisioning and configuration tools such as xCAT [23] and Ansible [24] are used to automate cluster deployment. These tools are integrated within Inception [5,25]. Inception is a set of open-sourced Ansible playbooks for creating an HPC environment on bare-metal machines or the cloud. Therefore, the Inception solution is simple for students to use in a lab environment when deploying a small computer cluster consisting of 2 to 10 nodes.

When a computer cluster is deployed in a cloud, as shown in Figure 3, the students must create at least three instances of Virtual Machines (VMs), for a headnode and two workers/compute nodes. The VMs are pre-configured with all the required basic hardware infrastructure (CPUs, memory and storage, and operating system software such as Linux CentOS). To define network connectivity for the virtual machines, a network interface is configured by specifying virtual network, subnet with a range of private IP addresses to be used for the interconnect within the local cluster network, and a public IP address for the headnode. Once the VMs are running, the VCC and cluster middleware can be deployed on the headnode and the worker nodes VMs. Even though the students are interacting with instances of virtual machines on the cloud, the procedure of the software environments configuration is no different to the procedure carried out using physical machines.



**Figure 3.** Cloud deployment of a virtual computer cluster.

Both OpenStack and Azure dashboards provide a convenient method for remotely interacting with the clusters. In the initial stages of building a cluster, it is useful to be able to interact with all nodes in the system when problems are encountered [26]. This poses a significant problem when working with physical nodes as a large amount of cabling is required for the Keyboard, Video and Mouse (KVM) switches and local consoles. In addition, frequent changing over of cables for different student groups is inconvenient and incurs accidental breakage.

## 6. Evaluation

### 6.1. Student Recruitment Metrics

The modules described in this paper continue to be a popular choice among students, evident from good student feedback, which often leads to their further study in HPC related subjects. In Table 2, we outline the recruitment metrics in terms of the number of students in the last five years. The course is compulsory for our undergraduate computer systems engineering degree (normally recruiting up to 10 students), and optional for other undergraduate degrees. Hence the students from other pathways chose this option. As evidence of success full course delivery, the student numbers are steadily increasing on the undergraduate courses, with consistent numbers on postgraduate courses. In addition, at least 1 alumnus taking the modules each year has continued to pursue a research degree project within the HPC research group at the University. Our graduates are taking roles in research institutes and industry for HPC system administration and maintenance. We feel that this is a good indicator of the performance of the HPC modules.

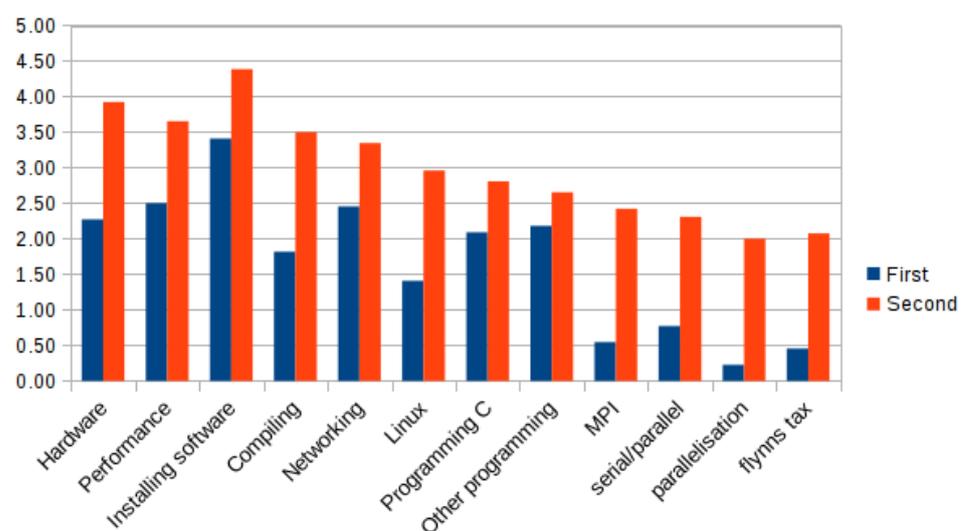
**Table 2.** Recruitment metrics for the PCA modules.

	15/16	16/17	17/18	18/19	19/20
Undergraduate	25	28	38	45	46
Postgraduate	25	23	28	28	28

### 6.2. Audit of Skills Acquired Using OSCAR and VCC Middleware

We solicited a skills audit from several cohorts of students. The first cohort used OSCAR middleware, the complex and detailed method with physical machines, and the second cohort used the container visualization approach with VCC middleware. The skills audit was administered at the beginning of the module (First survey) and the end (Second survey), in order to determine the relative difference in how the student perceived their own understanding in a variety of topics.

In Figure 4, the skills audit results for the cohort using the OSCAR middleware is shown. It can be seen that the HPC specific skills in the right have a low score at the start of the module, as expected. At the end of the module (the second survey), the students' scored their own understanding of these topics.



**Figure 4.** Skills Audit for OSCAR Cluster.

In Figure 5, the results for the cohort using the VCC middleware is shown. It can be seen that the relative increase in the way that the students' appreciated their own skills is comparable to the OSCAR middleware in Figure 4.

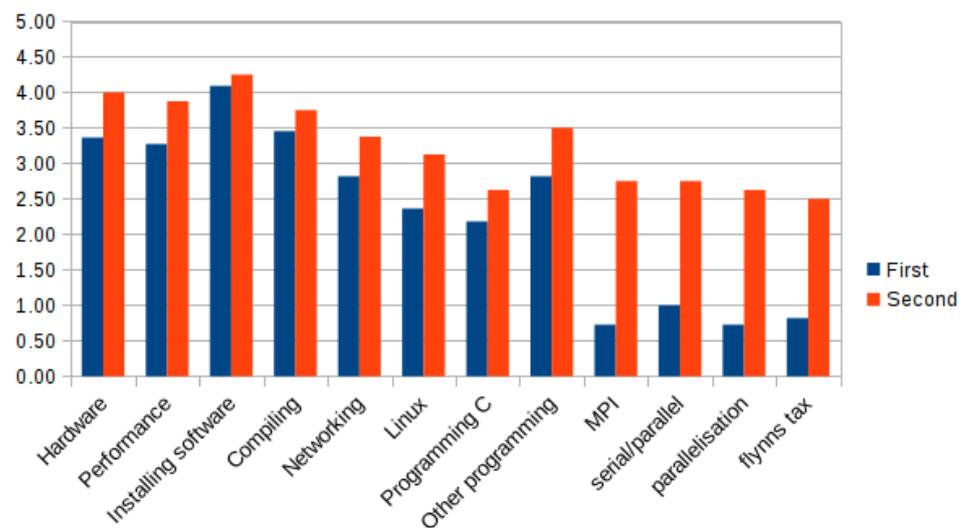


Figure 5. Skills Audit for VCC Cluster.

As shown in Figures 4 and 5, the students have improved their knowledge and skills of cluster computing technology. However, this improvement is more prominent with the use of the VCC cluster middleware. Therefore, it is suggested that this method is better meeting the desired learning outcomes of the course.

### 6.3. Multi-Tenant Usage

The flexibility and multi-tenant capabilities of the Cloud platform have enabled us to efficiently utilise the spare capacity of the resource to conduct research that is not associated with the main teaching function. In addition to teaching, the developed OpenStack cloud environment was also used for research. An example of this is to conduct robotic experiments, running VMs with Robot Operating System (ROS) in order to offload some of the complex tasks, such as mapping, negotiation, path planning and facial recognition. The cloud implementation was used with both humanoid robots, NAO, and Turtlebot, leading to a publication of a number of academic papers. This development in the HPC resource to support the teaching environment and research provides a sustainable multi-tenant solution for teaching and research. As a model for universities that do not currently provide a research-orientated HPC or computational facility, this aspect could provide the catalyst to consider a similar model.

### 6.4. Experience in the Classroom

The changes in the way we teach the modules have resolved some of the limitations previously identified. Flexibility is the main advantage, where some students requested the lab work to be individual rather than in groups since they have not got the full experience of building a cluster. This can be supported with the new approach as visualized resources are given to each student in order to build their own HPC system. It also allows us to scale the resource to suit demand from a number of students.

The new approach also complements the reuse of re-purposed PCs in the laboratory. A more centralized resource is maintained as public and private cloud, improving the sustainability of delivering the module and its learning outcomes going forward. Furthermore, standard PCs laboratories on campus can be utilized instead of requiring a dedicated space. Overall, we believe that this has a significant benefit in terms of the learning environment provided for the students.

Our approach is providing both practical and theoretical knowledge of the basic computer cluster architecture. It relies on the use of commodity PC with Linux operating system, CPU, memory storage and an Ethernet network. The students are made aware of the complexity of production HPC systems as part of their practical access to HPC clusters

on campus (production type clusters). Other complex architectures and different network topologies, complex data sharing, and heterogeneous environments are difficult to teach practically with limited resources. These are addressed in theory only.

## 7. Summary

In this paper, we have presented the results of research in provisioning flexible and scalable resources for the delivery of undergraduate and postgraduate courses in HPC using container virtualization and open-source cloud computing technologies.

The undergraduate and postgraduate Parallel Computer Architectures (PCA) modules at our University are popular among computing and engineering students. They have been delivered over twelve years with the motivation of increasing the knowledge of HPC system engineering and infrastructure, in addition to a solid foundation in parallel computing concepts and programming such as MPI and OpenMP.

The courses see a steady increase in the number of students who choose to take the modules due to their relevance to the HPC in research, business and industry. Continuous development of the HPC laboratory resources is challenging, and there were several different hardware and software solutions offered since the beginning of the delivery of the modules. However, in the past five years, a new approach was adopted to overcome the limitations presented by standard hardware and software for HPC laboratory resources, and it caters for increasing student numbers in the light of the modules' success. We have utilized private and public cloud environments and the HPC deployment tools to deliver a practical HPC course that is sustainable and resolves the limitations in terms of laboratory space, equipment and time constraints. This mechanism of creating virtual cluster environments still meets the set learning objectives and can be used to deliver the same technical content.

An evaluation of our activities over the last five years was presented. These show an increase in our student recruitment, and at least one research project that stems from the module each year. A skills audit of the students demonstrates that the learning objectives are met with the cloud visualized cluster compared to the physical machines.

The course presented in this paper is at an introductory level. Our teaching methodology includes both theoretical and practical computer cluster building and bench-marking using the cloud VMs and the laboratory hardware and MPI codes. In addition, the students have access to the production HPC cluster on campus to experience a real-life cluster usage and profiling.

As evident in Figure 4 the students were reflecting on their experience using hardware equipment and different middleware software. It is evident from the results of this survey that the students have built confidence in using hardware for building a computer cluster with the obsolete OSCAR middleware and a novel, current container-based VCC cluster middleware.

Finally, we explained how the idle capacity of the system, when not in use for teaching, was able to be exploited in a multi-tenant fashion to conduct research. This provides a potential path for establishing a computing facility with longevity and scalability, that promotes an environment where future researchers can gain access and skills on HPC resources, especially in the infrastructure and administration aspects, early in their academic career. Our model of delivering HPC related courses, including the usage of the Inception solution, has been adopted for teaching parallel and distributed architectures modules on the undergraduate courses in Blackburn College (HE), UK. We expect that this model will be adopted in other higher education institutions in the UK.

## 8. Future Work

In order to improve the opportunities for learning, extra-extra-curriculum activities will be conducted, such as a building university team of students, to take part in CIUK Cluster challenge competition [27],

In future, we are planning to invest in the more diverse hardware, including infiniband interconnect and GPU cluster. This approach will elevate current introductory course onto a higher level of skill and competency for students.

We plan to perform a further robust evaluation of this teaching method and publish supporting material for the courses, including resource deployment and student activities.

In addition, the security aspects will be examined to determine if there is any possible vulnerability that could come with implementing such a system within the University Network and Multi-Tenants. We expect that cloud deployment, as a mature platform, should already mitigate many types of misuse of the resource, but it should be quantified to ensure that the risk of deployment in production is low.

Currently, the University of Huddersfield QueensGate Grid uses Bearicade as an HPC portal with advanced security monitoring and attack prevention features [28,29]. We are planning to integrate the deployment of an HPC portal as part of the teaching materials, giving our students full experience in managing HPC systems.

We hope that this work will encourage collaboration and innovation in HPC projects with new, emerging and alternative domain technologies.

**Author Contributions:** All Authors have equally contributed toward this project. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the University of Huddersfield, UK.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Lwakatare, L.E.; Kuvaja, P.; Oivo, M. Dimensions of devops. In *International Conference on Agile Software Development*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 212–217.
2. Holmes, V.; Kureshi, I. Huddersfield university campus grid: Qgg of oscar clusters. *J. Phys. Conf. Ser.* **2010**, *256*, 012022. [CrossRef]
3. Holmes, V.; Kureshi, I. Developing High Performance Computing Resources for Teaching Cluster and Grid Computing Courses. *Int. Conf. Comput. Sci.* **2015**, *51*, 1714–1723.
4. Higgins, J.; Holmes, V.; Venters, C. VCC: A framework for building containerized reproducible cluster software environments. *J. Open Source Softw.* **2017**, *2*, 208. [CrossRef]
5. Higgins, J.; Al-Jody, T.; Holmes, V. Rapid Deployment of Bare-Metal and In-Container HPC Clusters Using OpenHPC Playbooks. In *Proceedings of the Super Computing Conference, Dallas, TX, USA, 11–16 November 2018*.
6. University of Bath. Scientific Computing. Available online: <http://www.bath.ac.uk/catalogues/2012-2013/ma/MA40177.htm> (accessed on 28 March 2018).
7. University of Liverpool. Big Data and High Performance Computing. Available online: <https://www.liverpool.ac.uk/study/postgraduate-taught/taught/big-data-msc/module-details/> (accessed on 28 March 2018).
8. University of Leeds. High-Performance Graphics and Games Engineering MSc. Available online: [https://engineering.leeds.ac.uk/info/201506/module\\_information/172/high-performance\\_graphics\\_and\\_games\\_engineering\\_msc\\_2017](https://engineering.leeds.ac.uk/info/201506/module_information/172/high-performance_graphics_and_games_engineering_msc_2017) (accessed on 28 March 2018).
9. The University of Edinburgh. High Performance Computing with Data Science. Available online: <https://www.ed.ac.uk/studying/postgraduate/degrees/index.php?r=site/view&id=871> (accessed on 28 March 2018).
10. The University of Edinburgh. High Performance Computing MSc. Available online: <https://www.ed.ac.uk/studying/postgraduate/degrees/index.php?r=site/view&id=187> (accessed on 28 March 2018).
11. University of Stirling. MSc Big Data. Available online: <https://www.stir.ac.uk/postgraduate/programme-information/prospectus/computing-science-and-mathematics/bigdata/#degree> (accessed on 28 March 2018).
12. The University of Dublin. MSc in High-Performance Computing. Available online: <https://www.tcd.ie/courses/postgraduate/az/course.php?id=DPTMA-HPCO-1F09> (accessed on 28 March 2018).
13. University of Plymouth. Mathematics with High Performance Computing. Available online: <https://www.plymouth.ac.uk/courses/undergraduate/bsc-mathematics-with-high-performance-computing> (accessed on 28 March 2018).
14. Cardiff University. Computer Science with High Performance Computing. Available online: <https://www.cardiff.ac.uk/study/undergraduate/courses/2018/computer-science-with-high-performance-computing-bsc> (accessed on 28 March 2018).

15. ARCHER. Training. Available online: <https://www.archer.ac.uk/training/> (accessed on 28 March 2018).
16. Warwick University. HPC Short Courses. Available online: [https://warwick.ac.uk/fac/cross\\_fac/hpc-sc/courses/academy/programme/](https://warwick.ac.uk/fac/cross_fac/hpc-sc/courses/academy/programme/) (accessed on 28 March 2018).
17. University of Bath. Parallel Computing. Available online: <http://www.bath.ac.uk/catalogues/2011-2012/cm/CM30225.htm> (accessed on 28 March 2018).
18. University College London. Research IT Training. Available online: <http://www.ucl.ac.uk/isd/services/research-it/training> (accessed on 28 March 2018).
19. Buyya, R. *Cluster Computing*; Nova Science Publishers: Huntington, NY, USA, 2001.
20. Higgins, J.; Holmes, V. Teaching Parallel Computing with Container Virtualization. Available online: [http://ijiset.com/vol2/v2s9/IJISSET\\_V2\\_I9\\_92.pdf](http://ijiset.com/vol2/v2s9/IJISSET_V2_I9_92.pdf) (accessed on 12 April 2018).
21. Awasthi, A.; Gupta, P.R. Comparison of openstack installers. *Int. J. Innov. Sci. Eng. Technol.* **2015**, *2*, 744–748.
22. Eickholt, J.; Shrestha, S. Teaching Big Data and Cloud Computing with a Physical Cluster. In Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education, Seattle, WA, USA, 8–11 March 2017; pp. 177–181.
23. Jeanvoine, E.; Sarzyniec, L.; Nussbaum, L. Kadeploy3: Efficient and scalable operating system provisioning for clusters. *USENIX Assoc.* **2013**, *38*, 38–44.
24. Red Hat. Ansible is Simple IT Automation. Available online: <https://www.ansible.com/> (accessed on 16 June 2019).
25. Clusterworks. Clusterworks—Inception. Available online: <https://github.com/clusterworks/inception> (accessed on 12 December 2018).
26. Kumar, R.; Gupta, N.; Charu, S.; Jain, K.; Jangir, S.K. Open source solution for cloud computing platform using OpenStack. *Int. J. Comput. Sci. Mob. Comput.* **2014**, *3*, 89–98.
27. STFC. Cluster Challenge. 2021. Available online: <https://www.scd.stfc.ac.uk/Pages/CIUK-2021-Cluster-Challenge.aspx> (accessed on 5 July 2021).
28. Al-Jody, T.; Holmes, V.; Antoniadou, A.; Kazkouzeh, Y. Bearicade: Secure access gateway to High Performance Computing systems. In Proceedings of the 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Guangzhou, China, 29 December–1 January 2021; pp. 1420–1427. [[CrossRef](#)]
29. Al-Jody, T. Bearicade. 2017. Available online: <https://github.com/TA3/bearicade> (accessed on 10 September 2020).