↗ xeci6a49

Der KITeGG Cluster – eine Infrastruktur für KI in der Gestaltungslehre

Anton Koch

Chunk 1 Das Verbundprojekt „KI greifbar machen und begreifen: Technologie und Gesellschaft verbinden durch Gestaltung“ (KITeGG), gefördert durch das Bundesministerium für Bildung und Forschung (BMBF) und das Land Rheinland-Pfalz, implementiert seit 2021 an fünf Hochschulen für Gestaltung eine gemeinsame Lehrplattform zur Vermittlung von Kompetenzen rund um das Thema „Künstliche Intelligenz“ (KI). Dabei teilen sich die Hochschule Mainz, HfG Offenbach, HfG Schwäbisch Gmünd, Köln International School of Design und die Hochschule Trier eine gemeinsame Infrastruktur, innerhalb derer eigene Hard- und Softwareressourcen bereitgestellt werden, mithilfe derer experimentelle Lehranwendungen und -methoden entwickelt werden können. Der folgende Artikel beschreibt die konzeptionelle Grundlage, die konkrete Ausgestaltung und die Herausforderungen und Perspektiven der Umsetzung einer gemeinsamen KI-Infrastruktur in der Gestaltungslehre.

Chunk 2 Der Begriff der „Infrastruktur“, aus dem Lateinischen vielleicht passend als „Unterbau“ übersetzt, taucht ursprünglich im Frankreich des ausgehenden 19. Jahrhunderts als Bezeichnung für die baulichen Grundlagen des Eisenbahnbaus auf und wird etwas später auch beim Militär verwendet. Zum Ende des 20. Jahrhunderts wird der Begriff jedoch auch in den Sozial- und Humanwissenschaften immer breiter aufgegriffen und untersucht[1].

1: Pinnix, A., Volmar, A., Esposito, F., Binder, N. (2023). *Rethinking Infrastructure Across the Humanities.* transcript Verlag. S. 16.

Chunk 3

In der Einleitung zu ihrer einflussreichen ethnografischen Studie eines großangelegten Softwareprojekts beschreiben die beiden Soziologinnen Susan Leigh Star und Karen Ruhleder Infrastruktur als „fundamental relationales Konzept“[2], welches „als relationale Eigenschaft in Erscheinung tritt, nicht als Gegenstand unabhängig seines Nutzens“[3]. Diese neue relationale Sichtweise ermöglicht es, Infrastruktur als analytisches Konzept in den Geisteswissenschaften einzusetzen, nicht zuletzt in den „Science and Technology Studies“ (STS). Hierdurch eröffnet sich besonders im Kontext der heutigen Formen digitaler Infrastrukturen die Möglichkeit einer verknüpften Betrachtung von materiellen Bedingungen und Wirkungen mit soziopolitischen Implikationen[4]. Laut Star ist Infrastruktur mehr als nur eine unsichtbare Grundlage, die die Funktion unseres technologischen Alltags gewährleistet. Sie ist eine Form menschlicher Organisation, deren spezifische Eigenschaften und Relevanz in der Beziehung zum Menschen jeweils unterschiedlich in Erscheinung treten. Was für manche eine unsichtbare Selbstverständlichkeit ist, ist für andere ein expliziter Fokus oder ein elementares Hindernis[5]. Im folgenden Text bezieht sich der Begriff der Infrastruktur nun konkret auf ein komplexes soziotechnisches System, das auf einem spezifischen Arrangement von Paradigmen basiert und diese in Abhängigkeiten und Wechselwirkungen zwischen Arbeit und Kapital sowie Politik und Ökonomie reproduziert. Infrastruktur stellt nicht nur passive Grundlagen für die Konzeption, Konstruktion und Funktion der auf ihr realisierten Suprastrukturen bereit, sondern ist in ihrer gesamten Beschaffenheit für diese maßgeblich formgebend und tief mit ihnen verwoben.

Folgen wir dem Physiker und Technologiekritiker Dan McQuillan, so ist „Künstliche Intelligenz“, wie wir sie gegenwärtig erleben und diskutieren, nicht bloß eine Technologie oder ein Forschungsfeld, sondern vielmehr ein „vielschichtiges und verflochtenes Arrangement von Technologie, Institutionen und Ideologie“[6] und damit einer Infrastruktur im erweiterten Sinne nicht unähnlich. Doch was bedeutet ein solches Verständnis von Infrastruktur für ein Projekt, das es sich zur Aufgabe macht, mit dem von ihr bereitgestellten Unterbau eine Untersuchung von Lehrmethoden im Kontext einer so komplexen, exponierten und umstrittenen soziotechnischen Erzählung wie die „Künstliche Intelligenz“ (KI) zu unterstützen? Da es sich hier um ein einerseits offenes und experimentelles, andererseits ein aus öffentlichen Mitteln finanziertes Verbundprojekt verschiedener Hochschulen für Gestaltung, also der angewandten Wissenschaften handelt, sieht sich eine relativ unscharfe Problemstellung einem konkreten Anforderungs- und Anwendungsrahmen gegenüber. Anstelle einer im Voraus explizit definierten Anwendung wird eine offene Umgebung benötigt, die den rechtlichen und organisatorischen Anforderungen des Hochschulbetriebs gerecht wird und gleichzeitig eine dynamische und offene Entwicklung konzeptioneller Methoden und konkreter Anwendungen ermöglicht.

Strategisch bedeutet dies eine konzeptionelle Positionierung auf mehreren Ebenen. Die Infrastruktur gliedert sich in existente technische und administrative Ökosysteme an den verschiedenen Standorten ein, stellt den Entwickler:innen dort eine an ihre Bedarfe angepasste technologische Plattform zur

2: Star, S. L., Ruhleder, K. *Steps Toward an Ecology of Infrastructure: Design and Access for Large Information Spaces* (1996), https://doi.org/10.1287/isre.7.1.111, S. 113.

3: ebd., S. 113

4: Pinnix, A., Volmar, A., Esposito, F., Binder, N. (2023). *Rethinking Infrastructure Across the Humanities*. transcript Verlag., S. 17-18

5: Star, S. L. (1999). The Ethnography of Infrastructure. https://doi.org/10.1177/000276499219553 26, S. 380

6: McQuillan, D. (2022). *Resisting AI: An Anti-fascist Approach to Artificial Intelligence*. Policy Press. ISBN: 978-1-5292-1349-2, S. 1

Verfügung und erfüllt die Anforderungen an die Bereitstellung von Diensten im Lehrbetrieb für Studierende und Lehrende. Im Folgenden sollen nun die wichtigsten grundlegenden strategischen Entscheidungen erläutert, die konkrete Entwicklung der Infrastruktur über den Projektzeitraum beschrieben und daraus eine Zukunftsperspektive entwickelt werden.

Was wiegt schwerer, Cloud oder Metall?

Der Begriff „Cloud“ ist eine in doppelter Hinsicht nebulöse Bezeichnung für ein Geschäftsmodell, basierend auf einer vereinfachten und weitestgehend unverbindlichen Bereitstellung digitaler Dienstleistungen und Anwendungen über das Internet und nach Bedarf. Im Bereich der Informationstechnik sind dies vor allem Hosting, also die Vorhaltung und Wartung von Hardwareressourcen und Konnektivität, ebenso wie komplette Anwendungen und Dienste. In der Regel bauen Anbieter:innen von Cloud-Technologie auf das Konzept eines „Clusters“ auf und vermieten diesen in einer eigenen Konfiguration in Verbindung mit einem Angebot spezieller Produkte und Dienstleistungen. Der Begriff „Cluster“ bezieht sich dabei auf eine Ansammlung von Rechnern, die miteinander in Verbindung stehen. Anders als beim weiträumig vernetzten „Grid“ befinden sich die Rechner in der Regel an einem Ort und verfügen über eine relativ homogene Hard- und Softwarekonfiguration. Innerhalb des Clusters können verteilte Anwendungen ausgeführt werden und die Rechner können von außen erreichbar sein, um Dienste öffentlich bereitzustellen. So betreiben Microsoft, Amazon und Google beispielsweise jeweils eigene globale Netzwerke aus unzähligen Clustern, die zusätzlich zu deren Rechen- und Speicherkapazität ein umfassendes Angebot an Dienstleistungen für den Betrieb und die Entwicklung digitaler Anwendungen bereitstellen. Demgegenüber wird der Begriff des „Bare-metal“ üblicherweise für die Bereitstellung eines „blanken“ Hardwaresystems verwendet, also ohne ein festgelegtes Betriebssystem oder spezifische Software.

Chunk 4

Dies erlaubt maximale Flexibilität in der Konfiguration, verlangt andererseits aber im Vergleich zur Cloud zusätzliche Kompetenz für den Betrieb eigener Hardware und der grundlegenden administrativen Software.

Chunk 5

Zum Entwicklungsaufwand der eigentlichen Anwendung addiert sich die nicht unerhebliche Aufgabe einer kontinuierlichen Wartung und Pflege des für den Betrieb grundlegenden Systems (Abb. 1).

Chunk 6

Abb. 1: Der KITeGG Cluster im Server-Rack an der Johannes Gutenberg-Universität Mainz

Chunk 7

III. p. 15, Chunk 11: AI Literacy for the Long Haul: …

Die Vorteile für den Betrieb von Anwendungen auf einer Cloud-Plattform im Rahmen eines zeitlich begrenzten Projekts liegen auf der Hand. Ohne hohe Investitionskosten, kann direkt eine Anwendungsumgebung aufgesetzt werden, die nach Bedarf wächst und gegebenenfalls auch wieder schrumpft.

Chunk 8

Es gibt jedoch eine Reihe von Aspekten, die in diesem Fall zu einer Entscheidung gegen die Cloud und für eigene Hardware geführt haben. Dies war nicht zuletzt eine Kosteneinschätzung, da sich, auf die Projektlaufzeit gesehen, die Anschaffungskosten gegenüber den Kosten für externe Dienstleister als wesentlich niedriger herausstellten. Mag die Anschaffung für ein kurzzeitiges Projekt von ein oder zwei Jahren in dieser Größe nicht tragbar sein, so amortisiert sich die Hardware bereits ungefähr auf der Hälfte der fünfjährigen Projektlaufzeit. Eine Arbeitsstelle in Vollzeit zur Entwicklung und Wartung der Hard- und Softwareplattform ist für beide Varianten unverzichtbar und addiert sich in jedem Fall. Unabhängig davon ist es jedoch, selbst bei geringeren Kosten durch Outsourcing, für ein aus öffentlichen Geldern finanziertes Forschungsprojekt angebrachter, diese Gelder in den Aufbau und den Betrieb öffentlicher Infrastruktur zu investieren. Zudem gibt es an vielen Hochschulen oder Universitäten bereits existente Hosting- und Housing-Kapazitäten, was die Ausgaben auf die Hardwareanschaffungen, Schrankmiete, Energie- und Kühlkosten beschränkt.

Im Hochschulkontext lohnt sich eine solche Investition insbesondere im Hinblick auf Personalentwicklung und lokale Kompetenzbildung. Wenn ausschließlich auf proprietäre Dienste kommerzieller Anbieter zurückgegriffen wird, entstehen zwar kurzfristig Kapazitäten, jedoch langfristig weniger tiefgreifende Kompetenzen am Standort. Stattdessen kann gerade ein Verbund von Bildungsinstitutionen aus den Angewandten Wissenschaften wichtige Mehrwerte generieren, indem er Studierende und Lehrende als maßgebliche Teilhaber:innen begreift, deren zentrales Interesse ein optimaler Wissenstransfer innerhalb einer experimentellen und praxisorientierten Infrastruktur ist. Zudem kann der Aufbau eigener Anwendungsfälle, Plattformen und Grundlagen unter aktiver Mitwirkung

7: Varoufakis, Y. (2024). *Technofeudalism: What Killed Capitalism?*. ISBN: 9781529926095, S. 97 ff.

8: Moss, S. (2024, 27. Oktober). *Blackstone's prospective data center pipeline hits $100bn, on top of $70bn portfolio*. Data Centre Dynamics. Abgerufen am 20. Dezember 2024 von https://www.datacenterdynamics.com/en/news/blackstones-prospective-data-center-pipeline-hits-100bn-on-top-of-70bn-portfolio/

von Studierenden stattfinden, die dabei wiederum praktische Kompetenzen ausbilden können. Im Gegensatz dazu schafft und verstärkt das „Cloud Sourcing“, also das Auslagern digitaler Infrastruktur an externe Unternehmen, Abhängigkeiten von volatilen Preismodellen, externen Fachkräften und eher allgemein und wenig spezifisch konzipierten Anwendungen. Dabei fließen große Teile öffentlicher Gelder direkt in die Renditen privater Anteilseigner:innen, während sich gleichzeitig technologisches Knowhow an wenigen Zentren in einem global organisierten monopolistischen Geschäftsfeld konzentriert. So beschreibt der Ökonom Yannis Varoufakis die Entstehung eines sogenannten „Cloud-Kapitals“, welches sich außerhalb der üblichen marktwirtschaftlichen Regeln aus der Akkumulation von Infrastruktur und deren Vermietung speist[7]. Am stetig wachsenden Markt für Rechenzentren hielt die Private-Equity-Firma Blackstone im Oktober 2024 bereits Investitionen in Höhe von 55 Milliarden US-Dollar und plante die Investition weiterer 100 Milliarden[8]. Dabei konzentrieren sich hier Macht und Ressourcen in den Händen weniger privater Firmen und Kapitalgeber:innen, von denen die globale digitale Infrastruktur zunehmend abhängiger wird.

Chunk 9 Die funktionale Abhängigkeit schließlich äußert sich bei einem voll entwickelten System in den sogenannten „Switching Costs“, also den Kosten und dem Aufwand eines möglichen Anbieterwechsels. Die von den Cloud-Anbietern bereitgestellten Dienste sind nicht unbedingt mit jenen von Konkurrenten interoperabel und sobald erst einmal ein signifikanter Datenbestand akkumuliert und komplexere Funktionalität in der Cloud implementiert wurde, wird ein Wechsel immer kostspieliger und unwahrscheinlicher, selbst wenn sich die wirtschaftlichen, rechtlichen und ethischen Konditionen dieser Geschäftsbeziehung über die Zeit verschlechtern.

Betrachtet man die heutige digitale Infrastruktur aus ökologischer Sicht, so ist ihr momentan prominentestes Problem der massive Energie- und Ressourcenverbrauch.

Chunk 10 Dies betrifft zunächst einmal Betrieb und Herstellung der benötigten Hardware, einschließlich der damit verbundenen Abhängigkeiten von extraktiven Industrien zur Gewinnung nötiger Rohstoffe. Nicht zuletzt aber sind auch die Entsorgung und das Recycling obsoleter Geräte aufwändige Prozesse, die ihrerseits ressourcenintensiv und ökologisch problematisch sind. Diese Probleme sind in der gegenwärtigen Konfiguration der globalen Wirtschaft kaum zu umgehen. Es gibt insofern keine hundertprozentig „grüne“ oder nachhaltige digitale Technologie, da die Hardware größtenteils auf geplante Obsoleszenz hin konzipiert wird und immer sowohl implizit als auch explizit mit einer problematischen Wertschöpfungskette verbunden ist. Dies betrifft ebenso den stark wachsenden Markt für Cloud-Dienstleistungen. Große Anbieter wie beispielsweise Google berufen sich auf eine steigende Effizienz ihrer Anlagen gemäß der „Power Usage Effectiveness“ (PUE) und Maßnahmen zur Reduktion des CO2-Ausstoßes durch den Einsatz erneuerbarer Energien oder sogenanntes „Offsetting“. Dabei ist allerdings die PUE mittlerweile umstritten, da sie nicht umfassend genug ist[9], ebenso wie der Ankauf sogenannter „Carbon-Credits“, einer Art von Ablasshandel über ausgleichende Investitionen in Klimaschutzprojekte und regenerative Technologien[10].

9: Horner, N., Azevedo, I. (2016). *Power usage effectiveness in data centers: overloaded and underachieving.* https://doi.org/10.1016/j.tej.2016.04.011

10: Google (2024). *Effizienz.* Abgerufen am 20. Dezember 2024 von https://www.google.com/about/datacenters/efficiency/

Chunk 11 Jedoch auch unbeachtet dieser Kritik werden diese Bestrebungen durch die spezifische Form der Cloud-Ökonomie konterkariert. Es werden im Zuge der allgemeinen Digitalisierung immer mehr Dienstleistungen „in die Cloud“ verlagert, wo sie einen enormen Bedarf an Rechenleistung erzeugen, dem wiederum von den Anbietern mit der Einrichtung von zusätzlichen und immer größeren Rechenzentren begegnet wird. Diese Rechenzentren stellen ihrerseits eine massive punktuelle Belastung der örtlichen Energieinfrastrukturen dar[11], sodass deren Bau mittlerweile mehr und mehr Widerstand begegnet, da es vermehrt Zweifel an deren Tragbarkeit und Nachhaltigkeit gibt[12].

Chunk 12 Betrachtet man beispielsweise den kumulativen PUE-Wert für Google, zeigen die letzten 10 Jahre eine Steigerung der Effizienz um ca.

Chunk 13 10%, während sich der kumulative Stromverbrauch im selben Zeitraum um mehr als 600% gesteigert hat[13]. Ebenso hat Microsoft seinen 2020 verkündeten „Carbon-Moonshot“[14], ein ambitionierter Plan zur Erreichung einer negativen CO2-Bilanz bis 2030, wieder zurückgenommen, indem Firmenpräsident Brad Smith in einem Interview mit Bloomberg zugab, „der Mond hat sich [durch den K.I.-Boom, Anm. d. Verf.] nun fünfmal so weit entfernt als zuvor“[15].

Chunk 14 Dies betrifft längst nicht mehr nur die USA, denn auch in Spanien, das mit massiver Trockenheit kämpft, plant Facebooks Mutterkonzern Meta die Errichtung eines Hyperscale-Rechenzentrums, das nach Schätzungen jährlich 600 Millionen Liter Trinkwasser verbrauchen wird[16]. Die Belastung der lokalen Wasserinfrastruktur ist ein wesentlicher Grund, warum es mehr und mehr Widerstand gegen den Bau neuer Rechenzentren gibt.

Chunk 15 Und während utopische Versprechungen von bahnbrechenden technologischen Transformationen durch KI-Technologie weiterhin größtenteils Science-Fiction bleiben, sind Unternehmen wie Google, Amazon und Microsoft bereits heute aktiv dabei, eben diese neuen technologischen Infrastrukturen für den Ausbau fossiler Förderkapazitäten[17] und der Automation von Kriegsführung[18] bereitzustellen, da diese Felder die größten unmittelbaren Erträge liefern.

Eine bewusste Konfrontation mit den materiellen Bedingungen von Digitalisierung und den damit verbundenen gesellschaftlichen Transformationen sollte ein wesentlicher Teil des Angebots einer umfassenden Hochschulbildung sein.

11: Kearney, L., Dareen, S., Vakil, D. K. (2024, 10. April). *US electric utilities brace for surge in power demand from data centers.* Reuters. Abgerufen am 20. Dezember 2024 von https://www.reuters.com/business/energy/us-electric-utilities-brace-surge-power-demand-data-centers-2024-04-10/

12: Bryan (2024). *Data centres curbed as pressure grows on electricity grids.* Financial Times. Abgerufen am 20. Dezember 2024 von https://www.ft.com/content/53accefd-eca7-47f2-a51e-c32f3ab51ad5

13: Google. (2024). 2024 Environmental Report. Abgerufen am 20. Dezember 2024 von https://sustainability.google/reports/google-2024-environmental-report/. S. 33

14: Smith, B. (2020, 16. Januar). *Microsoft will be carbon negative by 2030.* Official Microsoft Blog. Abgerufen am 20. Dezember 2024 von https://blogs.microsoft.com/blog/2020/01/16/microsoft-will-be-carbon-negative-by-2030/

15: Rathi, A., Bass, D. (2024, 15. Mai). *Microsoft's AI Push Imperils Climate Goal as Carbon Emissions Jump 30%.* Bloomberg. Abgerufen am 20. Dezember 2024 von https://www.bloomberg.com/news/articles/2024-05-15/microsoft-s-ai-investment-imperils-climate-goal-as-emissions-jump-30

16: Pascual, M. G. (2023, 15. November). *Artificial intelligence guzzles billions of liters of water,* El Pais, Abgerufen am 20. Dezember 2024 von https://english.elpais.com/technology/2023-11-15/artificial-intelligence-guzzles-billions-of-liters-of-water.html

17: Hao, K. (2024, 13. September). *Microsoft's Hypocrisy on AI,* The Atlantic, Abgerufen am 20. Dezember 2024 von https://www.theatlantic.com/technology/archive/2024/09/microsoft-ai-oil-contracts/679804/

18: González, R. J. (2024). *How Big Tech and Silicon Valley are Transforming the Military-Industrial Complex.* Watson Institute. Abgerufen am 20. Dezember 2024 von https://watson.brown.edu/costsofwar/papers/2024/SiliconValley

Chunk 16 Dies ist umso wichtiger bei einer derart mythologisierten soziotechnischen Apparatur wie KI, die von ihren Verfechtern als universelle Technologie angepriesen wird[19], jedoch in der Praxis gerade in dieser unspezifischen Anwendungsform eher ineffizient gegenüber spezialisierten Anwendungen ist[20] und zumeist nicht ohne zusätzliche menschliche Kontrolle und Intervention auskommt. Diese notwendige Auseinandersetzung kann jedoch auf mehreren Ebenen stattfinden, denn zunächst bedeutet der Besitz eigener Hardware eine bewusste Beschäftigung mit deren Anschaffungs- und Betriebskosten sowie den für den Betrieb nötigen Bedingungen.

19: Eloundou, T., Manning, S., Mishkin, P., Rock, D. (2023). *GPTs are GPTs: An Early Look at the Labor Market Impact Potential of Large Language Models.* https://doi.org/10.48550/arXiv.2303.10130

20: Luccioni, A. S., Jernite, Y., Strubell, E. (2023). *Power Hungry Processing: Watts Driving the Cost of AI Deployment?.* https://doi.org/10.48550/arXiv.2311.16863

Chunk 17 Das bedeutet, dass zumindest die Betriebskosten dort anfallen, wo sie verursacht werden, anstatt sie global dorthin auszulagern, wo diese am günstigsten wären, sei es durch niedrige Energiepreise oder schwache rechtliche Regulation. Ein weiterer Aspekt ist die freie Zuteilung von Ressourcen, da in einem kommerziellen Rechenzentrum ein erheblicher Teil des Energiebedarfs auf Redundanz entfällt, um Ausfallsicherheit und schnelle Verfügbarkeit zu garantieren. So hat beispielsweise die New York Times 2012 eine Studie in Auftrag gegeben, die zeigte, dass die darin untersuchten Rechenzentren nur zwischen 6% und 12% ihres Strombedarfs für tatsächliche Rechenoperationen verwendeten, während der überwiegende Teil für Geräte in Bereitschaft verwendet wird, um eventuelle Spitzen in der Auslastung abzufangen[21]. Diese Überkapazitäten werden jedoch nicht in einem solchen Maße für dieses Projekt benötigt, da nur bestimmte Ressourcen redundant sein müssen.

21: Glanz, J. (2012, 22. September). *Power, Pollution and the Internet.* The New York Times. Abgerufen am 20. Dezember 2024 von https://www.nytimes.com/2012/09/23/technology/data-centers-waste-vast-amounts-of-energy-belying-industry-image.html

III. p. 258, Chunk 7: KI-TeGG und nun?

Chunk 18 Die Hardware kann so kleiner spezifiziert werden, benötigt insgesamt weniger Energie und es kann in diesem Szenario für alle Teilhaber:innen transparenter dargestellt werden, welche materiellen Konsequenzen die Arbeit mit der Technologie tatsächlich hat. Letztlich ist allerdings ein wesentlicher Aspekt die Positionierung der Hochschule als ein Ort der Wissensproduktion und kritischer Reflexion, anstatt einer passiven Erfüllungsgehilfin für die konzeptionell beschränkten und spezifisch ideologisch geprägten Geschäftsmodelle multinationaler Konzerne. Die Aufgabe der Wissenschaften sollte nicht die Legitimation von Firmenstrategien durch die Suche nach passenden Anwendungsfällen sein, sondern die Vermittlung von Fähigkeiten zur eigenständigen Betrachtung und Bewertung der auf dem Markt verfügbaren technologischen Angebote und ihrer materiellen und ideologischen Bedingungen.

Strategien und Paradigmen – die theoretische Infrastruktur

Das anfangs erwähnte erweiterte Verständnis von Infrastruktur, welches über die Bereitstellung einer Grundlage zur Realisation von technologischen Implementierungen hinausgeht, verändert die Perspektive auf den Cluster, seine ihm eigene und die ihn umgebende Topologie sowie seine Rolle als Begleitung und praktische Bedingung eines dynamischen Lehr- und Forschungsprozesses. So wird die gesamte technologische Infrastruktur und die darauf

implementierte Software nicht als Produkt verstanden, sondern als Ausdruck einer durch die gegenwärtige Lesart des Konzepts der „Künstlichen Intelligenz“ geprägten technologischen Praxis im Kontext gestalterischer Ausbildung. Vor diesem Hintergrund sollte die Infrastruktur neben den materiellen Voraussetzungen eine theoretische und reflexive Komponente integrieren, die eine kritische und (de-)konstruktive Praxis auf allen Ebenen ermöglicht. Diese ist einerseits wichtig im Kontext des „Platform-Engineerings“, der Softwareentwicklung, aber auch der bewussten Nutzung dieser Technologien unter Einbeziehung ihrer geschichtlichen, sozialen und politischen Dimensionen sowie der Vielzahl ihrer Erzählungen und Mythen.

Strategisch bedeutet dies, nicht ausschließlich nach sogenannten „Best-Practices“ zu gestalten, sondern immer auch deren Ursprünge und Zielsetzungen zu untersuchen. So können beispielsweise gängige Qualitäten wie „Redundanz“ oder „Effizienz“ anders bewertet werden, als dies im Kontext einer klassischen Hosting-Anwendung der Fall wäre. Die Untersuchung und Reflexion angewandter Methoden und Paradigmen erfordert wiederum die Möglichkeit ihrer historischen, soziopolitischen und philosophischen Betrachtung. Auch wenn diese dabei in einer praxisorientierten angewandten Lehre nicht in die Tiefe gehen kann, bietet sich hier die Chance, Studierenden einen breiten Einblick in die Zusammenhänge und das Wirken dieser Infrastruktur zu gewähren und kontrastierend zur populären Darstellung auf die zahlreichen Positionen aus ökologischer, feministischer und antirassistischer Wissenschafts- und Technologiekritik hinzuweisen. Dabei ist es wichtig, im Kern zu vermitteln, dass unsere Vorstellungen von „Digitalität“ und in diesem Fall deren spezifische Auslegung als „Künstliche Intelligenz“ kulturell und historisch geprägt sind.

Chunk 19 Sie sind keine statischen und universellen Konzepte, sondern können immer auch alternativ gedacht und realisiert werden.

Die Anatomie eines Clusters – die technologische Infrastruktur

In seiner grundlegenden Ausrichtung muss das Design der Cluster-Infrastruktur und der darauf basierenden Software-Umgebung zwei Nutzungsebenen bedienen, die untereinander kontinuierlich in einem reziproken Verhältnis interagieren. Auf der einen Seite ist dies der Einsatz in der Lehre als möglichst stabile Ressource in Kursen, andererseits aber auch die konstante Erprobung und Entwicklung von neuen Anwendungen, die sich aus der Lehre ableiten und diese wiederum beeinflussen.

Chunk 20 Zudem sollte aufgrund des festen Projektzeitraums eine möglichst schnelle initiale Bereitstellung für die Lehrenden und Entwickler:innen an den Partnerinstitutionen gewährleistet werden, um möglichst früh mit praktischen Experimenten in der Lehre beginnen zu können. Daher wurde auf eine längere vorausgehende Entwicklungsphase verzichtet und stattdessen zunächst nur die Open-Source Software JupyterHub aufgesetzt, eine Mehrfachnutzerumgebung, über die einzelne Nutzer über den Web-Browser Zugriff auf GPU-Ressourcen, eigenen Speicherplatz, eine Programmier- und Linux-Umgebung bekommen. Somit konnte bereits wenige Wochen nach Projektstart an den Standorten praktisch auf dem Cluster gearbeitet werden, während parallel die Entwicklung einer eigenen Lehr-Lern-Plattform vorangetrieben wurde.

Chunk 21

Der Cluster basiert auf einer Ausstattung von fünf sogenannten „HGX-Servern“[22] mit jeweils acht Grafikprozessoren des Typs NVIDIA A100 (80 GB VRAM), zwei CPUs mit je 64 Kernen, 2 TB RAM sowie sechs SSD-Modulen mit je 30 TB, auf denen die Anwendungen für die Lehre laufen, zwei Servern als redundanten Datenspeichern mit jeweils zwanzig magnetischen Laufwerken von je 20 TB Größe und schließlich zwei Management-Servern, die die Verteilung von Ressourcen auf den Servern für Anwendungen koordinieren (Abb. 2).

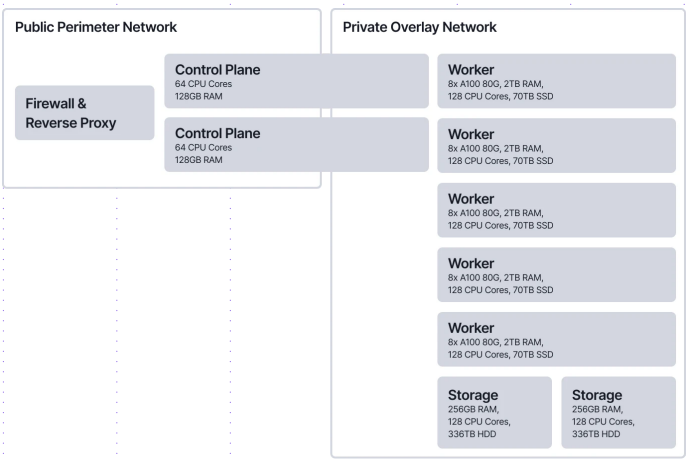

Abb. 2: Hardware-Topologie des KITeGG Clusters

Chunk 22

Auf allen Servern läuft das Linux-Betriebssystem „Ubuntu“, in dem dann Kubernetes, „ein Open-Source-System zur Automatisierung der Bereitstellung, Skalierung und Verwaltung von containerisierten Anwendungen“[23], zusammen mit einigen nötigen Treibern installiert wird (Abb. 3).

22: NVIDIA (2024). *NVIDIA HGX-KI-Supercomputer*. Abgerufen am 20. Dezember 2024 von https://www.nvidia.com/de-de/data-center/hgx/

23: The Linux Foundation, (2024). *Kubernetes*. Abgerufen am 20. Dezember 2024 von https://kubernetes.io/de/

24: Longhorn (2024). *Cloud native distributed block storage for Kubernetes*, Abgerufen am 20. Dezember 2024 von https://longhorn.io

25: Tigera (2024). *Project Calico*, Abgerufen am 20. Dezember 2024 von https://www.tigera.io/project-calico/

26: Keycloak (2024). *Open Source Identity and Access Management*, Abgerufen am 20. Dezember 2024 von https://www.keycloak.org

27: The Linux Foundation (2024). *Prometheus Website*, Abgerufen am 20. Dezember 2024 von https://prometheus.io

28: Traefik Labs (2024). *Traefik Proxy: The Cloud Native Application Proxy*. Abgerufen am 20. Dezember 2025 von https://traefik.io/traefik

29: The Linux Foundation (2024). *Harbor*. Abgerufen am 20. Dezember 2024 von https://goharbor.io

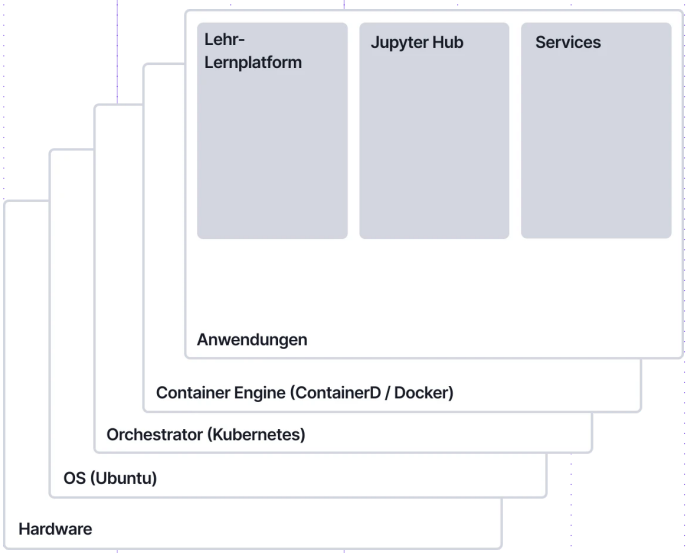

Abb. 3: Die Software-Architektur

Die weitere Verwaltung und Konfiguration des Clusters erfolgt hauptsächlich über die administrativen Schnittstellen von Kubernetes. So können benötigte Ressourcen im YAML-Format beschrieben und dann automatisch im Cluster bereitgestellt werden. Innerhalb von Kubernetes können so Dienste definiert werden, die existente Lösungen bereitstellen, beispielsweise für verteilte Datenspeicher[24], Konnektivität[25], Authentifizierung[26], Monitoring[27], aber auch Reverse-Proxies mit SSL-Terminierung[28] und eigene Docker Container Repositorien[29]. Diese grundlegende Softwareausstattung ermöglicht es, darauf schnell und unkompliziert einzelne Dienste testweise oder permanent einzurichten, um diese dann eingeschränkt oder im ganzen Verbund nutzen zu können.

Chunk 23

Somit entsteht eine flexible Basisinfrastruktur, die schon nach kurzer Einrichtungszeit einen grundlegenden Lehrbetrieb ermöglicht, während gleichzeitig die Entwicklung einer dedizierten Lehr-Lernplattform (LLP) startet, die die Lehre konstant begleitet. So konnte direkt zu Beginn des Projekts die Open-Source Software JupyterHub[30] eingerichtet werden, die es ermöglicht, einzelnen Nutzer:innen eigene JupyterLab-Instanzen[31] zur Verfügung zu stellen.

Chunk 24

Diese können über den Browser aufgerufen werden und bieten, neben einer Programmierumgebung in sogenannten „Notebooks“, Zugriff auf die GPU-Hardware in Verbindung mit einem Dateisystem und einer Umgebung zur Ausführung von Linux-Kommandos.

Chunk 25

30: Project Jupyter *(2024). JupyterHub.* Abgerufen am 20. Dezember 2024 von https://jupyter.org/hub

31: Project Jupyter (2024). *JupyterLab.* Abgerufen am 20. Dezember 2024 von https://jupyter.org

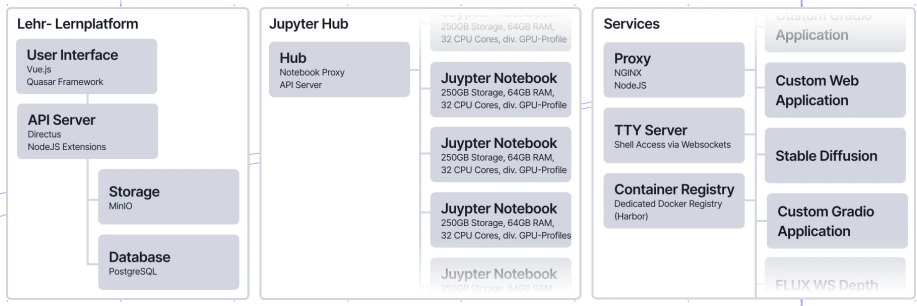

Abb. 4: Die drei wesentlichen Software-Komponenten des KITeGG Clusters

Chunk 26

Im Wesentlichen handelt es sich hierbei zwar um eine gängige Plattform für High-Performance-Computing (HPC), allerdings unterscheidet sie sich in ihren Anforderungen von gängigen Infrastrukturen dieser Art. Im Gegensatz zur üblichen Automation parallel abzuarbeitender Aufgaben („Jobs“), also im Voraus geplanter, definierter und dann nur noch auszuführender Rechenprozesse, muss hier die Möglichkeit einer interaktiven Arbeitsphase gegeben sein.

Chunk 27 Die Studierenden können dabei über einen gewissen Zeitraum auf bestimmte Ressourcen zugreifen und sich iterativ an eine Aufgabe annähern oder gänzlich ergebnisoffen experimentieren. Dabei muss die Gesamtkapazität so aufteilbar sein, dass zu jeder Zeit genügend parallele Sitzungen für die Studierenden verfügbar sind, um Kurse und Arbeiten an allen Standorten zu ermöglichen. Während also in üblichen HPC-Umgebungen ein „Job“ definiert und dieser dann in eine Warteschlange („Queue“) gegeben wird, werden hier für Personen und Gruppen jeweils spezifische Konfigurationen von Ressourcen über bestimmte Zeiträume reserviert, egal ob diese letztendlich genutzt werden. Kubernetes stellt ein System für die dynamische Zuweisung von Ressourcenlimits für laufende Instanzen sogenannter „Pods“ (funktional gruppierte Container) bereit, welches die Verteilung von CPUs, GPUs und Arbeitsspeicher erlaubt.

Chunk 28 Zudem erhalten alle Nutzer:innen dieser interaktiven Sitzungen eigene virtuelle Laufwerke von variabler Größe, die es ihnen ermöglichen, ihre Arbeitsumgebungen über die Sitzungen hinaus zu persistieren. Diese interaktive Umgebung wird ergänzt durch eigene Dienste, die von Lehrenden aus dem Verbund entwickelt und entweder ausgewählten oder allen Studierenden permanent bereitgestellt werden. So können beispielsweise Implementierungen von Open-Source Text-zu-Bild-Generatoren so verändert werden, dass sowohl deren Gebrauch einfacher wird als auch eine bessere didaktische Integration durch Anpassungen an bestimmte Aufgaben im Kurs unterstützt wird. Dabei können mehrere Studierende einen einzelnen Dienst nutzen, was wiederum Redundanz und somit den Ressourcenbedarf reduziert.

Forschung – die Entwicklungsumgebung

Innerhalb des Verbunds entstehen je nach Aufstellung und Ausrichtung der jeweiligen Hochschule und den Zielen der Lehrenden jeweils unterschiedliche Kombinationen allgemeiner technischer Bedarfe mit spezifischen Interessen, Einsatzfeldern und Strategien. Aus diesem Grund wird eine

gemeinsame Entwicklungsumgebung angestrebt, die eine autarke und individuelle Softwareentwicklung ermöglicht, gleichzeitig aber Transparenz im Verbund schafft, um gegenseitig auf die Entwicklungen und Erkenntnisse an anderen Standorten aufbauen zu können oder daran inhaltlich zu partizipieren.

Chunk 29 Das Land Rheinland-Pfalz stellt an der Johannes Gutenberg-Universität (JGU) eine eigene GitLab-Installation[32] zur Verfügung, auf der sich Softwareprojekte kollaborativ entwickeln lassen und Funktionen zur Automation und Organisation von Prozessen zur Bereitstellung dieser Software angeboten werden. Eine für das Verbundprojekt eingerichtete Gruppe bündelt die zahlreichen Unterprojekte und erlaubt eine verteilte kollaborative Entwicklung von Diensten, Anwendungen sowie Lehrmaterialien.

32: Rechenzentrumsallianz Rheinland-Pfalz, ZDV Universität Mainz (2024). *GitLab RLP*. Abgerufen am 20. Dezember 2024 von https://gitlab.rlp.net

Chunk 30 Dies geschieht sowohl verbundübergreifend als auch an den einzelnen Standorten mit individuellen Zugriffsrechten für Mitarbeitende und externe Partner:innen. Zudem können dort einzelne Funktionen, Fehlerberichte und Wünsche oder explizite Bedarfe zentral erfasst und diskutiert werden. Für die kurzfristige und unkomplizierte Erprobung von existenten Open-Source-Projekten und -Modellen können die Mitarbeitenden auf die JupyterLab-Umgebung zurückgreifen, um dort Anwendungen auszuprobieren oder eigene Modelle zu trainieren.

Da die Bereitstellung von Diensten im Cluster ausschließlich über Docker-Container realisiert wird, eignen sich die von GitLab angebotenen Funktionen für kontinuierliche Integration und Bereitstellung (CI/CD), um Mitarbeitenden die Möglichkeit zu geben, eigene, auf Docker basierende Anwendungen lokal zu entwickeln, diese automatisiert zu kompilieren und in das projekteigene Container-Repositorium laden zu lassen. Dies betrifft sowohl eigens angepasste Versionen des JupyterLab-Servers als auch beliebige andere Dienste, die auf bekannten Open-Source-Projekten oder eigenständigen Entwicklungen basieren.

Chunk 31 Diese Docker-Images können dann über die oben beschriebene Funktion in der LLP innerhalb des Clusters instanziiert und bei Bedarf auch außerhalb über das Internet oder in den Hochschulnetzwerken verfügbar gemacht werden. Dabei werden einzelnen Arbeitsbereiche innerhalb von Kubernetes als sogenannte „Namespaces“ abgebildet, zwischen denen selektiv Kommunikation erlaubt oder unterbunden werden kann.

Die JupyterLab-Instanzen (Abb. 5) haben dabei eine in mehrfacher Hinsicht wichtige Aufgabe. Sie sind Umgebungen für Studierende, die hier erste Schritte in Python-Programmierung mithilfe sogenannter „Notebooks“, einer Kombination von kurzen Programmierschritten und der illustrativen Darstellung deren unmittelbarer Effekte, machen können. Zudem kann das Training eigener oder die Anpassung bestehender Modelle durchgeführt und über eine Erweiterung eine lokale Programmierumgebung in Microsoft Visual Studio Code mit den Ressourcen im Cluster verbunden werden. Diese Möglichkeiten bieten sich allerdings wie bereits eingangs erwähnt als Entwicklungsumgebung für Mitarbeitende im Projekt an, die dort Modelle und Programmcode für den Einsatz in der Lehre erstellen und erproben können. So können die verfügbaren GPU- und CPU-Rechenkapazitäten unkompliziert und aus einer Hand sowohl für die Forschung als auch für die Lehre eingesetzt werden.

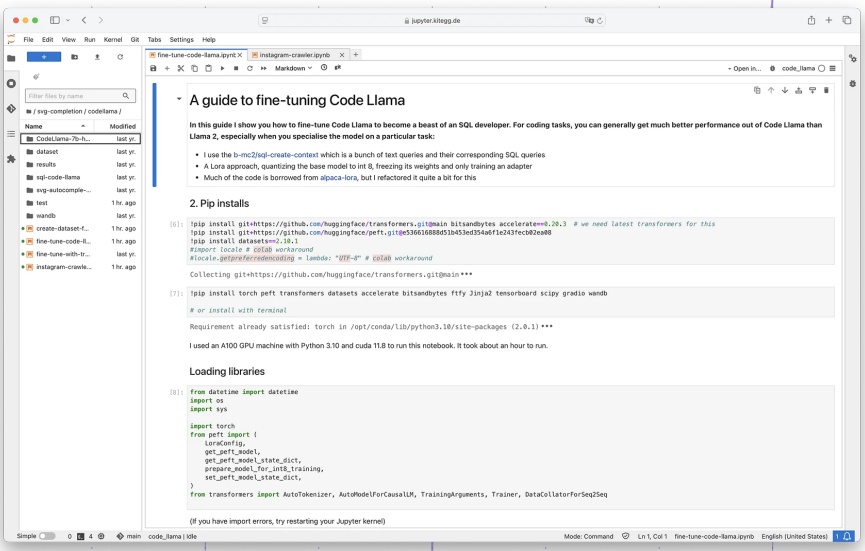

Abb. 5: Benutzeroberfläche einer JupyterLab-Instanz

Chunk 32

Praxis – die Lehr-Lernplattform

Die zentrale Nutzerschnittstelle des Clusters ist die LLP, über die Kursinhalte und Dokumentation bereitgestellt werden, aber auch die Verwaltung von Nutzerkonten, Ressourcenzuweisungen und die Bereitstellung weiterer Dienste realisiert wird. Im Vergleich zu bekannten Learning-Management-Systems (LMS) wie OLAT oder Moodle bedient die KITeGG-LLP einen etwas anders gelagerten Anwendungsfall.

Chunk 33

Da diesem Forschungsprojekt keine explizit vordefinierten Lehrmethoden und damit verbundene Organisationsstrukturen vorausgehen, fungiert die LLP zunächst als Anlaufpunkt für Studierende und Lehrende, um dort mit einem im Verlauf des Projekts stetig wandelnden Angebot von Funktionen, Diensten und Materialien zu interagieren. Darüber hinaus soll sie als zentrales Repositorium für die Dokumentation von Kursen und den Projekten von Studierenden im Verbund dienen. Somit versucht die LLP nicht mit eventuell an den Standorten bereits implementierten LMS zu konkurrieren, sondern vielmehr einen flexiblen experimentellen Rahmen zur Erprobung und Evaluation neuer Lehransätze und Methoden bereitzustellen.

Die Funktionalität der LLP richtet sich sowohl an die Lehrenden, indem sie die Verwaltung von Kursen, Materialien und Konten für Studierende ermöglicht, und bedient Studierende, indem sie den Zugang zu Kursmaterialien, Diensten und GPU-Ressourcen im Cluster bereitstellt (Abb. 6). Über die Werkschau können zudem einzelne Projekte aus den Standorten beispielhaft dokumentiert werden.

Chunk 34

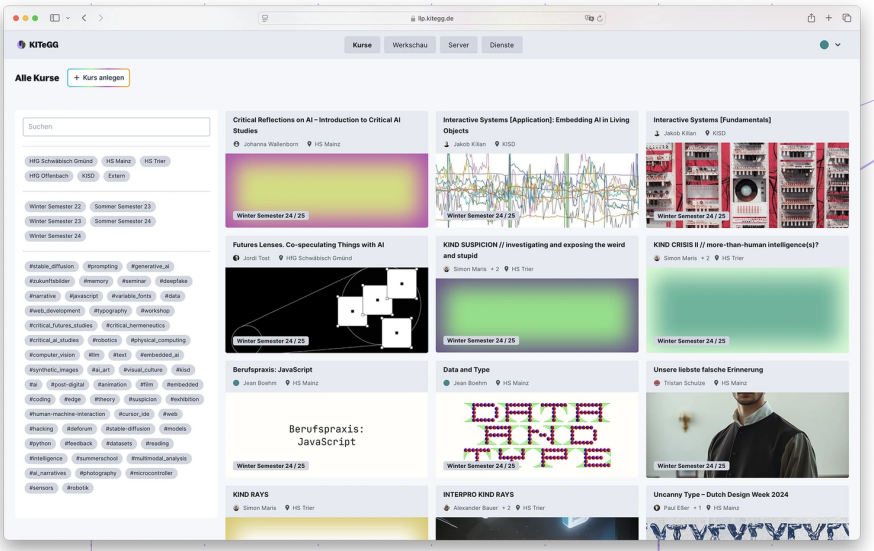

Abb. 6: Kursübersicht in der Lehr-Lernplattform (LLP)

Chunk 35 Ein für alle einsehbarer Kalender (Abb. 7) zeigt die verschiedenen im Cluster verfügbaren Konfigurationen von Grafikprozessoren und wie viele davon belegt oder in nächster Zeit durch Kurse oder Nutzer:innen reserviert sind. Dieses Reservierungssystem fungiert zugleich als verbindliche Garantie für verfügbare Ressourcen zu bestimmten Zeitpunkten und als transparente Kommunikation von Bedarfen der Lehrenden und Studierenden über das Semester hinweg.

Chunk 36 Ebenso können Lehrende eigens angepasste Versionen der JupyterLab-Entwicklungsumgebung einstellen und selektiv für einzelne oder alle Standorte und Nutzergruppen verfügbar machen.

Chunk 37

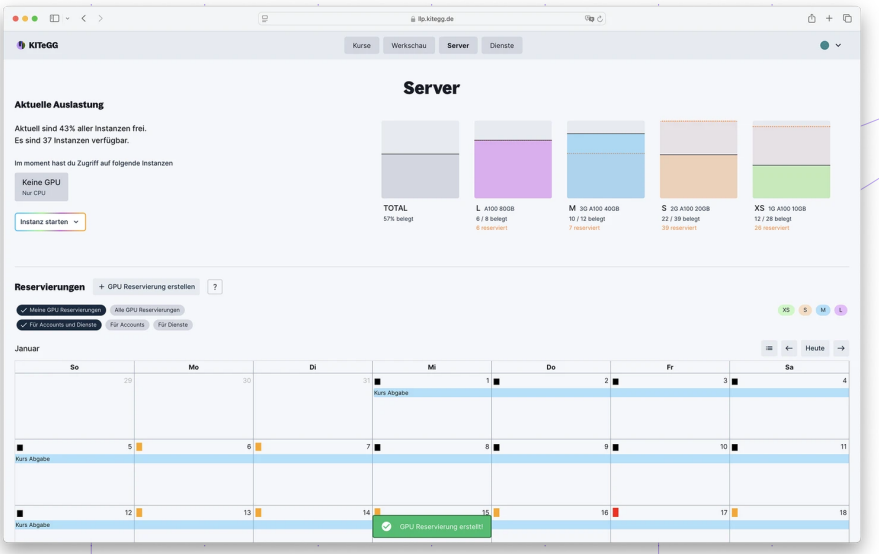

Abb. 7: Kalender für GPU-Reservierungen und Auslastung des Clusters

Chunk 38 Schließlich bietet die Option der „Services“ (Abb. 8) die Möglichkeit, eigene Dienste einzurichten, die als Zusammenstellungen von einem oder mehreren Docker-Containern und virtuellen Laufwerken konfiguriert werden. Diese

können dann intern im Cluster oder auch über das Internet erreichbar gemacht werden und sind entweder für den gesamten Verbund, einzelne Standorte oder einzelne Studierende gedacht.

Chunk 39 Letztere können einen von Lehrenden vorbereiteten Dienst „klonen“ und in einem vorgegebenen Rahmen für ihre Zwecke anpassen. Diese Dienste fungieren als alternativer Zugang zu Modellen, der sich rein auf deren Funktion und praktische Nutzung beschränkt, ohne dabei wie in JupyterLab Programmier- oder Linux-Kenntnisse vorauszusetzen.

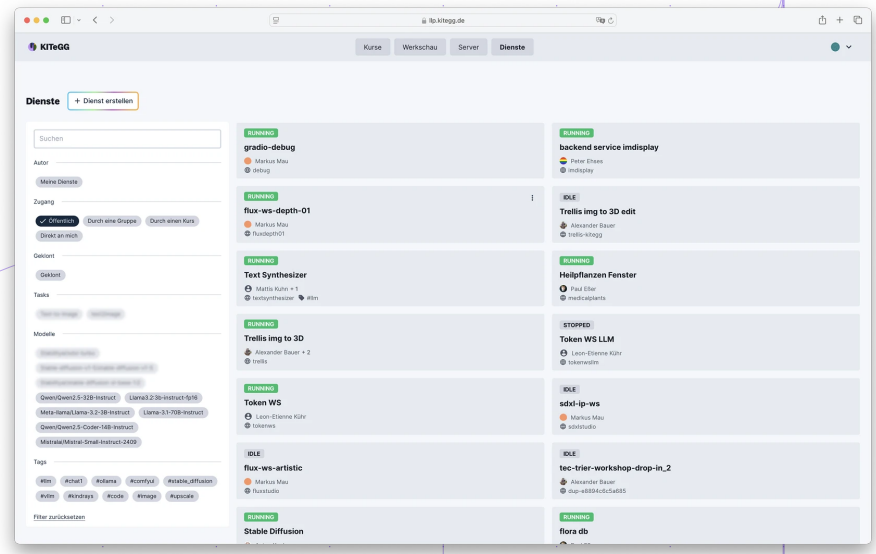

Abb. 8: Übersichtsseite für die verfügbaren Dienste

Chunk 40 Die LLP besteht im Wesentlichen aus einer eigens entwickelten Webanwendung basierend auf VueJS[33] und dem Quasar Framework[34] sowie eines Application Programming Interface (API), die mittels des „Content Management Systems“ (CMS) „Directus“[35] implementiert wird. Die Funktionalität von Directus lässt sich durch Erweiterungsmodule anpassen, über die sich unter anderem Funktionen des Kubernetes-Clusters steuern lassen und mit den anderen Systemkomponenten kommuniziert werden kann.

33: You, E. (2024). *Vue.js: The Progressive JavaScript Framework*. Abgerufen am 20. Dezember 2024 von https://vuejs.org

34: Stoenescu, R. (2024). *Quasar Framework*. Abgerufen am 20. Dezember 2024 von https://quasar.dev

35: Monospace Inc. (2025). *Directus*. Abgerufen am 12. Januar 2025 von https://directus.io

Chunk 41 So kann beispielsweise ein Dienst als Datenmodell in Directus angelegt werden und dann mittels einer Erweiterung innerhalb des Kubernetes Clusters automatisch instanziiert, dessen Logs abgerufen oder auf dessen Container zugegriffen werden. Der Zugang zur LLP und damit auch zu allen weiteren im Cluster instanziierten Dienst wird dabei mittels eines sogenannten „Single-Sign-On“ (SSO), einer singulären und zentralen Identifikationsquelle, gewährleistet, die sich wiederum an bestehende Authentifikationssysteme an den Standorten anbinden lässt.

Herausforderungen und Perspektiven experimentellen Platform-Engineerings

Das Projekt erfordert einen möglichst stabilen und verlässlichen Produktivbetrieb, verfolgt aber gleichzeitig eine offene Entwicklungsstrategie sowohl in Bezug auf das übergeordnete „Platform-Engineering“, also die Schaffung eines möglichst autarken und flexiblen Handlungsrahmens für alle Projektpartner:innen, als auch die kontinuierliche Weiterentwicklung der im Betrieb benötigten Dienste und Anwendungen, deren Anforderungen sich aus dem Produktionsbetrieb ergeben.

Chunk 42 Der grundlegende Betrieb erfordert die regelmäßige Rotation von Zugangsdaten auf den Servern und für die verwendeten Anwendungen, Inspektionen von Backups, Ressourcenverbrauch, Zustand der Hardware- und Softwarekomponenten, sowie Updates und Patches der verwendeten Anwendungen, wobei stets auf mögliche Inkompatibilitäten durch neue Versionen geachtet werden muss.

Chunk 43 Dies geschieht zum Teil während des laufenden Betriebs, erfordert aber auch regelmäßige größere Wartungsarbeiten, die nach Möglichkeit in der vorlesungsfreien Zeit stattfinden, um den laufenden Betrieb nicht zu stören.

Chunk 44 Neue und ungetestete Dienste werden zunächst isoliert in separaten Namespaces getestet und dann nach und nach in den Betrieb integriert. Größere Änderungen an der generellen Funktionsweise des Clusters oder der LLP werden dabei möglichst nur zwischen den Semestern veröffentlicht, um den Lehrenden Gelegenheit für weitere Tests zu geben. Generell ist das Projekt aber als experimentell einzustufen und arbeitet daher mit einer geringeren Verfügbarkeitsgarantie als in kommerziellen Szenarios tolerabel wäre, da sich kurzfristig notwendige Unterbrechungen im Zweifelsfall auf kurzem Wege innerhalb des Verbunds kommunizieren und abstimmen lassen.

Neben der Balance von Betrieb und Entwicklung stellt die Ressourcenverteilung eine weitere zentrale Herausforderung für die Entwicklung des Clusters dar.

Chunk 45 Die verwendeten A100-Grafikprozessoren erlauben mittels der „Multi-Instance GPU“ (MIG)[36] Technologie eine Partitionierung in maximal sieben sogenannte „Slices“, die laufenden Containern zugewiesen werden können, ihrerseits über 10 GB RAM verfügen und ungefähr ein Siebtel der auf dem Prozessor verfügbaren Kerne nutzen können. Bei dieser Partitionierung ergäbe sich für diesen Cluster eine maximale Anzahl von 280 parallel nutzbaren GPU-Einheiten. Alternativ kann jedoch jeder einzelne Grafikprozessor in einer Partitionierung in Segmenten von 10 GB (1/7), 20 GB (1/4), 40 GB (1/2), als gesamte Einheit oder mehrere Einheiten in Kombination genutzt werden. Da unterschiedliche Nutzungsszenarien wie das Ausführen größerer Modelle oder das Training neuer Modelle jeweils andere Anforderungen an verfügbare GPU-Partitionen stellen, muss stets der aktuelle Bedarf im Verbund eruiert und mit entsprechenden Änderungen der Konfiguration darauf reagiert werden, sollten sich Engpässe ergeben.

36: NVIDIA Corporation (2024). *MIG User Guide*. Abgerufen am 20. Dezember 2024 von https://docs.nvidia.com/datacenter/tesla/mig-user-guide/index.html

Chunk 46 Dies stellt insofern ein Problem dar, als dass der von NVIDIA bereitgestellte Open-Source GPU-Operator[37] bei einer Änderung der MIG-Konfiguration in der Regel einen Neustart des Servers erfordert, was einen Stopp aller darauf laufenden Container zur Folge hat.

37: NVIDIA Corporation (2024). *About the NVIDIA GPU Operator*. Abgerufen am 20. Dezember 2024 von https://docs.nvidia.com/datacenter/cloud-native/gpu-operator/latest/overview.html

Eine weitere konzeptionelle Herausforderung stellt die Sicherheit dar, die gerade im Hochschulkontext eine große Rolle spielt.

Chunk 47 Der experimentelle Charakter des Projekts und seine offene Implementierung als Entwicklungsplattform, die gleichzeitig einen Produktivbetrieb ermöglicht, verhindern zumindest teilweise eine standardisierte „Härtung", also die maximale Reduktion der möglichen Funktionalität auf eine Auswahl explizit erlaubter Operationen und die Unterbindung aller übriger sonstigen Netzwerk-, Nutzer- oder Rechenaktivität. Um dennoch ein Mindestmaß an Absicherung innerhalb der bestehenden Hochschulinfrastruktur zu erreichen, wird hier hauptsächlich auf Isolation und Risikominimierung gesetzt, in Verbindung mit einem eingegrenzten Nutzerkreis, innerhalb dessen ein gewisses Maß an Vertrauen herrscht.

Chunk 48 So ist der Cluster komplett vom übrigen Hochschulnetzwerk abgetrennt, nutzt seine eigene Firewall und kommuniziert ausschließlich in einem Umkreisnetzwerk (oft „DMZ" genannt, eine Abkürzung für „demilitarisierte Zone") über das öffentliche Internet. Ebenso werden innerhalb des Clusters bestimmte Anwendungspartitionen innerhalb von Kubernetes über sogenannte „Policies" voneinander isoliert. Im Cluster werden möglichst keine persönlichen Daten bezüglich der Studierenden verarbeitet, abgesehen von Namen und Mailadressen. Die im Cluster durch die Nutzer erzeugten Daten werden insofern als „ephemer" betrachtet, als dass bei der anfallenden Menge eine automatische Sicherung aller Daten nicht sinnvoll ist und daher die Studierenden angehalten sind, ihre wichtigen Daten selbst zu sichern. Die Daten werden allerdings im laufenden Betrieb immer mehrfach redundant gespiegelt, was zumindest eine grundlegende Sicherheit vor Verlust bietet. Die für den Projektbetrieb essenziellen Daten hingegen werden innerhalb und außerhalb des Clusters mehrfach gesichert, um im Ernstfall selbst den gesamten Cluster abschalten, löschen und neu aufsetzen zu können, was sich durch die Art der Konfiguration bei Kubernetes als verhältnismäßig einfach reproduzierbar darstellt. Der Betrieb des Clusters wird mittels gängiger Werkzeuge für Monitoring[38] und Datenvisualisierung[39] überwacht und auf auffälliges Verhalten wie außergewöhnliche Netzwerkaktivität oder exzessive Nutzung der Rechenleistung untersucht, um potenziell schadhaftes Verhalten unter den Nutzern zu identifizieren. Letztere sind zudem innerhalb der ihnen zur Verfügung gestellten interaktiven Umgebungen auch in ihren Möglichkeiten zur Installation von Software eingeschränkt.

38: The Linux Foundation (2024). *What is Prometheus?*. Abgerufen am 20. Dezember 2024 von https://prometheus.io/docs/introduction/overview/

39: GrafanaLabs (2024). *Grafana*. Abgerufen am 20. Dezember 2024 von https://grafana.com/grafana/

III. p. 245, Chunk 30: How to KITeGG
III. p. 240, Chunk 10: How to KITeGG
III. p. 251, Chunk 67: How to KITeGG

Chunk 49 Eine langfristige Herausforderung im Betrieb des Clusters ist der mögliche Ausfall von Komponenten oder deren finale Überalterung („End of Life"), was zu einer zunehmenden Reduktion der verfügbaren Ressourcen führt. Das KITeGG Projekt hat für alle verwendeten Hardwarekomponenten eine Garantie über die fünfjährige Projektlaufzeit vereinbart, jedoch ist ein Weiterbetrieb der Hardware, besonders im Hinblick auf deren Amortisation, für den gesamten

Verbund wünschenswert. Hier ergibt sich allerdings ein personeller und materieller Engpass, da für Betrieb und Wartung des Clusters mindestens eine Vollzeitstelle benötigt wird und ein möglicherweise notwendiger Austausch von Komponenten finanziert werden muss. Letzterer stellt besonders in Bezug auf die NVIDIA-Hardware ein Problem dar, da sich, bedingt durch den gegenwärtigen Wachstumsboom des von diesem Hersteller dominierten Markts für Grafik- und Tensorprozessoren, deren Preise seit Beginn des Projekts nahezu verdoppelt haben, was den Austausch einzelner Prozessoren oder gar komplette Neuanschaffungen der proprietären Servertechnologie HGX massiv erschweren würde. Aufgrund der modularen Konfiguration des Clusters ist hier allerdings auch eine mittel- oder langfristige Hinwendung zu anderen Herstellern von GPU-Hardware denkbar, sofern dies durch eine wachsende Unterstützung seitens der gängigen Softwarepakete begünstigt wird.

Chunk 50 So wäre perspektivisch eine Abkehr von geschlossenen Hard- und Software-Ökosystemen wie NVIDIA und deren HGX- und CUDA-Architektur zugunsten von offeneren Architekturen wie SYCL[40], OpenCL[41] und anderen Herstellern wie AMD oder Intel denkbar.

Fazit

Die KITeGG-Infrastruktur stellt einem Verbund von fünf Hochschulen für Gestaltung einen experimentellen Rahmen für die Erprobung von Lehrkonzepten, um ihren Studierenden und Lehrenden „K.I. greifbar und begreifbar [zu] machen“[42].

Chunk 51 Dabei stützt sich das Projekt auf einen erweiterten Begriff der „Infrastruktur“, der diese nicht als passiven Unterbau, sondern als Ausdruck einer politischen Positionierung versteht, die wiederum in dem durch sie realisierten technologischen Potenzial reproduziert wird.

Chunk 52 Davon leitet sich ein Verständnis von „Künstlicher Intelligenz“ als soziotechnischem Narrativ und komplexem Arrangement bestimmter Strömungen aus Politik, Ökonomie und Gesellschaft ab. Diese Grundannahmen sind prägend für die konkrete Ausgestaltung der technologischen Infrastruktur und der Interpretation der Projektziele, die die gleichzeitige Bereitstellung einer produktiven Lehrplattform und einer offenen und agilen Entwicklungsumgebung vorsehen. Die Entscheidung gegen die Verwendung von Cloud-basierter Servertechnologie und für den Erwerb eigener Hardware erfolgt aufgrund der Verpflichtung einer öffentlichen Lehreinrichtung aus den Angewandten Wissenschaften zur Kultivierung eigener Kompetenzen. Diese sollen auf möglichst vielen Ebenen der zu vermittelnden Technologien ausgebildet werden und verfolgen methodisch eine konsequente Auseinandersetzung mit deren materiellen Bedingungen. Nicht zuletzt zeigt bei dieser Projektlaufzeit auch der Vergleich von „Cloud Sourcing“ und eigenem „Bare-metal“ eine klare Kosteneinsparung durch die Anschaffung eigener Hardware.

Der erweiterte Begriff der Infrastruktur bedeutet für das Projekt eine konzeptionelle Trennung der zugrundeliegenden Theorien und Technologien. Die „theoretische Infrastruktur“ bildet die strategische Grundlage für das Platform-Engineering und Research-Software-Engineering (RSE), indem sie ein

40: Khronos Group (2024). *SYCL: C++ Programming for Heterogeneous Parallel Computing*. Abgerufen am 20. Dezember 2024 von https://www.khronos.org/sycl/

41: Khronos Group (2024). *OpenCL: Open Standard for Parallel Programming of Heterogeneous Systems*. Abgerufen am 20. Dezember 2024 von https://www.khronos.org/opencl/

42: KITeGG (2024). *KITeGG Internetseite*. Abgerufen am 20. Dezember 2024 von https://gestaltung.ai

Verständnis von Software nicht als Lösung oder Produkt propagiert, sondern als Ausdruck einer bestimmten Kultur und Technologiepraxis. Dabei sollte diese besonders im Forschungskontext stets kritisch reflektiert werden und nicht nur auf scheinbar universellen und neutralen „Best Practices“ basieren.

I. p. 37, Chunk 8: autoLab (HS Mainz)
III. p. 258, Chunk 7: KI-TeGG und nun?
II. p. 178, Chunk 7: Editorial

Chunk 53 „Digitalität“ und „Künstliche Intelligenz“ werden dabei als soziopolitische Erzählungen verstanden und sind damit instabile, wandelbare und weder singulär noch exklusiv zu definierende Konzepte. Andererseits bildet die „technologische Infrastruktur“ das duale Szenario von Produktivbetrieb in der Lehre und einem experimentellen Labor für die Evaluation und Entwicklung neuer Software, Methoden und Modelle ab. Dabei stellt sie zwar eine generelle „High Performance Computing“ (HPC) Plattform dar, die jedoch ergebnisoffenes Arbeiten in interaktiven Umgebungen priorisiert und damit eine andere funktionale Konfiguration benötigt als vergleichbare HPC-Infrastrukturen.

Die beiden grundlegenden Stränge von Forschung und Lehre werden jeweils durch eine offene Entwicklungsumgebung mittels der Software JupyterLab und eigenen, als Docker-Container verwalteten Diensten sowie einer eigens entwickelten LLP abgebildet.

Chunk 54 Während erstere den Mitarbeitenden an den Standorten die eigenständige Erprobung von Software und Modellen, wie auch die Entwicklung eigener Dienste und Anwendungen erlaubt, stellt letztere eine grundlegende Verwaltung von Kursen, Materialien und Ressourcen für Studierende bereit.

Zu den wichtigsten Herausforderungen im Verlauf des Projekts zählen die Balance zwischen einem verlässlichen Produktivbetrieb, gleichzeitiger agiler Entwicklung und sehr unterschiedlich gelagerten Priorisierungen an den jeweiligen Standorten. Die dynamische Verteilung benötigter Ressourcen und die transparente Kommunikation der jeweiligen Bedarfe innerhalb des Verbunds werden über ein Reservierungssystem mittels eines geteilten Kalenders realisiert. Die Sicherheit des Systems stellt eine Herausforderung dar, besonders im Kontext einer offenen und agilen Entwicklung. Hier wird zum einen auf eine Kombination von Isolation des Systems selbst von der Hochschulinfrastruktur sowie der einzelnen Komponenten und Arbeitsbereiche innerhalb des Clusters, in Verbindung mit einer Reduktion der Garantien an Verfügbarkeit und Ausfallsicherheit, gesetzt. Dieses Prinzip ist dabei nur in einem experimentellen Rahmen praktikabel und das System müsste für einen erweiterten Produktivbetrieb oder innerhalb eines explizit kommerziellen Kontexts stärker eingeschränkt und abgesichert werden. Schließlich muss eine Perspektive für eine langfristige Nutzung und Wartung der Hardware entwickelt werden, die durch den modularen Aufbau des Systems begünstigt wird, der selbst einen schrittweisen Wechsel zu anderen GPU- und Hardwaresystemen erlauben würde.

Zum Zeitpunkt dieser Veröffentlichung erreicht das KITeGG Projekt seine Endphase zum Abschluss seiner Laufzeit. Dies bedeutet für die Plattform nun weniger Entwicklung neuer Funktionalität und stattdessen die Konsolidierung und umfassende Dokumentation der vorhandenen Komponenten und des gesamten technologischen Aufbaus. Dazu erfordert das Ende der Projektlaufzeit die Stabilisierung und Absicherung der umgesetzten Anwendungen, also eine weniger offene und dafür gleichzeitig weniger wartungsintensive Konfiguration. Dazu werden restriktivere Netzwerk- und Firewall-Regeln eingeführt und die verfügbaren Funktionen auf das Nötigste reduziert, um im Nachgang des Projekts einen zumindest eingeschränkten Weiterbetrieb mit einer möglichst effizienten Wartung des Clusters zu ermöglichen, auch mit eingeschränkten personellen

Ressourcen. Vor allem aber soll das Projekt eine reproduzierbare Fallstudie für den Aufbau und die Entwicklung einer vergleichbaren Plattform hervorbringen, gemeinsam mit den dafür entwickelten Anwendungen. Analog zur Dualität einer theoretisch-technologischen Infrastruktur ist das Ergebnis dieser experimentellen RSE- und Platform-Engineering-Prozesse damit ein praktisch und methodisch reproduzierbares Artefakt in Form einer nachvollziehbaren Schritt-für-Schritt-Anleitung sowie aller benötigten Ressourcen und Code-Repositorien, jedoch in Verbindung mit einer kritischen Reflexion der getroffenen Entscheidungen und der konzeptionellen Herleitung der zugrundeliegenden Annahmen und Paradigmen.

Es bleibt weiterhin diskutabel, ob eine breite Abhängigkeit von der spezifischen Konfiguration unserer technologischen Infrastruktur als KI langfristig wirklich sinnvoll und tragbar ist, gleichwohl kann sich dieser Frage auf institutioneller Ebene nur erschöpfend und in der nötigen Tiefe genähert werden, indem diese Annäherung außerhalb der geschäftlichen Interessen kommerzieller Dienstleister stattfindet.

Chunk 55 Somit steht diese Fallstudie auch explizit für eine politische Positionierung, die sich gegen eine Monopolisierung von technologischen Kompetenzen, kritischer Infrastruktur und funktionalen Paradigmen in den Händen weniger Konzerne stellt.

Chunk 56 Dabei geht es einerseits um arbeitsrechtliche, ethische und rechtliche Implikationen einer Abhängigkeit von global operierenden, jedoch lokal nur beschränkt rechenschaftspflichtigen Firmen mit weitestgehend intransparenten geschäftlichen Strukturen. Darüber hinaus jedoch sollten gerade in diesem institutionellen Bereich öffentliche Gelder zuerst als lokale Investitionen in den Aufbau eigener Kapazitäten, den nötigen personellen Kompetenzen und in die Förderung eigenständiger Wissensproduktion und -vermittlung fließen. Die in diesem Sinne verwendeten Mittel reduzieren nicht nur unmittelbare Kosten innerhalb des Projekts, sondern schaffen zudem nichtmonetäre Mehrwerte, die nachhaltig eine souveräne und diverse Digitalkultur und -landschaft fördern, indem sie in der Hochschulbildung eine tiefgreifende Kompetenzbildung in der Aneignung anstatt nur in der Anwendung emergenter Technologien vorantreiben.

Dieser Artikel wurde zusätzlich separat publiziert als ↗ 10.25358/open-science-11844.