

Konzeption einer flexiblen Systemarchitektur  
zur dynamischen Konfiguration  
hybrider Lernräume

MASTERARBEIT  
ausgearbeitet von  
Dennis Dubbert

zur Erlangung des akademischen Grades  
MASTER OF SCIENCE (M.Sc.)

vorgelegt an der  
TECHNISCHE HOCHSCHULE KÖLN  
CAMPUS GUMMERSBACH  
FAKULTÄT FÜR INFORMATIK UND  
INGENIEURWISSENSCHAFTEN

im Studiengang  
MEDIENINFORMATIK (MASTER)

Erster Prüfer: Prof. Dr. Christian Kohls  
Technische Hochschule Köln

Zweiter Prüfer: Alexander Dobrynin, M.Sc.  
Technische Hochschule Köln

Gummersbach, im August 2021

---

**Technology**  
**Arts Sciences**  
**TH Köln**

**Adressen:** Dennis Dubbert  
ORCID:  
<https://orcid.org/0000-0002-6624-170X>  
dennis.dubbert@th-koeln.de

Prof. Dr. Christian Kohls  
Technische Hochschule Köln  
Cologne Institute for Digital Ecosystems  
Steinmüllerallee 1  
51643 Gummersbach  
christian.kohls@th-koeln.de

Alexander Dobrynin, M.Sc.  
Technische Hochschule Köln  
Institut für Informatik  
Steinmüllerallee 1  
51643 Gummersbach  
alexander.dobrynin@th-koeln.de

## Kurzfassung

In dieser Arbeit wird eine dynamische Systemarchitektur für die Schaffung hybrider Ökosysteme konzipiert, welche eine flexible und kontextgerechte Konfiguration hybrider Lernräume ermöglichen. Aufgrund der variierenden Definitionen hybrider Lehre und Lernräume sowie der schwachen Abgrenzung zu verwandten Begriffen findet hierfür zunächst eine Konkretisierung eigener Arbeitsdefinitionen statt. Aufbauend auf diesen Begriffsbestimmungen und ausführlichen Literaturrecherchen, werden anschließend sich interdisziplinär überschneidende Kernelemente hybrider Lehre und Lernräume identifiziert sowie in einem Übersichtsmodell zusammengefasst. Durch die Literaturrecherche ersichtliche Charakteristiken des hybriden Paradigmas werden ebenso in Beziehung gesetzt und zusammenfassend aufgeführt.

Auf Basis dieser Übersichten und unter Berücksichtigung weiterer Forschungsergebnisse wird nachfolgend ein Vorgehensmodell kreiert, welches kontextgerechte Designs hybrider Systeme strukturiert ermöglicht. Neben didaktischen, physischen und digitalen Komponenten werden dafür diverse Typen von Lernaktivitäten identifiziert, welche als Ausgangspunkt eine effiziente und praxisorientierte Planung ermöglichen.

Alle erstellten Modelle bilden abschließend die Grundlage der konzipierten Systemarchitektur, in welcher die hybriden Kernkonzepte aus einer technischen Perspektive beleuchtet werden. Das hierdurch repräsentierte Microservice-Websystem beinhaltet neben notwendigen Basis-Services vor allem einen dynamischen Plugin-Mechanismus, welcher die flexible Integration neuer Bestandteile ermöglicht. So kann der Funktionsumfang auf sich wandelnde Anforderungen der Nutzer angepasst werden und die Anwendung mit der Zeit zu einem umfassenden Ökosystem heranwachsen.

## Abstract

In this thesis, a dynamic system architecture for the creation of hybrid ecosystems is designed, which enables a flexible and context-appropriate configuration of hybrid learning spaces. Due to the varying definitions of hybrid learning and hybrid learning spaces as well as the weak demarcation to related terms, a concretization of own working definitions takes place first. Based on these and an extensive literature research, interdisciplinary overlapping core elements of hybrid learning and hybrid learning spaces are identified and summarized in an overview model. Characteristics of the hybrid paradigm evident through the literature review are also related and summarized.

Based on these overviews and considering further research results, a process model is created, which enables a context-appropriate design of hybrid systems in a structured way. In addition to didactic, physical and digital components, various types of learning activities are identified for this purpose, which as a starting point enable efficient and practice-oriented planning.

Finally, all the models created form the basis of the conceived system architecture, in which the hybrid core concepts are illuminated from a technical perspective. In addition to the necessary basic services, the microservice Web system represented by this architecture contains, above all, a dynamic plug-in mechanism that enables the flexible integration of new components. In this way, the range of functions can be adapted to changing user requirements and the application can grow into a comprehensive ecosystem over time.

# Inhaltsverzeichnis

<b>1. Einleitung</b>	<b>7</b>
1.1. Motivation . . . . .	7
1.2. Relevanz . . . . .	9
1.3. Zielsetzung & Forschungsfragen . . . . .	12
1.4. Vorgehensweise . . . . .	13
1.5. Definitionen hybrider Lehre . . . . .	15
1.5.1. Blended Learning . . . . .	15
1.5.2. Gemischte Terminologien . . . . .	17
1.5.3. Hybrid Pedagogy . . . . .	20
1.5.4. Internationaler Vergleich des Hybrid und Blended Learnings . . . . .	21
1.5.5. Seamless Learning . . . . .	22
1.5.6. Arbeitsdefinition . . . . .	24
1.6. Definitionen hybrider Lernräume . . . . .	25
1.6.1. Blended Spaces . . . . .	26
1.6.2. Hybrid Learning Spaces (HLS) . . . . .	27
1.6.3. Seamless Learning Environments . . . . .	31
1.6.4. Arbeitsdefinition . . . . .	32
<b>2. Eigenschaften und Entitäten hybrider Lehre</b>	<b>35</b>
2.1. Grundlegende Modelle . . . . .	35
2.2. Dichotomien hybrider Lehre . . . . .	39
2.3. Dimensionen als Systemeigenschaften und -entitäten . . . . .	40
2.4. Identifikation gemeinsamer Kernaspekte . . . . .	49
2.4.1. Übersicht identifizierter Entitäten . . . . .	49
2.4.2. Übersicht identifizierter Eigenschaften . . . . .	56
<b>3. Framework zur Konzeption hybrider Systeme</b>	<b>61</b>
3.1. Aktivitäten als Zentrum der Raumplanung . . . . .	61
3.2. Detaillierte Betrachtung der Lehrformen, Methoden und Aktivitäten . . . . .	65
3.3. Übersichtsmodell notwendiger Designschritte . . . . .	71
3.3.1. Lehr- / Lernszenario definieren . . . . .	71
3.3.2. Planung des Lernprozesses, benötigter Arbeitsschritte und beteiligter Aktivitäten . . . . .	73
3.3.3. Erkundung digitaler Räume . . . . .	74
3.3.4. Betrachtung physischer Räume . . . . .	75
3.3.5. Integration von Ökosystemen und privaten Medienökologien . . . . .	77
3.3.6. Vollständiges Modell & hilfreiche Fragen beim Designprozess . . . . .	78

<b>4. Entwicklung eines hybriden Ökosystems</b>	<b>80</b>
4.1. Domänen-Modell der Bestandteile . . . . .	80
4.1.1. A: Akteurverwaltung . . . . .	81
4.1.2. B: Infrastruktur des Lernens . . . . .	84
4.1.3. C: Hintergrundprozesse . . . . .	85
4.1.4. D: Lernaaktivitäten und -ressourcen . . . . .	86
4.2. Anwendungstyp und Architekturstil . . . . .	88
4.2.1. Grundlagen des Microservice-Ansatzes . . . . .	89
4.2.2. Auswahl kritischer Technologien . . . . .	91
4.3. Architektur des Systems . . . . .	97
4.3.1. Zentrales Grundsystem . . . . .	98
4.3.2. Bereitstellung der Services . . . . .	102
4.3.3. Einbindung von Plugins . . . . .	104
<b>5. Fazit &amp; Ausblick</b>	<b>109</b>
5.1. Fazit . . . . .	109
5.2. Ausblick . . . . .	112
<b>Abbildungsverzeichnis</b>	<b>114</b>
<b>Tabellenverzeichnis</b>	<b>115</b>
<b>Literaturverzeichnis</b>	<b>121</b>
<b>ANHANG</b>	<b>122</b>
<b>A. Dichotomien für das Design hybrider Lehrveranstaltungen</b>	<b>123</b>
A.1. Dichotomien des Blended Learnings . . . . .	123
A.2. Dichotomien der Hybrid Pedagogy . . . . .	124
<b>B. Beispielhafte Fragen zur Ermittlung notwendiger Aktivitäten</b>	<b>125</b>
<b>C. Zu beantwortende Fragen beim Design hybrider Systeme</b>	<b>127</b>
<b>D. Wichtige Bestandteile einer Microservice-Architektur</b>	<b>130</b>
D.0.1. Service . . . . .	130
D.0.2. Datenbank . . . . .	132
D.0.3. Kommunikation . . . . .	134
D.0.4. Service Discovery . . . . .	135
D.0.5. API Gateway . . . . .	137
D.0.6. Micro Frontend . . . . .	138
D.0.7. Sicherheit . . . . .	139

# 1. Einleitung

Hybride Lehre ist heutzutage im Alltag und den Medien ein gängiger Begriff, welcher jedoch von vielen unterschiedlich verstanden wird. Einige sehen bereits reine Online-Veranstaltungen als Hybride, da sie stark von der Präsenzlehre abweichen. Andere beschreiben die Hybridität als Mischform aus physischer und digitaler Präsenzlehre und wieder andere beschreiben weitaus mehr charakterisierende Dimensionen wie beispielsweise zeitliche Faktoren. Eines haben all diese Auffassungen jedoch gemeinsam. Ihre Umsetzung erfordert eine technische Unterstützung. Anwendungen für den Kommunikationsaufbau, Dokumentationen, kollaborative Aufgaben, Prüfungen, Seminarvorträge, Feedback und vieles mehr sind heutzutage im Einsatz. Der technische Fortschritt in diesem Feld ist demnach ein hochaktuelles Forschungsgebiet geworden, um hybride Lehre und somit die Ausbildung von Lernenden zu verbessern. Diese Thesis fokussiert sich auf diesen Bereich. Ihr Ziel ist die Identifikation einer geeigneten und flexiblen Systemarchitektur, welche die Erstellung kontextgerechter Lehr- und Lernräume ermöglicht. Dabei wird, neben der Einbindung theoretischer Grundlagen, ein größtmöglicher Praxisbezug von Ergebnissen angestrebt.

Zunächst wird in diesem Kapitel die Motivation zur Wahl des Themas erläutert, gefolgt von dessen Relevanz für die heutige Gesellschaft und Forschungsgemeinde. Daraufhin findet sich eine detaillierte Zielsetzung, in welcher neben den gewünschten Zielen auch die zentralen Forschungsfragen aufgeführt werden. In einer abschließenden Vorgehensbeschreibung werden zudem die wichtigsten Schritte der Zielerreichung kurz erläutert.

## 1.1. Motivation

Die vorherrschende Corona-Pandemie führte zu starken familiären, wirtschaftlichen und sozialen Veränderungen auf der ganzen Welt. Die Menschheit musste gewohnte Muster ablegen und sich täglich neuen Herausforderungen stellen. Auch didaktisch fand ein Wandel der Lehre statt. Galten zunächst physische Präsenzveranstaltungen als Regelfall, so musste nun nahezu sämtliche Lehre auf eine Online-Variante umgestellt werden. Diese Umstellung betitelte Gabi Reinmann [2021] als einen *Online-Schock*, da Lehrende und Lernende gleichermaßen mit den völlig neuen Herausforderungen

der Online-Veranstaltungen konfrontiert waren. Didaktische Vorlesungsgestaltungen, begrenzte Interaktionsmöglichkeiten, eine flexible Anpassung der Veranstaltungen an neue Pandemie-Gesetze, Prüfungsplanungen und die Verteilung von Lernmaterialien sind nur einige Beispiele solcher neuen Hürden. Besonders die technischen Aspekte stellten vielerorts einen noch unbekanntem Themenkomplex dar. So galten die ersten Semester und Schuljahre als unumgängliche Versuchsphasen, in welchen die Möglichkeiten, aber auch Grenzen unterschiedlicher Softwareprodukte ergründet und Unterrichtsformen auf diese abgestimmt wurden. Nicht selten standen Videokonferenz-Programme wie Zoom<sup>1</sup> dabei im Vordergrund. Lehrveranstaltungen sind jedoch vielfältig und mehrdimensional, sodass auch ein Toolset mit variierendem und vielschichtigem Funktionsumfang notwendig ist, um sämtliche Inhalte abzudecken. So bestand die Lehre meist aus einer Zusammenstellung unterschiedlicher Anwendungen, teils bekannt und teils völlig neu, deren Orchestrierung einen zusätzlichen Aufwand darstellte.

Mittlerweile haben sich einige Online-Praktiken bewährt und gefestigt. Positive und negative Erfahrungen führten zu Änderungen und Anpassungen des Lehrprogramms. Auch Lernende haben sich an die neuen Paradigmen und Technologien gewöhnt und diese in ihren täglichen Ablauf integriert. Sie wissen dessen Vorteile zu nutzen, sodass beispielsweise Gruppenarbeiten auch ohne physischen Kontakt möglich sind. Doch finden sich immer noch Hürden, welche eine reine Online-Lehre nicht überbrücken kann [Capra, 2011]. Besonders Kollaborationen mehrerer Personen erfordern häufig eine stärkere Organisation und längere Vorbereitungsphasen als es in der physischen Präsenz notwendig gewesen wäre. Allgemein erscheint zudem die Gruppendynamik häufig als träge und distanziert, da beispielsweise Emotionen über Gestik oder Mimik im Online-Raum schwieriger signalisiert werden können. Diese Aspekte und die voranschreitende Immunisierung der Gesellschaft durch Impfungen rufen die Frage hervor, welche Unterrichtsformen nach der Pandemie genutzt werden sollten. Es könnte wieder in eine reine Präsenzlehre übergegangen werden, jedoch sind nun auch viele Vorteile der Online-Lehre ersichtlich. Der Gesellschaftswandel und die veränderten Denkstrukturen unterstreichen zudem, dass eine neue Form der Lehre sinnvoll sein könnte.

In den Medien werden deshalb oft gemischte Unterrichtsformen genannt. Eine Lehre, welche sowohl auf der physischen, als auch der Online-Präsenz aufbaut. Diese Kombination wird häufig als *hybride Lehre* bzw. *hybrid learning* betitelt und aktuell als das zentrale Ziel vieler Lehrplanänderungen gesetzt.

---

<sup>1</sup>Zoom: Siehe <https://zoom.us/>, zuletzt gesichtet 23.08.2021



## 1.2. Relevanz

Nun stellt sich jedoch die Frage, warum diesem neuen Lehrparadigma solch eine Relevanz zugeschrieben wird und warum sich die Forschung damit befassen sollte.

Als erster wichtiger Aspekt gilt, dass sowohl Online-, als auch Präsenzlehre einige Nachteile aufweisen. Reine Online-Veranstaltungen erfordern beispielsweise eine stärkere Strukturierung und Planung, sodass Lehrende mehr Zeit für die Unterrichtsvorbereitung aufbringen müssen. Ebenso führen solche Veranstaltungen häufiger zu Motivationsproblemen bei Lernenden, besonders wenn diesen keine festen Vorgaben gegeben werden oder sie mit den technischen Anforderungen überfordert sind. Physische Präsenzlehre hingegen ist besonders abhängig von den physischen Gegebenheiten. Teilnehmende müssen sich an einem Ort versammeln, wodurch eine Benachteiligung einzelner Personen entstehen kann. Dies ist beispielsweise der Fall, wenn Lernende weit entfernt wohnen, die Verkehrsanbindung Probleme aufwirft oder sie eine körperliche Behinderung aufweisen und die Lehrräume nicht barrierefrei sind. In hybriden Lehrveranstaltungen wird nun das Potential gesehen, die Nachteile der einzelnen Paradigmen mit den Vorteilen der jeweils anderen auszugleichen und somit eine Verbesserung der Lehre als Ganzes zu erzielen [Zhang, 2008, S. 253]. Schon vor einigen Jahren wurden erste Ansätze für hybride Lehre veröffentlicht. Zhang [2008, S. 253] benannte bereits 2008 viele Vorteile dieses Ansatzes im Vergleich zu der herkömmlichen Präsenzlehre, beispielsweise ein stärkeren Fokus auf die Bedürfnisse der Lernenden und eine höhere Kosteneffizienz. Weiterhin erkannten Norberg et al. [vgl. 2011, S. 6 f.], dass gemischte Kurse eine höhere Erfolgs- und niedrigere Rücktrittsquote aufwiesen, als alleiniger Face-to-Face- oder Online-Unterricht. Ihrer Forschung folgend zeigten Studenten durch dieses neue Format den höchsten Grad der Zufriedenheit, sodass bei offenen Wahlmöglichkeiten vor allem die gemischte Lehre präferiert wurde. Ebenso verhielt es sich mit dem Lehrkörper, welcher aufgrund der gesteigerten Anzahl und Qualität von Interaktionen mit Lernenden einen ebenso hohen Zufriedenheitsgrad bekräftigte. Durch solche Vorteile und eine zeitliche Aktualität des Begriffs vermutete Reinmann [2021, S. 1] bereits, dass „der Grundgedanke [...] ein grundsätzliches Leitprinzip in der Gestaltung von Hochschullehre wird“.

Doch ist noch kein einheitlicher Ansatz für solch einen didaktischen Wandel ersichtlich. Im Gegenteil, es finden sich mannigfaltige und divergierende Definitionen und Ansätze dieser Lehrform, weshalb ein Bedarf nach uniformen Modellen und damit einhergehenden technischen Lösungen herrscht. Es wurden zwar über die Jahre einige Modelle in unterschiedlichen Artikeln veröffentlicht, jedoch begrenzen sich diese größ-

tenteils auf eine theoretische Ebene. Durch den starken Theoriebezug fehlen immer noch weitreichende Tests hybrider Veranstaltungen in natürlichen Umgebungen. Dies ist problematisch, da Unterschiede zwischen der Theorie und praktischen Anforderungen gravierend sein können. Auch wenn bereits einzelne Bestandteile für die aktuelle Online-Lehre eingesetzt werden, so kann aus dem isolierten Gebrauch beispielsweise keine zuverlässige Aussage über ihre kombinierte Nutzung in einem Hybriden getätigt werden. Über technische und lernraumbedingte Anforderungen ist also noch wenig bekannt, abgesehen davon, dass diese einen großen Mehraufwand für Lehrende und Lernende darstellen.

Ein weiteres Problem aktueller Theorien ist, dass sie oft versuchen, ein allumfassendes Bild des Themenkomplexes darzustellen. Die Größe des vorliegenden Themas ist jedoch enorm, nicht zuletzt, da es einen starken Kontextbezug aufweist. Ergründende Modelle umfassen deshalb zahlreiche Dimensionen, welche für eine Überführung in die Praxis zu komplex erscheinen. Wird hingegen zu stark abstrahiert, könnten kontextbedingt wichtige Aspekte zugunsten der Allgemeingültigkeit verworfen werden. Es ist also relevant, keinen reinen Theoriebezug zu wählen, sondern Erfahrungswerte und Praxiswissen zu beachten.

Besonders zu berücksichtigen ist zudem der rasant voranschreitende technische Wandel. Neue Technologien bergen Verbesserungspotential für die Lehre, sei es durch die Ermöglichung neuer Lehrmethoden und -paradigmen, die Erweiterung funktionaler Perspektiven, eine Kombination bestehender Anwendungen oder die Beseitigung vorheriger Hürden und Nachteile. Vor allem ein Feld wie hybride Lehre, welches zu großen Teilen durch dessen Onlinebezug charakterisiert wird, profitiert besonders von diesem Fortschritt. Aktuell finden sich hier zwar einige Teillösungen, ein umfassendes und zusammenhängendes Software-Ökosystem ist jedoch vorerst nicht aufzufinden. Im Gegenteil, diese Anwendungen stellen oft in sich geschlossene Systeme dar, welche keinerlei Einbindungsmöglichkeiten für Dritte in Form von *APIs* bzw. *Open Source Code* anbieten oder diese stark eingrenzen. Ein gutes Beispiel hierfür sind *Learning Management Systeme (LMS)*, welche häufig auf die Bedürfnisse einzelner Institutionen abgestimmt sind oder einem statischen und monolithischen Architekturstil folgen und somit eine Anpassung auf spezielle bzw. wandelnde Bedürfnisse erschweren oder unmöglich machen. Hier würden Fortschritte dynamischer Ökosysteme besonders für die Qualität, Nutzbarkeit und Verbreitung des hybriden Ansatzes einen hohen Gewinn darstellen, nicht zuletzt, da eine der größten Hürden momentan der Verwaltungsaufwand darstellt. Für aktuelle Praktiken der Online-Lehre werden mehrere Anwendun-

gen parallel benötigt, unter anderem Zoom für Videokonferenzen, Google Docs<sup>2</sup> für eine kollaborative Erarbeitung von Dokumenten, Mentimeter<sup>3</sup> und Miro<sup>4</sup> für Brainstorming-Prozesse, Youtube<sup>5</sup> zur asynchronen Wissensübermittlung und vieles mehr. Dieses neue Paradigma verlangt viel von Lehrenden und Lernenden, vor allem, da solche Anwendungssammlungen keine optimale Funktionalität bereitstellen und eher als notwendiger Kompromiss zu sehen sind. Ihre Zusammenstellung und damit verbundenen Recherchen stellen einen unverhältnismäßig hohen Zusatzaufwand dar. Sie sind nicht optimal aufeinander abgestimmt, die Übermittlung der Inhalte zwischen den Anwendungen ist mit viel manuellem Aufwand verbunden und selbst die Recherche dieser Anwendungen nimmt bereits aufgrund fehlender Erfahrungen viel Zeit in Anspruch. Weiterhin mögen diese Anwendungen für ein Lehrszenario funktionieren, für ein Selbststudium oder kollaborative Aktivitäten können sie jedoch unpraktisch sein. Hierfür werden dann alternative Anwendungen herangezogen, wodurch das Ökosystem verwendeter Softwareprodukte weiterhin steigt und auch der damit verbundene Verwaltungs- und Lernaufwand wächst. Die Umfrageergebnisse von Alqudah et al. [2020] unterstreichen dies. Als größte Limitationen nannten sie die schlechte Infrastruktur bestehender Lösungen, fehlende Erfahrungen, sowie den damit verbundene Mehraufwand.

Es fehlt also ein geordnetes und übergreifendes Konzept, welches die Kompatibilität von Lehr- und Lernaktivitäten herstellt, sowie die Kommunikation aller verwendeter Anwendungen innerhalb einer bekannten Anlaufstelle ermöglicht. Diese Aspekte sind besonders in Bezug auf *Personal Learning Environments* problematisch, welche häufig im Zusammenhang mit E-Learning angesprochen werden. In diesen erhält ein Lernender die Möglichkeit, den eigenen Lernprozess bzw. die eigene Lernumgebung zu gestalten und sie so an eigene Bedürfnisse anzupassen [vgl. Van Harmelen, 2006, S. 1]. Hier wäre nun eine umfassende und flexible Anlaufstelle zur Erstellung und Verwaltung notwendiger Teilanwendungen sinnvoll. Die Schaffung eines grundlegenden Frameworks, auf dessen Basis die Entwicklung kontextgerechter Lösungen vereinfacht wird, stellt dafür einen ersten, notwendigen Schritt dar. Darauf aufbauende Teillösungen könnten dann aufgrund des gemeinsamen Kerns mit der Zeit zu einem großen und umfangreichen Ökosystem zusammenwachsen. Eine Voraussetzung dafür ist jedoch, dass der Kern keinem monolithischen Gedanken folgt und vielmehr eine dynamische und verteilte Erweiterbarkeit aufweist.

---

<sup>2</sup>Google Docs: Siehe <https://docs.google.com/>, zuletzt gesichtet 23.08.2021

<sup>3</sup>Mentimeter: Siehe <https://www.mentimeter.com/>, zuletzt gesichtet 23.08.2021

<sup>4</sup>Miro: Siehe <https://miro.com/>, zuletzt gesichtet 23.08.2021

<sup>5</sup>Youtube: Siehe <https://www.youtube.com/>, zuletzt gesichtet 23.08.2021

Letztlich ist auch der zeitliche Aspekt zu beachten. In den vergangenen Monaten wurden viele Erfahrungen mit alternativen Lehrszenarien und Technologien gewonnen. So finden sich zahlreiche aktuelle Umfragen und Studien, welche die Erkenntnisse aus unterschiedlichen Praktiken dokumentieren. Durch die voranschreitende Immunisierung der Gesellschaft gegenüber dem COVID-19-Virus ist zudem die Frage nach dem *Danach* präsent. Wie wird Lehre zukünftig fortgeführt? Der Mensch neigt dazu, Gewohnheiten zu präferieren, sodass starke Anreize für einen Wechsel notwendig sind und die Bereitschaft dazu stetig sinkt [vgl. Duhigg, 2012]. Dabei hat die Stabilität des Umfeldes und aktuellen Kontextes laut Danner et al. [2008] eine große Auswirkung auf Verhaltensänderungen. Neue oder selten frequentierte Kontexte fördern vorsätzliche Handlungen und reduzieren jene auf Basis einer Gewohnheit. Der pandemiebedingte Kontextwechsel kann demnach aktuell als großer Motivator zum Ablegen vorheriger Gewohnheiten beitragen, sodass die Offenheit gegenüber neuen Paradigmen gesteigert ist. Kehrt man jedoch irgendwann zu dem altbewährten Modell der Präsenzlehre zurück, so ist die Chance auf eine dauerhafte Veränderung weitaus geringer. Für die großflächige Einführung einer neuen Art von Lehre, wie dem hybriden Ansatz, bleibt demnach nur ein kurzes Zeitfenster. So müssen für eine größtmögliche Akzeptanz zeitnah Modelle und Technologien entwickelt werden, welche eine solide Grundlage der Umsetzung darstellen können.

### **1.3. Zielsetzung & Forschungsfragen**

Mit diesem Problem befasst sich nun die vorliegende Arbeit. Es wird untersucht, welche zentralen Bestandteile eine hybride Lehre beinhaltet. Ein besonderer Fokus wird dabei auf die dynamische Erstellung angepasster Lernräume gelegt, welche ad-hoc für den jeweiligen Kontext konfigurierbar sein sollen. Das Ziel ist somit, neben der Spezifikation eines themenspezifischen Modells, vor allem die Identifikation der Designkriterien. Es soll ein Vorgehensmodell entstehen, mit welchem eine kontextgerechte Planung hybrider Lernräume strukturiert erfolgen kann. Hierdurch wird eine Beschleunigung des Designprozesses, sowie eine Steigerung der Qualität resultierender Räume angestrebt. Das finale Ziel ist dann die Schaffung einer flexiblen Systemarchitektur, auf deren Basis die Erstellung kontextgerechter hybrider Lernräume vereinfacht und vereinheitlicht wird, sodass mit der Zeit ein hybrides Ökosystems heranwachsen kann.

Zwei der wichtigsten Punkte des zukünftigen Systems sind dessen Flexibilität und Erweiterbarkeit. Wie zuvor angedeutet, ist hybride Lehre ein umfassendes Themenfeld. Alle Aspekte detailliert zu berücksichtigen, kann zu einem komplexen Konstrukt führen, welches für den alltäglichen Gebrauch ungeeignet ist oder eine schnelle Adaption

auf eigene Kontexte verhindert. Die Systemarchitektur sollte demnach eine grundlegende und abstrahierte Basis schaffen, mithilfe derer kontextbedingt notwendige Funktionalitäten ad-hoc eingebunden werden können. Sie darf jedoch nicht zu abstrakt sein, um dennoch eine Überführbarkeit und Kompatibilität dynamisch hinzugefügter Bestandteile zu gewährleisten. So muss auch eine Integration vorhandener, technischer Mittel in hybride Lehrveranstaltungen mithilfe dieses Systems möglich sein.

Zusammenfassend sollen die Ergebnisse dieser Arbeit in Kombination den hybriden Gedanken festigen, praktische Hinweise für eine didaktische Planung fachgerechter Hybrid-Lehre bieten, sowie die technische Umsetzung geplanter Lehrveranstaltungen unterstützen. Hierfür ist zunächst eine theoretisch fundierte Ergründung des Themenkomplexes erforderlich, bei der drei Forschungsfragen eine zentrale Rolle einnehmen. Diese lauten wie folgt:

1. Was sind die Kerneigenschaften und Bestandteile hybrider Lehre und hybrider Lernräume?
2. Welches Vorgehensmodell ermöglicht einen strukturierten, zielorientierten und kontextgerechten Designprozess hybrider Lernräume?
3. Welche technische Systemarchitektur kann einem hybriden Ökosystem zugrunde liegen, sodass dieses eine dynamische Konfiguration hybrider Lernräume ermöglicht und zeitgleich flexibel auf sich ändernde Kontextbedingungen eingehen kann?

Bei der Beantwortung dieser Forschungsfragen ist zudem eine praxisorientierte Sichtweise essentiell. Entstehende Konzepte sollen nicht nur die Forschungsgemeinde vorantreiben, sondern vor allem in realen Szenarien zum Einsatz kommen.

## **1.4. Vorgehensweise**

Die genannten Forschungsfragen stellen bereits eine Hierarchie dar, sodass ihre Beantwortung in der dargestellten Reihenfolge erfolgen muss. Bevor die Kerneigenschaften und Bestandteile hybrider Lehre und Lernräume identifiziert werden können, ist zunächst die zentrale Bedeutung dieser Begriffe zu klären, was in Kapitel 1.4 geschieht. Dabei werden zunächst bestehende Definitionen analysiert und mit denen verwandter Begriffe verglichen, um anschließend Gemeinsamkeiten und Abgrenzungen in eigenen Arbeitsdefinitionen festzuhalten.

Aufbauend auf diesen grundlegenden Begriffsdefinitionen beschäftigt sich Kapitel 2 mit der ersten Forschungsfrage. Hier werden deshalb bestehende Modelle auf unterschiedlichen Abstraktionsebenen und aus unterschiedlichen Sichtweisen analysiert. Ziel der Analysen ist die Identifikation übergreifender Kernelemente und beschreibender Eigenschaften hybrider Lehre, sowie die Darstellung ihrer Beziehungen und Konstellationen in Übersichtsmodellen, welche ein tieferes Verständnis der Thematik ermöglichen.

Mithilfe dieser Modelle wird anschließend in Kapitel 3 die zweite Forschungsfrage beantwortet. So widmet es sich der Erarbeitung eines Frameworks für das Design hybrider Lernräume und Systeme. Dabei wird auf Basis erprobter Design-Ansätze ein Grundgerüst erarbeitet und dieses anschließend mit didaktischen Terminologien sowie den Erkenntnissen der Übersichtsmodelle angereichert. Ein zentraler Aspekt ist hierbei die Überführung abstrakter Konzepte in konkrete Design-Schritte, durch welche das Zusammenspiel physischer Gegebenheiten und technischer Komponenten systematisch, praxisgerecht sowie kontextorientiert eingeplant werden kann.

Das folgende Kapitel 4 beschäftigt sich dann mit einer möglichen Systemarchitektur hybrider Ökosysteme, also der dritten Forschungsfrage. Zuvor ermittelte Kernelemente und Dimensionen werden dabei aus einer technischen Perspektive betrachtet, um darauf aufbauend sowohl den notwendigen Anwendungstypen, als auch einen geeigneten Architekturstil zu ermitteln. Diese technischen Rahmenbedingungen dienen anschließend der Konzeption einer möglichen Systemarchitektur. Um eine strukturierte Umsetzung dieses Konzepts zu ermöglichen, werden hierbei die Zuständigkeitsbereiche sämtlicher Bestandteile detailliert beschrieben und begründet, was auch die Darstellung wichtiger Hintergrundinformationen und Designentscheidungen einbezieht.

Das letzte Kapitel 5 beinhaltet abschließend ein Fazit sowie einen Ausblick, in welchen die Ergebnisse vor dem Hintergrund des aktuellen Forschungsstandes kritisch betrachtet und logische Erweiterungen sowie Folgeschritte dieses Forschungsprozesses dargestellt werden.

Vor der Modellierung eines Systems müssen zunächst Arbeitsdefinitionen der hybriden Lehre und hybriden Lernräume vorhanden sein, da diese das zentrale Konzept festhalten. Trotz der momentanen Relevanz dieses Themengebietes existieren jedoch noch keine einheitlichen Begriffsverständnisse in aktueller Literatur. Dieses Kapitel beschäftigt sich nun mit den unterschiedlichen Ansichten hinsichtlich hybrider Lehre und Lernräume. Dabei werden Gemeinsamkeiten und Abgrenzungen zu verwandten Themenfeldern genannt, auf deren Basis anschließend Arbeitsdefinitionen entstehen.

## 1.5. Definitionen hybrider Lehre

Durch eine ausführliche Literaturrecherche ist ersichtlich, dass hybride Lehre mit einer Varietät unterschiedlicher Begriffe in Verbindung gebracht wird. So finden sich häufig die Begriffe Hybrid Learning, Blended Learning, Seamless Learning, Hybrid Courses, Hybrid Education und neuerdings HyFlex oder Multi Access Learning, deren Bedeutungen nicht vollständig trennscharf sind. Auch innerhalb der einzelnen Begriffsdefinitionen sind über die vergangenen Jahre mehr oder minder starke Änderungen festzustellen. Um den tatsächlichen Umfang hybrider Lehre einschätzen zu können, werden diese Begriffe nun erläutert und ihre Unterschiede, sowie Gemeinsamkeiten genannt.

### 1.5.1. Blended Learning

Der wohl am häufigsten als Synonym für hybride Lehre genannte Begriff ist *Blended Learning*. So sind Aussagen wie „Hybrid learning, sometimes called "Blended learning" [...]“ [Zhang, 2008, S. 253] oder „The terms blended learning, hybrid learning [...] are used interchangeably in current research“ [O'Byrne and Pytash, 2015, S. 137] aufzufinden. Doch ähnlich der hybriden Lehre finden sich auch für das Blended Learning unterschiedliche Definitionen, sodass Driscoll [2002] bereits unterschiedliche Auffassungen bemängelte: „The point is blended learning means different things to different people“ [Driscoll, 2002, S. 1]. Laut ihm finden sich jedoch vier hauptsächliche Ausprägungen dieses Lern-Paradigmas [vgl. Driscoll, 2002, S. 1]:

1. Die Kombination oder Mischung von webbasierten Technologien um ein Lehrziel zu erreichen.
2. Die Kombination unterschiedlicher pädagogischer Herangehensweisen um ein Lernziel optimal zu erreichen, sei es mit oder ohne Unterrichtstechnologien.
3. Die Kombination unterschiedlicher Unterrichtstechnologien mit lehrergeleitetem *Face-to-Face* Unterricht.

4. Die Integration echter Arbeitsaufträge in das Unterrichtsdesign, um das Lernen und Arbeiten zu vereinen.

Dabei ist ein starker Bezug zu der Online-Lehre ersichtlich, da er das Blended Learning lediglich als Zwischenschritt zwischen vollständiger *Face-to-Face* Lehre und vollständiger Online-Lehre beschreibt, welcher den Übergang zum E-Learning vereinfachen sollte.

Im Gegensatz zu Driscoll [2002] sieht Graham [2006] Blended Learning als vollwertiges Lehr-Paradigma und nicht nur als Zwischenschritt zum vollständigen E-Learning. Auch er nennt drei Ausprägungen des Blended Learnings, darunter die Kombination von Unterrichtsmodalitäten oder Sendemedien, die Kombination von Unterrichtsmethoden, sowie die Kombination von *Face-to-Face* und Online-Anweisungen [vgl. Graham, 2006, S. 4]. Da die ersten beiden Ausprägungen jedoch zu offen sind und jedes Lernsystem beschreiben können, fokussiert er sich auf die dritte Variante, auf deren Basis er folgende Arbeitsdefinition formuliert:

„Blended learning systems combine face-to-face instructions with computer-mediated instructions.“ [Graham, 2006, S. 5]

Mit dieser will er verstärkt die zentrale Rolle von computerbasierter Technologie als Ergänzung des traditionellen Unterrichts hervorheben. Für die Wahl des Ansatzes erkennt Graham [2006], angelehnt an vorherige Erkenntnisse, vor allem drei Vorteile als ausschlaggebend [vgl. Graham, 2006, S. 8 ff.]:

1. Die verbesserte pädagogische Vielfalt, da sowohl Online- als auch Präsenzlehre unterschiedliche Lernstrategien fördern und ihre Kombination wiederum neue Strategien ermöglicht.
2. Die erhöhte Erreichbarkeit der Lehre und Lernmaterialien, sowie die Flexibilität, Lehre jederzeit und ortsunabhängig stattfinden zu lassen.
3. Die erhöhte Kosteneffizienz durch reduzierte Kosten für physische Infrastruktur und eine effizientere Zeitplanung.

Grahams Erkenntnisse werden auch in aktueller Literatur weiterhin referenziert (siehe beispielsweise O'Byrne and Pytash [2015], Boelens et al. [2017] oder Lowenthal et al. [2020]) und sind auf der Plattform *Google Scholar* einige der meist zitierten Quellen des Themenkomplexes. So hat sich im Bereich des Blended Learnings ein starker Bezug zu technologischen Entwicklungen abgebildet und ähnliche technologiezentrierte Definitionen folgten. Zhang [2008] betitelt Blended Learning beispielsweise als Lehre, welche auf dem physischen Kontakt bzw. der physischen Präsenz von Lernenden aufbaut, jedoch viel des Lernmaterials und der Interaktionen ebenso online bereitstellt,



etwa durch *Content Delivery Software* oder Lernanwendungen [vgl. Zhang, 2008, S. 253]. So sollen laut ihm die besten Eigenschaften beider Welten in einem Paradigma vereint werden. Angelehnt an die ermittelten Vorteile von Graham [2006] nennt er dabei die Vorteile einer gesteigerten pädagogische Vielfalt, Lerneffektivität und Kosteneffizienz. Auch der einfachere Zugang zu Wissen und die Revisionsfreundlichkeit werden in diesem Zusammenhang erwähnt. Um diese Vorteile ausschöpfen zu können, ist laut Zhang [2008] jedoch eine durchdachte Kombination der Ansätze notwendig.

Bisherige Definitionen fokussieren vorwiegend die reine Kombination der physischen und Online-Präsenz, sodass genauere Eingrenzungen innerhalb solcher Kombinationen noch nicht bedacht werden. Bower et al. [2015] betrachten weitere Unterteilungen anhand der Synchronität beinhalteter Modalitäten. Ihren Ansatz, das *Blended Synchronous Learning*, beschreiben sie dabei wie folgt:

„Learning and teaching where remote students participate in face-to-face classes by means of rich-media synchronous technologies such as video conferencing, web conferencing, or virtual worlds.“ [Bower et al., 2015, S. 1]

Somit bringen sie alle zuvor definierten Blended Learning Ansätze vorwiegend mit einem asynchronen bzw. abwechselnden Gebrauch der Modalitäten in Verbindung. Die parallele Nutzung sehen sie als Neuerung und Alleinstellungsmerkmal, welches erst mithilfe von synchroner Technologie umsetzbar ist. Dieser Mehrwert neuer Technologie wird auch von der Plattform *e-teaching.org* hervorgehoben, welche ebenfalls Blended Learning und hybrides Lernen als Synonym darstellt:

„Blended Learning (Auch: hybrides Lernen); ein Lehr-/Lernkonzept, das eine didaktisch sinnvolle Verknüpfung von Präsenzveranstaltungen und virtuellem Lernen auf der Basis neuer Informations- und Kommunikationsmedien vorsieht.“ [vgl. e-teaching.org, [n. d.]]

### 1.5.2. Gemischte Terminologien

Aufbauend auf dem Blended Learning finden sich weitere Definitionen, deren Benennungen einen stärkeren Bezug zur Hybridität aufweisen. Garnham and Kaleta [2002] beschreiben beispielsweise das Konzept der *Hybrid Courses*. Hybrid Courses sind demnach Kurse, in welchen ein erheblicher Anteil der Lehre online stattfindet, um die benötigte Zeit in einem Klassen- / Vorlesungsraum zu reduzieren, jedoch nicht vollständig zu beseitigen [vgl. Garnham and Kaleta, 2002, S. 1]. Mithilfe von computerbasierter Technologie werden dabei einige der bestehenden Vorlesungs- und Übungsinhalte in neue Online-Aktivitäten umgestaltet, beispielsweise Fallstudien, Tutorials, Selbsttests,

Simulationen oder digitale Gruppenarbeiten. Mit dem Wort *Hybrid* wird hier also eine Kombination von Online- und Offline-Unterricht angesprochen. Als Ziele nennen sie die Zusammenführung der besten Eigenschaften beteiligter Unterrichtsformen, sowie die Unterstützung des aktiven selbstständigen Lernens und die Reduktion der benötigten physischen Präsenzzeit [vgl. Garnham and Kaleta, 2002, S. 1]. Diese Beschreibung erinnert stark an die zuvor genannten Definitionen des Blended Learnings, sodass eine Synonymität vermutet werden kann. Doch beschrieben Hinterberger et al. [2004] Hybrid Courses vor allem als Zwischenschritt zum Blended Learning. So seien Hybrid Courses lediglich traditionelle Kurse, welche mit E-Learning Aspekten im Nachhinein angereichert werden, während Blended Learning solche Kurse darstellt, bei deren Konzeption das E-Learning bereits strategisch bedacht und integriert wird [vgl. Hinterberger et al., 2004, S. 3 ff.].

Ein weiteres, sich von dem Blended Learning abhebende Konzept ist in dem Ansatz von Beatty [2007] dokumentiert, welchen er aufgrund der mangelnden Flexibilität des Paradigmas entwickelte. Dabei werden ebenfalls die Begriffe *Hybrid* und *Blended* als Synonym verwendet, weshalb er seinen Ansatz *HyFlex-Learning* nennt. Das *Hy* steht für die Kombination von *Face-to-Face* und Online-Lehre bzw. Lernaktivitäten, während *Flex* die gewonnene Flexibilität zum Ausdruck bringen soll [vgl. Beatty, 2007, S. 15]. Lernende erhalten in diesem Ansatz selbst die Entscheidungsmöglichkeit ob sie dem physischen Unterricht beiwohnen wollen oder nicht. Alternativ müssen also synchrone Online-Partizipationen und / oder asynchrone Zugriffe auf Lernmaterialien ermöglicht werden, ohne dass diese zu Lerndefiziten führen.

Auch Irvine et al. [2013] befassten sich mit der Flexibilität von Lehre auf Basis des Blended / Hybrid Learnings. In ihrem *Multi-Access*-Ansatz untersuchen sie die unterschiedlichen Möglichkeiten der Partizipation, für welche die vier Varianten *Face-to-Face*, *Synchronous Online*, *Asynchronous Online* und *Open Learning* genannt werden [vgl. Irvine et al., 2013, S. 175 ff.]. *Face-to-Face* Unterricht beschreiben sie dabei stets als synchron und physisch. *Synchronous Online* repräsentiert das *Blended Synchronous Learning*, während *Asynchronous Online* beispielsweise die spätere Vergabe von Aufnahmen synchroner Veranstaltungen beinhaltet. Letztlich benennen sie das *Open Learning*, bei welchem Lerninhalte und Materialien dauerhaft und unabhängig von einer Vorlesung bereitgestellt werden. Dazu gehören beispielsweise Veranstaltungen, welche vollständig auf Skripten und Screencasts basieren. Das Besondere des *Multi-Access* Ansatzes ist nun, dass den Lernenden die Kombination der Modalitäten freigestellt wird, sodass diese selbst entscheiden können, wie sie einen Kurs absolvieren. Es gibt jedoch Kurse, in denen eine reine asynchrone Partizipation nicht möglich ist

bzw. synchrone Teilnahme vorausgesetzt wird. Lehrende erhalten dann die Möglichkeit, Modalitäten auf Gegebenheiten des Kurses abzustimmen [vgl. Irvine et al., 2013, S. 175].

Die Vielfalt der genannten Ansätze fasst Irvine [2020, S.3 f.] nach ausführlichen Recherchen in einem Vergleich zusammen, in welchem verschiedene Charakteristiken zur Gegenüberstellung genutzt werden:

- **Face-to-Face:** Als *Face-to-Face* betitelte sie alle physischen Präsenz-Lehren /-- Lehrformen.
- **Synchronität und Gleichzeitigkeit:** Im Gegensatz zu vorherigen Definition, welche die Synchronität als Gleichzeitigkeit der Nutzung unterschiedlicher Modalitäten benannten, führt Irvine eine weitere Unterteilung ein, in welcher sie Synchronität und Gleichzeitigkeit unterscheidet. Mit den Ausprägungen *synchronous* und *asynchronous* beschreibt sie rein zeitlich bedingte Kontakte zwischen Lehrenden und Lernenden (z.B. sind diese zeitgleich aktiv oder zu unterschiedlichen Zeitpunkten). Mit den Ausprägungen *concurrent* und *consecutive* hingegen soll die Gleichzeitigkeit der Modalitäten ausgedrückt werden (z.B. Präsenz- und Online-Lehre parallel oder sequentiell).
- **Open Access:** Mit *Open Access* beschreibt sie die Möglichkeit des *Open Learnings*, welche sie in dem *Multi-Access* Konzept schildert.

Ihr Vergleich wird in Tabelle 1.2 dargestellt. Mit dem \* signalisiert Irvine die Möglichkeit einer optionalen Wahl für Lernende, während die  $\hat{\phantom{x}}$  eine optionale Integration der Modalität in die Kurs-Lehre darstellt.

Tabelle 1.2.: Vergleich hybrider Lehransätze. Inhalt entnommen aus Irvine [2020, S. 3]

	Face-to-Face	synchronous concurrent	synchronous consecutive	asynchronous consecutive	open access
Blended (Hybrid)	X		$\hat{X}$	$\hat{X}$	
HyFlex	X*	X*		X*	
Multi-Access	$\hat{X}$	$\hat{X}$	$\hat{X}$	$\hat{X}$	$\hat{X}$
Blended Synchronous (Synchronous Hybrid)	X*	X*			

Eine wichtige Erkenntnis ist, dass auch sie hybride Lehre mit Blended Learning gleichsetzt, innerhalb des Themenfeldes jedoch eindeutige Unterschiede sieht. Der *Multi-Access* Ansatz ist ihr zufolge der dynamischste hybride Ansatz, gefolgt von Beattys *HyFlex* Konzept. Auch wenn diese beiden verwandt zu sein scheinen (*HyFlex* als eine Ausprägung von *Multi-Access*), so nennt Irvine [vgl. 2020, S. 5 f.] einen entscheidenden Unterschied. In der *HyFlex*-Lehre müssen den Lernenden alle Auswahlmöglichkeiten als Optionen angeboten werden, während im *Multi-Access* angebotene Modalitäten an das Lehrszenario anzupassen sind, also optional bei der Erstellung bedacht werden können. Auch innerhalb des Blended Learnings hebt sie die Unterschiede zwischen ursprünglichen und synchronen Hybriden hervor. Während das ursprüngliche Blended Learning synchrone oder asynchrone Online-Angebote sequentiell mit einem *Face-to-Face* Unterricht kombiniert, muss beim *Blended Synchronous*-Ansatz sowohl physische, als auch Online-Präsenz synchron angeboten werden und parallel aktiv sein.

### 1.5.3. Hybrid Pedagogy

Hybridität ist naturgemäß vielschichtig. So finden sich auch unterschiedliche Standpunkte, aus welchen sie betrachtet werden kann. Stommel skizziert diese in einigen Artikeln über die *Hybrid Pedagogy* aus einer didaktischen und pädagogischen Sichtweise. Hybridität im engsten Sinne beschreibt auch Stommel [2012] als eine Kombination von Lehre in Klassen- / Vorlesungsräumen und der Online-Lehre, wodurch er ebenfalls die Ähnlichkeit zum Blended Learning ersichtlich macht. Jedoch bestehen seiner Meinung nach einige Unterschiede zwischen diesen beiden Begriffen. Blended Learning beschreibt er als einen Prozess bzw. eine Praktik für eine bestimmte Konstellation. Es ist etwas Taktisches und fest Definiertes. Mit Blended Learning wird dabei vor allem der Raum angesprochen, in welchem die Lehre geschieht, sei es im physischen Klassenraum oder einer Online-Umgebung. Hybrid Pedagogy hingegen beschreibt er als Methodologie bzw. einen Kriterienkatalog, welcher eine Erstellung angepasster Prozesse und Praktiken ermöglicht. So ist sie für ihn etwas Strategisches und Dynamisches. Dabei ist eine tiefergehende Resonanz ersichtlich, welche auf einem fundamentalen Umdenken des Konzepts von Räumen basiert. Es wird eine bestmögliche Vereinigung der unterschiedlichen Konzepte anhand ihrer Eigenschaften und Möglichkeiten angestrebt:

„So, hybrid pedagogy does not just describe an easy mixing of on-ground and online learning, but is about bringing the sorts of learning that happen in a physical place and the sorts of learning that happen in a virtual place into a more engaged and dynamic conversation.“ [vgl. Stommel, 2012]

Auch wenn in aktueller Literatur die Hybrid Pedagogy von Stommel meist mit der hybriden Lehre gleichgesetzt wird, beispielsweise von Hilli et al. [2019], so grenzt er

selbst diese beiden Begriffe voneinander ab. Das *Hybrid* soll seiner Aussage nach mehr beinhalten als nur hybride Lehre. Es verweist auf sämtliche Hybriditäten in der modernen Pädagogik und soll zum Denken über das reelle und virtuelle Leben anregen. So sieht er in der hybriden Pädagogik mehr Dimensionen als nur den physischen und virtuellen Raum. Es geht vielmehr darum, bestehende Dichotomien zu erkennen und sich ihrer Rolle im Lernkontext bewusst zu werden. So hinterfragt er beispielsweise die Rollen von Lehrenden und Lernenden, den Mix von Disziplinarität und Interdisziplinarität oder die Möglichkeiten der akademischen und außerakademischen Umgebungen [vgl. Stommel, 2012]. Diese Mehrdimensionalität ist für ihn in einem Hybriden zentral, jedoch ist sein Ziel keine unbemerkte Grenzüberschreitung zweier zunächst unabhängiger Disziplinen [vgl. Stommel and Rorabaugh, 2012]. Die Grenzen sollten erkundet werden und bewusst sein. Anhand der Gemeinsamkeiten und sich ergänzender Elemente sollte eine Vereinigung entlang der Grenzen erfolgen, bei der dennoch die einzelnen Bestandteile klar ersichtlich sind. Eine Abweichung von altbekannten Ansichten und Strukturen ist bei der Schaffung solch eines neuen Hybriden unumgänglich. Dies beherbergt immer Risiken, jedoch auch Chancen der Verbesserung [vgl. Stommel, 2012]. Als Grundlage solch eines Paradigmas sieht Stommel vor allem ein soziales Lernen, aufbauend auf Diskussionen und Kommunikationen bzw. Abstimmungen. Zudem ist für ihn der Entstehungsprozess wichtiger als der resultierende Hybrid selbst:

„Hybridity is about the moment of play, in which the two sides of the binary begin to dance around (and through) one another before landing in some new configuration. [Hybrid Education] is not just about what will become of us in the wake of technological and cultural transformation, but also (and perhaps more predominantly) about the process of becoming itself.“  
[Stommel, 2012]

#### **1.5.4. Internationaler Vergleich des Hybrid und Blended Learnings**

Auch Reinmann [2021] beschäftigt sich mit der Frage nach einer einheitlichen Definition hybrider Lehre. Hierzu vollzieht sie sowohl national als auch international ausführliche Recherchen bezüglich des Begriffsverständnisses. Schon auf den Webseiten verschiedener Deutscher Hochschulen kann sie variierende Auffassungen wahrnehmen [vgl. Reinmann, 2021, S. 1 f.]. Während einige Institute einen Hybriden bereits durch die pure Integration von on- und offline Elementen definieren, beschreiben andere eine stärkere Trennung der Begriffe *Hybrid* und *Blended*. So wird das Blended Learning vorwiegend mit einem abwechselnden Gebrauch der Modalitäten in Verbindung gebracht, während hybrider Lehre eine parallele Nutzung der on- und offline Elemente zugeschrieben wird. Auffallend ist zudem, dass die Parallelität im Gegensatz zu der

Wortwahl von Irvine [2020] hier vor allem mit den Begriffen *synchron* und *asynchron* umschrieben wird. Bei der internationalen Recherche englischsprachiger Texte bemerkt sie ebenfalls eine häufige Kategorisierung aufgrund der Synchronität, jedoch werden die Begriffe *Hybrid* und *Blended* vorwiegend als Synonym verwendet [vgl. Irvine, 2020, S. 2f.]. So nimmt auch sie eine Vielzahl verschiedener Definitionen wahr, welche bereits national variieren, jedoch vor allem international unterschiedlich sind. Einen Grund hierfür sieht sie in dem digitalen Wandel mit dessen neuen Möglichkeiten, da man nun „[...] für die Beschreibung neuer Optionen beim Lehren und Lernen zwangsläufig bestehende Begriffe erweitern, modifizieren oder neue finden muss“ [Irvine, 2020, S. 3]. Auch wenn der Bedarf nach einem einheitlichen Vokabular besteht, sieht sie durch den stetigen Wandel eine klare Unterscheidung zwischen Blended Learning und Hybrid Learning als noch wenig erfolgsversprechend. Auf dieser Annahme aufbauend lautet ihr Kategorisierungsvorschlag wie folgt [vgl. Irvine, 2020, S. 4]:

- **Präsenz-Lehren:** Lehrende und Lernende befinden sich an demselben physischen Ort und interagieren zur gleichen Zeit.
- **Online-Lehren:** Lehrende und Lernende befinden sich an verschiedenen Orten und interagieren zur gleichen Zeit oder zeitversetzt über digitale Umgebungen.
- **Synchrones Hybrid-Lehren:** Lernende befinden sich an demselben physischen Ort wie Lehrende oder sind von anderen Orten online zugeschaltet, wobei eine doppelte Synchronität der physischen und digitalen Präsenz vorherrscht.
- **Asynchrones Hybrid-Lehren:** Lehrende und Lernende befinden sich zu verschiedenen Zeiten am gleichen Ort oder interagieren online zeitgleich oder zeitversetzt an verschiedenen Orten, sodass eine Asynchronität der physischen und digitalen Präsenz vorherrscht.

### 1.5.5. Seamless Learning

Ein letzter Begriff, mit dem hybride Lehre häufig in Verbindung gebracht wird, ist das *Seamless Learning*. Kuh [1996] sieht im Seamless Learning die zentrale Idee, ein Lernen innerhalb und außerhalb des Klassenraums zu unterstützen, diese Aspekte also zu verbinden und nicht als Dichotomie zu behandeln. Das Gelernte soll dabei durch Lebenserfahrungen gefestigt und weiter ergründet werden.

„In seamless learning environments, students are encouraged to take advantage of learning resources that exist both inside and outside of the classroom [...] students are asked to use their life experiences to make meaning of material introduced in classes [...]“ [Kuh, 1996, S. 136]

Chan et al. [2006] beschreiben Seamless Learning zudem als ein Modell, in welchem Lernende zu jeder Zeit und an jedem Ort auf unterschiedliche Weisen lernen können. Dabei wird ihnen die Möglichkeit gegeben, schnell und einfach von einem Szenario oder Kontext in einen anderen zu wechseln, wobei persönliche Geräte wie Smartphones als Mediator eine zentrale Rolle einnehmen. WONG and SHARPLES [2013, S. 98] ergänzen weiterhin, dass der Lernfortschritt bei einem Szenarien- oder Kontextwechsel bestehen bleibt, also in das neue Setting übernommen wird. Als mögliche Szenarien nennen sie individuelles Lernen, das Lernen mit einer weiteren Person, sowie das Lernen in kleinen Gruppen oder großen Gemeinschaften. Zum Kontext gehören zudem der Einbezug von Lehrenden, Verwandten, Experten und anderen unterstützenden Instanzen, sowie die Modalität der Interaktion (z.B. Face-to-Face oder online). Letztlich ist der Ort des Lernens zu berücksichtigen, sei es ein Klassenraum, ein Außenbereich, ein öffentlicher Ort wie Museen oder ein digitaler Raum wie virtuelle Welten oder soziale Medien. Seamless Learning hat laut ihnen das Ziel, autonomes Lernen zu unterstützen und dabei auf den bestehenden Kenntnissen und Fähigkeiten des 21ten Jahrhunderts aufzubauen [vgl. WONG and SHARPLES, 2013, S. 98]. Dabei sollen nicht nur einzelne Aktivitäten, sondern ganze Programme angepasst werden, um eine Transformation der Lernenden zu selbstgesteuerten Individuen hervorzurufen. Wong and Looi [2011] heben dabei besonders hervor, dass das Lernen zu jeder Zeit und an jedem Ort ermöglicht wird, jedoch keinesfalls einen stetigen Zwang darstellen sollte:

„The notion of seamless learning advocates “learning anytime, anywhere” and not “learning everytime, everywhere.”“[Wong and Looi, 2011, S. 2]

Zwei der wichtigsten Charakteristiken des Seamless Learnings sind laut WONG and SHARPLES [2013, S. 99] die *Seamless Adaptivity* und *Seamless Connectivity*. Ersteres steht dafür, dass sich die Technologie auf die Lernenden einstellt, ohne dass sie sich diesem bewusst sind. So könnten beispielsweise Lerninhalte oder Lernaktivitäten automatisch auf den Lernfortschritt, aktuellen Ort oder allgemeine Eigenschaften der Lernenden angepasst werden (z.B. Sprache, Schwierigkeitsgrad, Hilfestellungen etc.). Mit *Seamless Connectivity* hingegen soll die Kontinuität des Lernprozesses gesichert werden. Lernfortschritte sollten unabhängig von Geräten und Kontexten fortgeführt bzw. bestehende Lernergebnisse eingesehen werden können. Damit verbunden wird auch das Prinzip *BYOD - Bring Your Own Device* häufig angesprochen, also die Integration der alltäglichen Geräte von Lehrenden und Lernenden.

Looi et al. [2010, S. 156] nennen zudem den Aufbruch bestehender Dichotomien durch den Einsatz mobiler Technologien als wichtigen Aspekt. Brüche oder auch *Seams* zwischen diesen müssen aufgehoben werden, um eine nahtlose Lernerfahrung zu ermög-

lichen. Zu diesen Dichotomien gehören vor allem das formale und informelle Lernen. Formales Lernen beschreiben sie als fixierte Lehrpläne / Curricula in einem Klassenzimmer, während informelles Lernen bewusste oder unbewusste Lernerfahrungen außerhalb der Schulumgebung darstellt. So soll Seamless Learning privates und öffentliches Lernen über unterschiedliche Kontexte hinweg verbinden, wobei ebenfalls die Dichotomien des individuellen und kollektiven Lernens, sowie des Lernens im Schulkontext und außerhalb des Schulkontexts betroffen sind.

### **1.5.6. Arbeitsdefinition**

Die Recherche zeigt, dass sowohl der Begriff der hybriden Lehre, als auch damit verbundene und beinhaltete Konzepte unterschiedlich verstanden werden. Der Bezug zum Blended Learning ist stark vertreten, sodass eine genaue Unterscheidung immer noch nicht möglich ist bzw. diese in der Literatur noch nicht anhand einheitlicher Kriterien erfolgt. Reinmann [2021] bringt diese unscharfe Trennung mit dem technologischen Fortschritt in Verbindung, da hybride Lehre stark von technischen Möglichkeiten und Grenzen geprägt wird. Ebenso gibt sie zu bedenken, dass eine Trennung der Begriffe noch zu früh und wenig vielversprechend ist. Es wird zwar häufig die Synchronität als unterscheidender Faktor genannt, jedoch finden sich auch variierende Konzepte innerhalb des Blended Learnings, welche auf unterschiedlicher Synchronität aufbauen.

Nun wird in dieser Arbeit ein System fokussiert, welches die Flexibilität der Konfiguration und Nutzung hybrider Räume ermöglicht. Hier sollten demnach auch alle Synchronitäten abgebildet werden, sodass ähnlich dem Ansatz von Reinmann [2021] für grundlegende Definitionen auf eine Trennung der Begriffe verzichtet wird. Zudem ist trotz der variierenden Begriffsentwicklungen ein einheitlicher Ursprung erkenntlich. So beziehen sich das Blended Learning und die hybride Lehre beide auf die Kombination unterschiedlicher pädagogischer Ansätze, Modalitäten und Technologien. Demnach sind viele der genannten Aspekte aus den unterschiedlichen Konzepten für eine flexible und kontextgerechte Nutzung zu berücksichtigen. Neben den unterschiedlichen zeitlichen Aspekten von Irvine [2020] sollten beispielsweise auch die Dichotomien und das Raumverständnis von Stommel [2012] berücksichtigt werden. Auch wenn Seamless Learning einen ähnlich starken Bezug zum Aufbruch bestehender Dichotomien aufweist, so erscheint es dennoch als besser separierbar. Im Gegensatz zum Blended Learning und Hybrid Learning, fokussiert sich das Seamless Learning nicht auf die allgemeine Kombination verschiedener Präsenzformen bzw. dadurch entstehende Vorteile, sondern vielmehr auf die Technologie als Mediator. So betrachtet es notwendige



Eigenschaften dieser Technologie, welche einen fließenden Übergang zwischen den Dichotomien ermöglichen oder unterstützen. Der Ausgangspunkt ist also die Technik und wie sie im Alltag eingesetzt werden kann oder bereits eingesetzt wird. Mobile Technologie ist dabei besonders hervorzuheben. Seamless Learning erscheint somit eher als wichtige Charakteristik / Voraussetzung eines hybriden Systems. Es sollte nahtlos in bestehende Praktiken integrierbar sein, um so die Vorteile unterschiedlicher Praktiken besser ausnutzen zu können, sowie die Hürden der Nutzung zu reduzieren.

Auf Basis dieser Erkenntnisse wird im weiteren Verlauf der Ausarbeitung folgende Arbeitsdefinition für hybride Lehre angenommen:

#### Definition 1.1: Hybride Lehre

*Hybride Lehre beschreibt ein Lehr- / Lernparadigma, welches grundlegend auf der Kombination unterschiedlicher Lehr- / Lernparadigmen aufbaut und dabei sowohl physische als auch digitale Komponenten einsetzt und kombiniert. Abhängig vom Lehrkontext können dabei Lehrende und Lernende gemeinsam (synchron) oder zeitversetzt (asynchron) Lerninhalte in physischer oder digitaler Präsenz erarbeiten. Der Einsatz von Modalitäten innerhalb einer Veranstaltung kann parallel oder sequentiell erfolgen. Zusätzlich können Lernmaterialien für ein Selbststudium auch unabhängig von Präsenzveranstaltungen ausgegeben werden (Open Access). Digitale Anwendungen, ihre Möglichkeiten und der allgemeine technische Fortschritt haben demnach eine große Auswirkung auf dieses Paradigma. Ein zentraler Aspekt der hybriden Lehre ist zudem die Offenheit gegenüber Neuerungen. Neben der Kombination von Off- und Online-Lehre gilt es auch weitere Dichotomien und Disziplinen der Lehre zu ergründen, durch ihre Überbrückung neue Unterrichtsformen zu schaffen, sowie ein aktives, flexibles, authentisches und nutzerzentriertes Lernen kontextgerecht mit dem alltäglichen Leben zu vereinigen.*

--- Dennis Dubbert

## 1.6. Definitionen hybrider Lernräume

Aufbauend auf der Definition für hybride Lehre kann nun definiert werden, was hybride Lernräume sind und wie sich diese im Rahmen hybrider Veranstaltungen eingliedern. Auch hierzu finden sich, angelehnt an das Blended Learning und die hybride Leh-

re, unterschiedliche Begriffsverständnisse und Benennungen, welche in den folgenden Abschnitten genauer betrachtet werden.

### 1.6.1. Blended Spaces

Im Zusammenhang mit hybriden Lernräumen wird häufig das Prinzip der *Blended Spaces* referenziert, ein Begriff, der stark durch David Benyons Veröffentlichungen geprägt wurde. Benyon and Mival [vgl. 2012, S. 2] beschreiben Blended Spaces als eine *Mixed-Reality-Umgebung* jeder Größenordnung, in welcher reelle und virtuelle Welten bezüglich einiger Inhalte sorgfältig und bedacht zusammengeführt werden. Dabei kann laut ihnen der physische Raum durch sämtliche, sich darin befindenden Objekten beschrieben werden, durch topographische Beziehungen zwischen Objekten und Menschen innerhalb des Raums, sowie durch die Flüchtigkeit des Raums. Der digitale Raum hingegen ist beschreibbar durch dessen Ontologie und Topologie bzw. die Informationsarchitektur. Ebenso beschreibend sind Personen mit all ihren Interaktionen im Raum (Agency) und wiederum dessen Veränderlichkeit. Die Zusammenführung der physischen und digitalen Räume schafft stets Möglichkeiten für neue Erfahrungen, sei es zu Unterhaltungszwecken, zur primären Informationsübermittlung oder zur Funktionsanreicherung der jeweiligen physischen bzw. digitalen Komponente.

Dabei verweisen Benyon and Mival [2012] auf die vier mentalen Domänen von Fauconnier and Turner [2008] (siehe Abbildung 1.2). Sie nennen zwei *Input Spaces*, den physischen und digitalen Raum, welche Gemeinsamkeiten mit einem generischen Informationsraum aufweisen. Als geteilte Hauptkomponenten des generischen Raums sehen sie beispielsweise die Objekte des Raums (Ontologie), die Beziehungen zwischen den Objekten (Topologie), die Menge der Änderungen innerhalb des Raums (Volatility = Flüchtigkeit) und beteiligte Personen mitsamt ihrer Wertvorstellungen, Prioritäten, Fähigkeiten und Ziele, sowie ihre Aktionen und Interaktionen (Agency = Organ). *Blending* beschreiben sie nun als Operation, welche auf beide Input Spaces angewandt wird und so einen neuen Blended Space aufspannt. Der Hauptgedanke ist, dass beim *Blending* die Projektionen der unterschiedlichen Input Spaces völlig neue Beziehungen ermöglichen und bekannte kognitive und kulturelle Strukturen so auf eine neue Weise wahrgenommen werden. Die Kombination ermöglicht somit ebenso neue Ideen und Erkenntnisse. Der entstehende *Blend* kann dann wiederum durch die Kombination mit weiteren mentalen Räumen stetig neue Möglichkeiten bieten.

In einer folgenden Veröffentlichung ergänzt Benyon [2014, S. 79], dass Blended Spaces über eine einfache Mischung der Räume hinausgehen. Ziel des Designs ist es nach

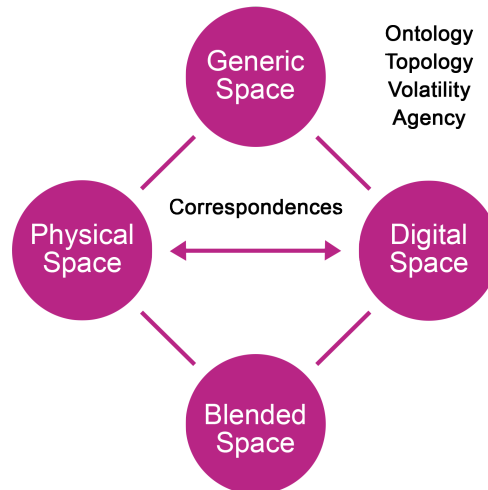


Abbildung 1.2.: Benyons mentale Räume, inhaltlich entnommen aus Benyon and Mival [2012]

ihm, dass sich Menschen in einem Blended Space präsent fühlen und direkt auf den Inhalt des Blended Space einwirken können. Innerhalb einer weiteren Veröffentlichung beschreiben Benyon and Mival [2015, S. 225] die Kombination der Räume detaillierter. Statt einer einfachen Kombination ohne Regeln, wird beim Blending ein digitaler Raum entworfen und in einen physischen Raum integriert, um eine neuartige Oberfläche (UX) zu schaffen. Erst durch diese Integration des Digitalen ins Physische kann dann ein neues Gefühl der Präsenz hervorgerufen werden.

### 1.6.2. Hybrid Learning Spaces (HLS)

Im direkten Bezug zu hybriden Lernräumen werden Stommel and Rorabaugh [2012] häufig genannt, welche neben einer grundlegenden Definition von Hybridität den Begriff der hybriden Pädagogik und damit verbundener Räume diskutieren. Wie bereits in Abschnitt 1.5.3 angesprochen, bestehen Lehrveranstaltungen auf Basis der hybriden Pädagogik aus vielen unterschiedlichen Dimensionen und Disziplinen. Auf Basis dieser Vielfalt regen Stommel and Rorabaugh [2012] bei der Planung eines hybriden Lernraums ein Umdenken des Konzepts von Räumen an, welche diese Interdisziplinarität und Mehrdimensionalität widerspiegeln sollen. Bei der Planung hybrider Lehre ist das Bewusst-Machen der Unterschiede und Grenzen beteiligter Räume unumgänglich. Aus den verschiedenen Disziplinen werden kontextabhängig die besten, sich gegenseitig ergänzenden Elemente extrahiert, jedoch kein fließender Übergang angestrebt. Ähnlich der Beschreibung von Benyon [2014] (siehe Abschnitt 1.6.1) ermöglicht jeder Raum eine

andere Form der Präsenz, welche auch als solche dem jeweiligen Ursprung zuordenbar sein sollte. Auch wenn diese Räume innerhalb einer Konstellation angeboten werden und sich ergänzen, können ihre Anwendungsgebiete sowie ihre kombinierte Nutzung innerhalb des Kontextes variieren, um verschiedene Ziele zu verfolgen. Es werden also nicht einfach zwei bestehende Elemente verschmolzen, sondern aus ihren Bestandteilen etwas stetig Neues geschaffen, welches zeitgleich immer noch die Ausgangselemente erkennen lässt.

Hilli et al. [2019] greifen diesen Gedanken auf. Sie betonen dabei, dass hybride Lernräume über die Zusammenführung physischer und digitaler Elemente hinaus gehen:

„Following this, a Hybrid Learning Space is a context of learning that not only moves beyond distinctions between online and offline spaces, but also often challenges divisions between teacher/student roles, formal/informal contexts, analogue/digital communication/media and other traditionally separable dimensions. Hybrid Learning Spaces and hybrid pedagogy offer new ‘complex hybrid breeds’ and as such potentially new possibilities for collaboration in higher education.“ [Hilli et al., 2019, S. 67]

Sie sehen hybride Lernräume als eine spezielle und kontextspezifische Konfiguration ökologischer Hochschullehre, welche sie auf Basis von Köppe et al. [2017] durch Offenheit, Risikobereitschaft, Experimentierfreude, Empathie, Dialog und kritische Kreativität charakterisieren. So vergleichen sie hybride Lernräume auch mit *Medienökologien*. Medienökologien beschreiben sie dabei als gleichzeitig individuell und geteilt. Sie bestehen aus einer Mehrzahl von Medien auf unterschiedlichen Ebenen wie beispielsweise Terminals, Plattformen und Diensten. Teilnehmende in der Medienökologie beschäftigen sich mit verschiedenen Teilen der Ökologie um verschiedene sinnvolle Ziele zu verfolgen, wie Beziehungsaufbau, Aufgabenlösung, Lernen oder Spaß haben. Die durch Medien aufrechterhaltenen oder hergestellten Beziehungen aber werden oft über eine Vielzahl von Kanälen und nicht nur über einen geführt [vgl. Hilli et al., 2019, S. 72]. So beschreiben sie Medienökologien selbst als hybride Phänomene, welche der Überbrückung unterschiedlicher Dichotomien dienen. Räume sind dadurch stets mit Bedeutung gefüllt, da sie Relationen aufweisen und Objekte beherbergen, welche für Nutzer in dem jeweiligen Kontext wichtig und bedeutungsvoll sind [vgl. Hilli et al., 2019, S. 70].

Ähnlich beschreibt auch Kohls [2019] die Aufgabe hybrider Lernräume:

„The hybrid learning spaces are linked to certain strategic and operational objectives, which relate both to the university context and to social development.“ [Kohls, 2019, S. 230]

Dabei sollen sie auf der operativen Ebene projektbasiertes und fähigkeitsgesteuertes Lernen unterstützen, sowie eine zielorientierte *Maker-Kultur* fördern. Um teamorientiertes, selbstgesteuertes und informelles Lernen zu fördern, erkennt er die Raumgestaltung in akademischen Lehrkontexten als einen der wichtigsten Punkte an [vgl. Kohls, 2019, S. 230]. Im Gegensatz zu Benyon (siehe Abschnitt 1.6.1) ist laut Kohls [2017] die Anreicherung innerhalb eines Hybriden keine einseitige Aktivität. So werden nicht nur physische Räume mit digitalen Produkten angereichert, sondern auch digitale Räume mit physischen Artefakten:

„The dichotomy of digital and non-digital artefacts is resolved in a hybrid learning space by seamlessly bridging different types of artifacts, making digital data touchable and graspable, enhancing physical objects with digital information and digitizing physical objects.“[Kohls, 2017, S. 3]

Ähnlich sehen es auch Ellis and Goodyear [2016, S. 161], welche bestehende Studien zum Thema *Learning Spaces* in zwei Kategorien einteilen:

1. Studien, welche sich auf den physischen Ort fokussieren und dessen Bestandteile in digitale Orte überführen.
2. Studien, welche sich auf digitale Artefakte und virtuelle Orte fokussieren und für diese eine Überführung ins Physische untersuchen.

Cohen et al. [2020] bemerken zusätzlich, dass hybride Lernräume eine neue Qualität bereitstellen. Darin getätigte Aktivitäten generieren Daten, welche zur Verfolgung der Prozesse individuellen und sozialen Lernens dienen können. Diese Daten können zu personen- oder kontextspezifischen Anpassungen des Raums und der Lehre genutzt werden, wodurch ein besseres Bewusstsein für Lehre und Lernen entstehen kann. Ebenso wie Stommel [2012] (siehe 1.5.3) sehen sie also den Prozess des Lernens selbst und dessen Kontrolle als wichtige Bestandteile hybrider Lehre und Lernräume an. Lernenden wird ein persönlicheres Lernangebot ermöglicht, sodass diese im Zentrum der Gestaltung stehen. Eine ähnliche zentrale Stellung von Lernenden beschreibt Trentin [2015]. Lernende sollten als *Prosumer* (proaktive Konsumenten) beim Gestaltungsprozess der Räume eingebunden werden, sie auf ihre Bedürfnisse anpassen oder sogar eigene Räume erstellen können. Dabei beschreibt Trentin [2015] das Prinzip *BYOD - Bring Your Own Device* als wichtige Voraussetzung für eine flexible und nutzerzentrierte Gestaltung. Gegenüber traditioneller Lehre sieht er zudem veränderte Bedeutungen

von Objekten, Personen und Rollen innerhalb eines hybriden Lernraums. Ihre Gegenüberstellung ist in Tabelle 1.3 aufgelistet.

Tabelle 1.3.: Vergleich der Bedeutungen von Objekten, Personen und Rollen in traditioneller und HLS-Lehre nach Trentin [vgl. 2015, S. 7]

	Traditionelle Lehre	HLS-Lehre
<b>Lehrende</b>	Besitzen und übertragen Wissen	Leiten beim Lernen an
<b>Lernende</b>	Empfangen Inhalte passiv	Interagieren mit Inhalten und Tutoren / Lehrenden - Lernen selbstständig und in Gruppen
<b>Unterricht / Klasse</b>	Ort, an dem Wissen induziert wird	Ort, an dem Wissen aufgebaut und geteilt wird
<b>Wissen &amp; Erfahrung</b>	Vertikal übertragen von Lehrenden zu Lernenden	Horizontal geteilt zwischen Mitgliedern der Lerngruppe, zu der auch Lehrende gehören
<b>Lernprozess</b>	Stark geleitet	Aktiv und kollaborativ
<b>Lerninhalte</b>	Vordefiniert und standardisiert	Flexibel und offen strukturiert, um mehrere Lernwege zu ermöglichen
<b>Wahrnehmungen bezüglich der Bildungstechnologie (ET)</b>	ETs als Surrogat für die Rolle von Lehrenden	ETs als Mittel zur Stimulierung von Lernenden, zur Verbesserung und Verstärkung der Lernumgebung

Letztlich sind die von Wals et al. [2012] angesprochenen *Hybrid Learning Configurations (HLCs)* zu nennen. HLCs beschreiben sie als kontextspezifische und soziale Praktiken, welche aufbauend auf lose definierten und authentischen Aufgabenstellungen Probleme einbeziehen, deren Lösung ein grenzüberschreitendes Lernen benötigt. Grenzen sind dabei unterschiedliche Disziplinen, der Bruch zwischen traditionellen und aktuellen Strukturen bzw. Rollen, sowie unterschiedliche Formen des Lernens. Cremers et al. [2016] sehen zudem Überschneidungen mit *Context-based Learning Environments* (siehe De Putter-Smits et al. [2013]), *Powerful Learning Environments* (siehe De Bruijn and Leeman [2011]) und *Authentic Learning Environments* (siehe Herrington and Oliver [2000]), da neben authentischen und selbstorganisierten Lernprozessen auch die Förderung einer beruflichen Identität angestrebt wird.

---

### 1.6.3. Seamless Learning Environments

Seamless Learning wurde in Abschnitt 1.5.6 als wichtige Eigenschaft bzw. als Ziel eines Systems für hybride Lehre erkannt, sodass es auch bei der Definition hybrider Lernräume zu beachten ist. Looi et al. [2010] beschäftigen sich in diesem Zusammenhang mit den Charakteristiken von *Seamless Learning Environments*. Wie schon beim Seamless Learning erkennen sie einen starken Bezug zu mobilen Techniken und dem *Mobile Learning* [vgl. Looi et al., 2010, S. 155]. Das größte Problem dieser Lehre sehen sie in der vorherrschenden Trennung zwischen formalen und informellen Umgebungen, da ein Seamless Learning Environment nur durch die Überbrückung dieser Dichotomie bzw. durch die kombinierte Nutzung der unterschiedlichen Kontexte und Umgebungen entstehen kann. Besonders die Mobilität, Vielseitigkeit und Konnektivität von mobilen Geräten sind dabei treibende Faktoren, um einen didaktischen Wandel vom lehrerzentrierten zum lernendenzentrierten Paradigma zu vollziehen, sodass den Lernenden in Seamless Learning Environments durch die Wahl solcher Technologien ein aktiver Lernprozess ermöglicht wird [vgl. Looi et al., 2010, S. 156]. So können auch weitere Probleme traditioneller Lehre angegangen werden, namentlich die Dekontextualisierung von Informationen, sowie indirektes und abstraktes Wissen durch Erfahrungen aus zweiter Hand. Als wichtige Grundlage für solche Lernumgebungen erachten Looi et al. [vgl. 2010, S. 159 f.] die Theorie der verteilten Wahrnehmung von Hollan et al. [2000]. Diese nennen drei Prinzipien für kognitive Prozesse:

1. Sie sind über die Mitglieder der sozialen Gruppe verteilt.
2. Sie sind über die Zeit verteilt.
3. Der Betrieb eines kognitiven Systems beinhaltet die Koordination zwischen interner und externer (materieller oder Umwelt-) Struktur.

Dieser Annahme entsprechend wird in Seamless Learning Environments privates und öffentliches Lernen überbrückt, sodass Lernprozesse sowohl individuell als auch kollaborativ über verschiedene Kontexte und Zeitpunkte hinweg vollzogen werden können. Individuelles Lernen geschieht dabei in privaten Lernräumen und kollaboratives Lernen in öffentlichen Lernräumen. Zusätzlich finden sich kognitive Artefakte, die innerhalb eines Kontextes über Zeit und physische oder virtuelle Räume hinweg mithilfe von Technologie Wissen vermitteln. Lernräume sind also nicht länger gebunden an einzelne physische Räume und vielmehr abhängig von den Lernaktivitäten [vgl. Looi et al., 2010, S. 156 ff.]. Lernprozesse können dabei auch von Lernenden selbst angestoßen und von festen Vorgaben losgelöst improvisiert werden. Mithilfe mobiler Geräte erstellen sie digitale Artefakte, welche sie teilen, überarbeiten, erweitern und kombinieren können, um dadurch neues Wissen zu schaffen. Somit besitzen sowohl physische als auch

digitale und zeitbedingte Faktoren eine gravierende Rolle innerhalb solcher Lernumgebungen, jedoch entfalten sich ihre Vorteile nur, wenn sie flexibel und kontextspezifisch eingesetzt werden.

#### **1.6.4. Arbeitsdefinition**

Wie schon bei der hybriden Lehre sind auch für hybride Lernräume in den unterschiedlichen Ansätzen starke Gemeinsamkeiten, jedoch auch einzelne Alleinstellungsmerkmale zu erkennen.

Blended Spaces nach Benyon lassen sich als mehrschichtige Räume zusammenfassen, welche anhand generischer Eigenschaften wie der Ontologie, Topologie, Volatility und Agency charakterisiert werden. Sie ermöglichen die Integration neuer digitaler Räume in physische Umgebungen, woraus eine neue Wahrnehmung des Kontextes und ein neues Gefühl der Präsenz resultieren, durch die ebenso neue Ideen und Erkenntnisse begünstigt werden.

Im Zusammenhang mit Begriffsdefinitionen von *Hybrid Learning Spaces (HLS)* ist ein Umdenken bestehender Strukturen ersichtlich. Neben veränderten Rollen und Wahrnehmungen der Lehre ist auch ein stärkerer Bezug zu Raum-Dimensionen und davon verkörperten Präsenzen zu erkennen. Hybride Lernräume scheinen demnach Orte zu sein, welche für bestimmte Kontexte geschaffen werden und ähnlich einer Medienökologie unterschiedliche Aktivitäten, Medien, Objekte und Funktionen bereitstellen. Die Verwendung und Kombination dieser Elemente ist den Nutzern freigestellt und wird von diesen zielorientiert selbst bestimmt. Bei der Planung wird die Überbrückung von Dichotomien und Disziplinen angestrebt, jedoch können Elemente stets ihrem Ausgangspunkt zugeordnet werden. Dafür müssen sowohl Grenzen als auch Gemeinsamkeiten identifiziert werden. Das Ziel solcher Räume ist die Einführung einer zielorientierten *Maker-Kultur*, sowie die Förderung von teamorientiertem, selbstgesteuertem und informellem Lernen. Dafür müssen Lernende maßgeblich beim Entstehungsprozess bedacht oder sogar integriert werden. Hybride Lernräume unterliegen deshalb einer stetigen Anpassung an die Bedürfnisse der Nutzer, wobei der Prozess des Erstellens und Nutzens für das Lernen häufig wichtiger ist als das Ergebnis selbst.

Seamless Learning Environments verkörpern wiederum den starken Bezug zu mobilen Geräten, welche durch ihre Flexibilität und Konnektivität ein dynamisches und lernendenzentriertes Lernen ermöglichen. Dabei ist die Überbrückung bzw. Kombination von formaler und informeller Lehre ein Kernkonzept, durch welches eine aktivere Rolle der



Lernenden gefördert wird. Ein größerer Praxisbezug wechselt das zuvor abstrakte und dekontextualisierte Wissen ab, wofür sowohl individuelle als auch kollaborative Lernprozesse zu unterstützen sind. Neben physischen und digitalen Umgebungen spielt die zeitliche Flexibilität eine erhebliche Rolle, ebenso wie Artefakte zur Überbrückung der unterschiedlichen Dimensionen. Diese Artefakte werden von Lernenden selbst erstellt und kontrolliert, um ein autonomes Lernen sowie die kontextgerechte Schaffung von Wissen zu ermöglichen.

Unter Berücksichtigung der genannten Begriffsverständnisse wird im weiteren Verlauf der Ausarbeitung folgende Definition für hybride Lernräume angenommen:

#### Definition 1.2: Hybride Lehr-/Lernräume

*Hybride Lehr-/Lernräume verkörpern den Gedanken hybrider Lehre (siehe Definition 1.1) und ermöglichen dieses Lehrparadigma für gezielte Kontexte. Lernende erhalten die Chance, ihre eigene Lehre zu steuern, wozu physische, digitale und kombinierte Räume kontextgerecht ebenso physische und digitale Komponenten als Ontologie bereitstellen. Auch wenn Aktionen innerhalb klar definierter Grenzen ermöglicht und dadurch gezielte Verhaltens- und Interaktionsmuster gefördert werden, entscheidet nicht allein die Ontologie über die Topologie des jeweiligen Lernraums. Im Sinne von Medienökologien können bereitgestellte Komponenten von Lernenden situationsbasiert genutzt, sowie abhängig von eigenen Präferenzen, Fähigkeiten, Wertvorstellungen oder Zielsetzungen kombiniert werden, wodurch sie dem Raum aktiv eine eigene Bedeutung geben und an der Formung des eigenen Lernprozesses beteiligt sind. Hierzu sollten hybride Lehr- / Lernräume ebenso die Integration persönlicher Medienökologien, also vielgenutzter Dienste, Geräte und Materialien der Lernenden ermöglichen. So unterliegen hybride Lernräume einem stetigen Wandel, in welchem sie an die Bedürfnisse der Lehrenden und Lernenden, sowie den technischen und gesellschaftlichen Wandel angepasst werden.*

--- Dennis Dubbert

Aufbauend auf dieser Definition bleibt festzuhalten, dass ein hybrider Lehr- / Lernraum zwar geplant und vorbereitet werden kann, jedoch erst dessen Interpretation durch Lernende und Lehrende, sowie die Art der Nutzung beinhaltenen Objekte über den wirklichen Charakter des Raums entscheidet. So können auch alltägliche Räume zu einem hybriden Lernraum werden, beispielsweise wenn sich Lernende in einem *Café* treffen, über das campuseigene *Learning Management System (LMS)* Aufgabenstellungen erhalten, über das jeweilige Thema diskutieren und *Videos* oder *Live-Streams* einer Vorlesung schauen, Schlagworte im Sinne eines Brainstormings in eigenen *Notizblöcken* sammeln und anschließend gemeinsam mithilfe ihrer *Smartphones* oder *Laptops* in einem geteilten *Google Docs Dokument* Notizen für die folgende Ausarbeitung festhalten. In diesem Fall wurde ein einfacher sozialer Raum von den Lernenden, beispielsweise aufgrund der bereitgestellten Sitzcken, als passendes Setting für eine Gruppenarbeit erkannt, mithilfe vorgeschriebener und selbstgewählter physischer sowie digitaler Komponenten angereichert und so als hybrider Lernraum genutzt.

Diesem Beispiel folgend gilt es zudem eine häufige Fehlinterpretation anzusprechen, derer nach *Learning Management Systems (LMS)* die digitale Repräsentation hybrider Lernräume darstellen. Die Aufgabenstellung eines LMS ist der eines hybriden Lernraums in der Tat ähnlich. So definiert e-teaching.org [2021b] diese wie folgt:

„Die Aufgabe eines LMS liegt oft in der Abbildung bestimmter Lernstrukturen. Kurse können angelegt, Materialien hochgeladen, Kommunikationswerkzeuge aktiviert und Studierende für diese Elemente freigeschaltet werden.“ [e-teaching.org, 2021b]

Doch bemerkt e-teaching.org [2021b] auch gleichzeitig kritisch, dass „[...] dieser Ansatz die Gestaltung des Lernens auch limitiert, da nicht der Lernende selbst, sondern der Lehrende bzw. die Hochschule die Konzeption und Organisation der Umgebung bestimmen“. Gerade dieser Aspekt ist bei hybriden Lernräumen grundlegend verschieden. Während LMS statische Vorgehensmodelle darstellen und ermöglichen, sind hybride Lernräume für flexible und adaptive Lernprozesse nutzbar, in denen Lernende nicht nur an der Lehre partizipieren können, sondern auch aktiv bei dessen individueller Gestaltung involviert sind. Dabei gehen hybride Lernräume über schulische Strukturen hinaus und beziehen auch interdisziplinäre, private und gesellschaftliche Komponenten ein. LMS stellen somit nur einen Teilaspekt hybrider Lehr- / Lernräume dar, weshalb sie weniger als Synonym und mehr als Bestandteil solcher Räume anzusehen sind.

## 2. Eigenschaften und Entitäten hybrider Lehre

Die Arbeitsdefinitionen bieten bereits eine Eingrenzung des Themengebietes. Es fehlen jedoch noch beschreibende Übersichtsmodelle des Lehr- / Lernansatzes, welche auch die Eingliederung eines unterstützenden Systems in das Paradigma thematisieren. Hierfür muss zunächst identifiziert werden, wie ein System als unterstützendes Medium zu nutzen ist und welche Möglichkeiten es dafür bereitstellen sollte. Als erster, notwendiger Schritt gilt somit die Ermittlung der Kerneigenschaften und Charakteristiken hybrider Lehre, sowie der grundlegenden Komponenten und ihrer Relationen. Diese Aspekte können anschließend in einem Übersichtsmodell visualisiert werden und so als Basis späterer Systemspezifikation dienen.

Es finden sich einige Modelle auf unterschiedlichen Abstraktionsebenen und aus unterschiedlichen Betrachtungswinkeln. Viele dieser Quellen fokussieren sich auf die Konzeption einfacher Lernräume, doch sind auch deren Eigenschaften bei der Erstellung hybrider Lernräume hilfreich. Es finden sich jedoch auch Modelle, welche sich speziell auf die bereits angedeuteten Dimensionen und Dichotomien hybrider Lehre und Lernräume beziehen. Eine Auswahl all dieser Modelle wird folgend analysiert, mit dem Ziel, dort gewonnene Informationen in einem umfassenden Modell zusammenzufassen. In vorherigen Abschnitten wurden zudem bereits wichtige Aspekte identifiziert, welche bei der Analyse beachtet werden und so ebenfalls in das Modell einfließen.

### 2.1. Grundlegende Modelle

Als Erstes muss ein Verständnis für die grundlegenden Bestandteile hybrider Lehre aufgebaut werden. So gilt es besonders zu verstehen, welche Rollen Räume und Technologien in dem Paradigma einnehmen und in welcher Beziehung diese stehen.

In Kapitel 1.4 wurde bereits angesprochen, dass die Pädagogik, das Konzept von Räumen und technologische Entwicklungen wichtige Bestandteile bzw. Entitäten oder Komponenten hybrider Lernräume sind. Ebenso sieht es Radcliffe [2009]. In seinem

*Pedagogy–Space–Technology Framework* (siehe Abbildung 2.1) nennt er diese als Kernbestandteile eines Lernraums und beschreibt ihr Zusammenspiel. Die Pädagogik (Pe-

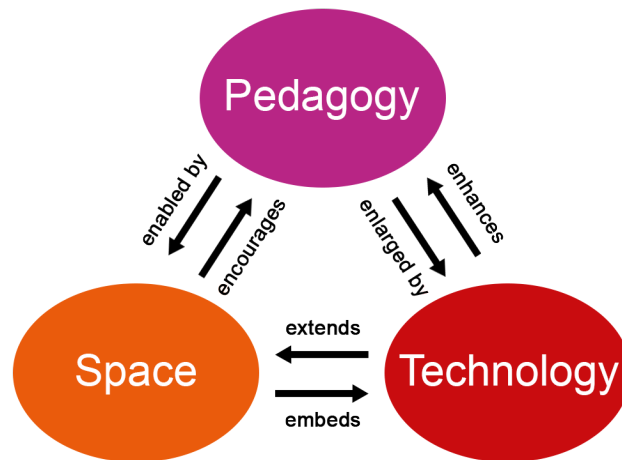


Abbildung 2.1.: Pedagogy–Space–Technology Framework, inhaltlich entnommen aus Radcliffe [2009]

dagogy) beschreibt dabei den Typus des Lernens. Sie entscheidet, welche Paradigmen, Unterrichtsformen und Methodiken zu nutzen sind. Ermöglicht wird sie durch den Raum (Space), welcher hier die physische Infrastruktur darstellt [vgl. Radcliffe et al., 2008, S. 3]. Die Objekte und Möbel des Raums, sowie ihre Anordnung verkörpern einen pädagogischen Gedanken und regen dadurch wiederum bestimmte Paradigmen, Unterrichtsformen und Methodiken an. Ähnlich zu Benyon and Mival [2012] (siehe Kapitel 1.6.1) nennen also auch Radcliffe et al. [2008] die Ontologie und Topologie als wichtige Faktoren eines Raums. Die Technologie erweitert nun diesen Raum und schafft für diesen neue Möglichkeiten. Ebenso ist diese aber auch in dem Raum integriert, wodurch sie dieselben Eigenschaften verkörpern muss. Bestimmte räumliche Gegebenheiten können dabei zu Einschränkungen oder Bevorzugungen führen. So fördert die Technologie im angemessenen Kontext auch pädagogische Herangehensweisen, für welche sie ebenfalls neue Erweiterungen und Möglichkeiten schafft. Sie kann demnach als eine treibende Kraft hinter Lernräumen gesehen werden. Zusammengefasst bedeutet dieses Modell also, dass Unterrichtsformen und Methoden einer Pädagogik durch den Raum darstellbar und durch die Technologie des Raums ermöglicht werden. Dabei gelten alle wichtigen Charakteristiken der anderen Komponenten auch als Anforderungen der Technologie, da diese den Raum und die Pädagogik unterstützt und widerspiegelt.

In Bezug zu hybriden Lernräumen sieht Trentin [2015] die Hybridisierung des Raums als entscheidend. Demzufolge besteht ein hybrider Lernraum nicht nur aus einem physi-

schen Raum, sondern auch aus einem virtuellen Pendant. Als wichtig erachtet er zudem die Lernaktivitäten als eigenständige Komponente, deren Zusammenspiel mit den beiden Raumtypen für einen Lernraum unerlässlich ist. Die Möglichkeiten für Lernaktivitäten hängen dabei stark von dem gewählten Raum ab und erst die Kombination dieser unterschiedlichen Möglichkeiten erzeugt den hybriden Gedanken. So können die hier als Lernaktivitäten betitelten Arbeitsschritte mit Methodiken der Pädagogik-Komponente vorheriger Modelle verglichen werden. Die Komponenten und ihre Beziehungen visualisiert er in den folgenden zwei Modellen (Abbildung 2.2).

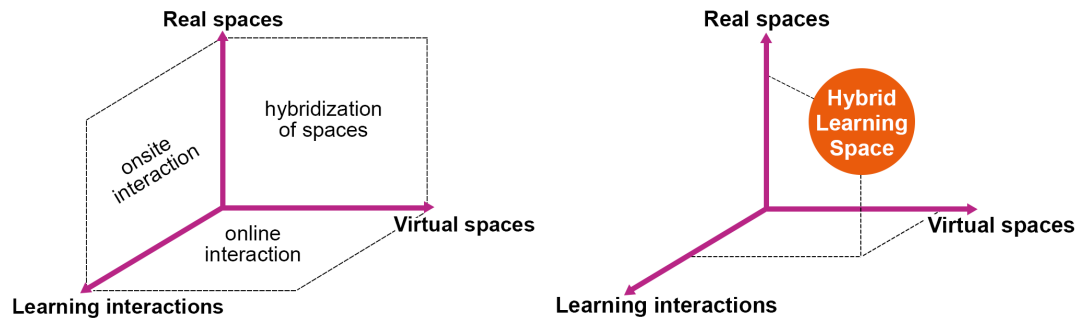


Abbildung 2.2.: Komponenten hybrider Lernräume und deren Beziehungen, inhaltlich entnommen aus Trentin [2015, S. 3]

Auch Ellis and Goodyear [2016] erstellten ein abstrahiertes Modell für Lernräume, in welchem sie weitere Komponenten bedenken (siehe Abbildung 2.3).

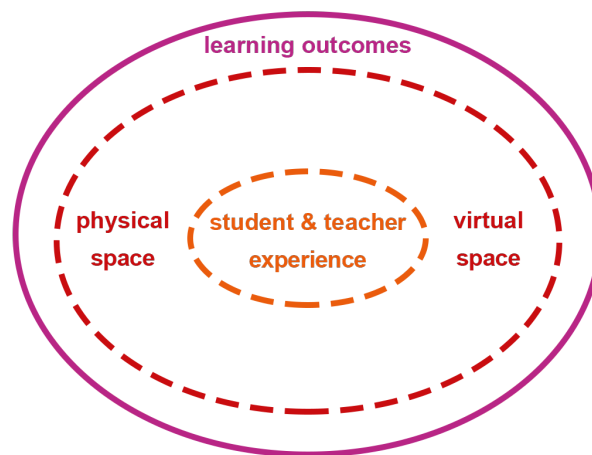


Abbildung 2.3.: Abstrahiertes Modell von Lernräumen, inhaltlich entnommen aus Ellis and Goodyear [2016, S. 175]

Ihr Modell besteht aus mehreren Schichten, deren Zentrum die Lehrenden und Lernenden bilden. Diese erleben innerhalb des Lernraums eine bestimmte Lernerfahrung,

zu deren Zweck sie sowohl den physischen als auch den virtuellen Raum nutzen. Diese Räume befinden sich auf der mittleren Ebene. Auf der äußersten Ebene stehen die angestrebten *Learning Outcomes*. Die Reihenfolge dieser Schichten gibt an, welche Bestandteile zur Erreichung einer Schicht notwendig sind und welche in einem direkten Kontakt stehen. So basieren physische und virtuelle Räume auf den Personen des Raumes und müssen in Abhängigkeit der benötigten Lernerfahrungen erstellt werden. Ebenso sind Learning Outcomes abhängig von den Räumen und dadurch auch von den Lernerfahrungen. Räume sind demnach nicht nur auf die Lernerfahrungen sondern auch speziell auf das jeweilige Learning Outcome abzustimmen. Auffällig ist zudem der durchgezogene äußere Rand, welcher die Abgeschlossenheit des Lernraumes repräsentiert. So ist dieser stets kontextabhängig bzw. auf ein bestimmtes Learning Outcome ausgerichtet, was bereits in Kapitel 1.4 beschrieben wurde. Die Autoren Ellis and Goodyear [2016] ergänzen hier jedoch, dass ein Learning Outcome auch die Kombination unterschiedlicher Zwischenziele darstellen kann und somit auch verschachtelte Lernräume nutzbar sind:

„[...] the structure of learning space involves mutually shaping interactions between different kinds of nested learning spaces (i.e. spaces within spaces) and different kinds of nested learning purposes.“[Ellis and Goodyear, 2016, S. 174]

Übertragen auf eine technische Perspektive bedeutet dieses Modell, dass Lehrende und Lernende neben dem physischen Raum auch ergänzende digitale Systeme nutzen können müssen, deren Funktionsumfang und Lernmaterialien / Ressourcen auf gewünschte Lernerfahrung ausgerichtet sind. Solche Systeme müssen auch Möglichkeiten der Repräsentation von Learning Outcomes bieten. Angestrebtes Wissen kann dabei digital in Form von unterschiedlichen Artefakten festgehalten werden, beispielsweise über Textdokumente, Bilder und andere kontextgerechte Repräsentationsformen.

Diese Grundmodelle verdeutlichen einige Kernaspekte für technische Systeme, welche innerhalb von Lernräumen eingesetzt werden. So sind in ihnen bereits unterschiedliche Bestandteile solch eines Systems angedeutet. Zunächst sind Repräsentationen für Lehrende und Lernende, also für die beinhalteten Personen und ihre Rollen beim Lernen notwendig. Betrachtet man die neue Aufgabenverteilung bezüglich hybrider Lehre aus Tabelle 1.3, so haben Lehrende in dem System eine vorwiegend anleitende und betreuende Rolle, während Lernende als proaktive Schaffer und Gestalter des eigenen Lernens gelten. Dies ist bei der pädagogischen Komponente zu berücksichtigen, welche sich in Form der benötigten Lernmethoden äußert. Lernerfahrungen bilden sich dann als Kombination von Methoden und daran partizipierenden Personen, sodass auch ih-

---

re Interaktionen und Kommunikationen zu beachten sind, was von Benyon and Mival [2012] als Agency betitelt wird (siehe Kapitel 1.6.1). Lernerfahrungen werden in dem Modell von Ellis and Goodyear [2016] mit der Erreichung eines Learning Outcomes in Verbindung gebracht, also als Prozess dargestellt. Bereits in Kapitel 1.4 wurde der Lernprozess als wichtige Komponente hybrider Lehre identifiziert, sodass auch dieser vom System als eigenständige Entität berücksichtigt werden sollte. So könnte dieser als eigenständiges Learning Outcome definiert werden, da die Identifikation erreichten Wissens in vorherigen Kapiteln mehr mit ihm als mit den Ergebnissen selbst in Verbindung gebracht wurde. Dennoch sind auch solche Ergebnisse als Learning Outcome von Relevanz, sodass auch die Erstellung unterschiedlicher Artefakttypen angeboten werden muss. Die Darstellung des Raums selbst basiert auf dessen Infrastruktur beziehungsweise der Ontologie und Topologie. Dabei sind sowohl physische, als auch digitale Objekte des Raums zentral, wozu aus technischer Sicht beispielsweise Ressourcen / Lernmaterialien und Teilfunktionen gehören. Ebenso sind technische Geräte als Zugangsmedium und Bestandteile der Räume entscheidend. Da Räume auch kombiniert und verschachtelt werden können, gelten zudem Kombinationen verschiedener Teilanwendungen und beinhaltete Interaktionsmöglichkeiten zwischen diesen Anwendungen als wichtige Komponenten. Bezüglich des Raums weist das Modell von Ellis and Goodyear [2016] zudem auf die kombinierte Nutzung physischer und virtueller Räume hin. In Abschnitt 1.6.2 wurde bereits eine bidirektionale Einbindung der unterschiedlichen Räume angesprochen. Demnach sollte auch das System, neben der eigenen Eingliederung in physische Räume, die Überführung des Physischen ins Virtuelle unterstützen.

Letztlich wurde in den Modellen verdeutlicht, dass die Eigenschaften der Technologie von den Anforderungen der Pädagogik und der darzustellenden Lernräume abhängig sind. Auch in Kapitel 1.4 wurde bereits die Mehrdimensionalität hybrider Lehre und Lernräume angesprochen, diese Dimensionen jedoch nur oberflächlich betrachtet. Somit gilt als nächster notwendiger Schritt die Analyse dieser Dimensionen, um so Charakteristiken hybrider Lehre zu identifizieren und mit den Entitäten zu verbinden.

## 2.2. Dichotomien hybrider Lehre

Bereits bei den Definitionsbildungen fanden sich Bezüge zu unterschiedlichen Präsenzformen und Dimensionen, welche jedoch zunächst abstrakt und oberflächlich beschrieben wurden. Zwei der meistzitierten Quellen auf *Google Scholar* präsentieren nun solche Dimensionen in Form von binären Skalen, welche der Beschreibung hybrider Settings dienen können. Bonk and Graham [2012, S. 20] nennen 13 solcher Dichotomien aus Sicht des Blended Learnings und Stommel [2012] 17 Dichotomien in Bezug zur Hybrid

Pedagogy, welche alle in Anhang A einsehbar sind. Diese Dichotomien veranschaulichen die Reichweite der Ansätze und erinnern daran, dass bei der Planung eines Hybriden keine eindimensionale Betrachtung des Themenfeldes erfolgen darf:

„Instead of a single blend that calls for x percent of this and y percent of that, I propose we take the blends of many of the authors here into account. We must replace one-dimensional thinking with simultaneous consideration of dozens of pie charts.“[Bonk and Graham, 2012, S. 21]

Ebenso bedeuten diese binären Skalen nicht, dass es nur zwei Ausprägungen gibt. Ganz im Gegenteil sollen die Extrema lediglich eine Eingrenzung bieten und zur Kombination anregen, da Hybridität von Natur aus eine Mischform darstellt. Für ein System bedeutet dies nun, dass es funktional die Extrema und Sonderformen unterstützen muss, jedoch auch flexible Kombinationen zur Schaffung neuer Mischformen möglich sein sollten. Eine genaue Eingrenzung notwendiger Systemeigenschaften und -entitäten ist über diese beiden Modelle jedoch nicht möglich, da sie stark verallgemeinert sind, lediglich abstrahierte Ausprägungen didaktischer Herangehensweisen darstellen und weder Erläuterungen noch beschreibende Beispiele gegeben sind. In dem folgenden Abschnitt werden deshalb Modelle fokussiert, welche die Kernbestandteile eines (hybriden) Lernraums greifbar machen, ihre Extraktion ermöglichen und eine praktische Übertragbarkeit auf unterstützende Systeme aufweisen.

### **2.3. Dimensionen als Systemeigenschaften und -entitäten**

Durch den starken technischen Bezug des Blended und Hybrid Learnings werden häufig andere Attribute des Ansatzes übersehen. Doch schon frühe Modelle beschreiben auch abseits der Technologie zahlreiche Dimensionen. Die folgenden Unterabschnitte befassen sich nun mit der Analyse solcher Modelle, um die wichtigsten Dimensionen hybrider Lehre zu identifizieren. Dimensionen können dabei sowohl Entitäten, als auch Eigenschaften repräsentieren. So werden diese für jede Quelle zunächst als Schlüsselbegriffe festgehalten und ihre Bedeutungen innerhalb des jeweiligen Kontextes beschrieben. Gemeinsamkeiten zwischen Modellen werden dabei bereits kurz angesprochen. Eine abschließende Extraktion der Entitäten und Eigenschaften, sowie die vollständige Zusammenführung übereinstimmender Dimensionen findet jedoch erst in Abschnitt 2.4 statt. Für eine bessere Separierung der Ergebnisse einzelner Quellen werden diese folgend in eigenständigen Unterkapiteln präsentiert.



### Smythe [2012]: Toward a framework for evaluating blended learning

Smythe [2012, S. 1] nennt in seiner Veröffentlichung vier wichtige Dimensionen des Blended Learnings. Demnach ist zunächst die Modalität der Wissensübertragung entscheidend, welche traditionelle (z.B. face-to-face) oder webbasierte Ausprägungen annehmen kann. Neben den Kombinationen eingesetzter Technologien sind auch die Kombinationen eingesetzter pädagogischer Ansätze und Methodiken für ihn eigenständige Dimensionen. Letztlich nennt er die Chronologie beziehungsweise Synchronität des gewählten Ansatzes als Dimension.

### Trentin [2015]: Orientating pedagogy towards hybrid learning spaces

Trentin [2015] präsentiert, aufbauend auf den in Kapitel 2.1 vorgestellten Basismodellen, ein weiteres Modell mit drei Dimensionen, welche jeweils über eine Dichotomie spezifiziert sind (siehe Abbildung 2.4). Als beschreibende Dimensionen erkennt er den Lernprozess, das Setting bzw. den Ort des Lernens und den Lernraum selbst. Ein Lernprozess kann dabei individuell oder kollaborativ sein, das Setting schulisch oder außerschulisch und der Raum physisch vor Ort oder online. Auf Basis dieser Dimensionen und Dichotomien definiert er vier Kategorien möglicher Aktivitäten, welche im Laufe des Lernens sequentiell und frei kombinierbar sind (siehe Abbildung 2.4, linke Seite). Weiterhin beschäftigt sich Trentin [2015, S. 10] mit der Kategorisierung hybrider Lerntypen. Dafür sieht er die Dimensionen des Lernprozesses und des Lernraums als ausschlaggebend (siehe Abbildung 2.4, rechte Seite) und unterscheidet zwischen Online-Kollaborativem Lernen (I), Vor-Ort-Kollaborativem Lernen (II), Vor-Ort-Individuellem Lernen (III) und Online-Individuellem Lernen (IV).

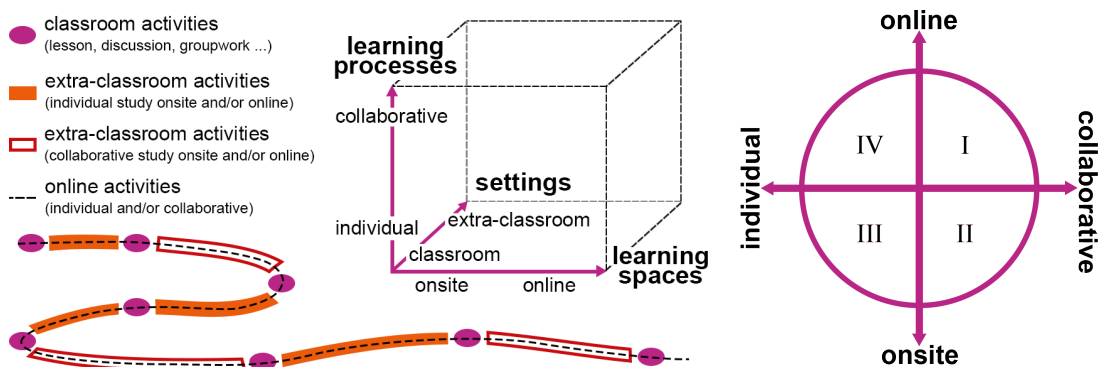


Abbildung 2.4.: Dimensionen hybrider Lernräume und Aktivitätstypen, inhaltlich entnommen aus Trentin [2015, S. 9]

### Graham [2006]: Blended learning systems

Weitere Dimensionen nennt Graham [2006, S. 7], allem voran die Dimension des Raumes (Space), welche die Ausprägungen *Live* (physisch / face-to-face), *gemischte Realität*, sowie *virtuell* bzw. *verteilt* einnehmen kann. Ebenso gilt für ihn die Zeit als Dimension, sodass Lehre synchron oder asynchron erfolgen kann. Als dritte Dimension führt er die Natürlichkeit bzw. Authentizität (Fidelity) des Lernens auf, welche hoch (spricht alle Sinne an), medium (z.B. Audio) oder niedrig sein kann (z.B. nur Text). Letztlich nennt er die Menschlichkeit bzw. Sozialität als vierte Dimension, geprägt durch die Menge an Menschen- oder Maschinenkontakt. Dieses Modell nutzen Suo and Shi [2008] als Grundlage und erweitern es um die Dimension der Sprachunterschiede, derer nach sämtliche Muttersprachen erlaubt oder eine Gruppenisolation in Abhängigkeit der Muttersprache stattfinden kann. Zudem ergänzen sie die Ausprägungen der anderen Dimensionen jeweils mit einer zugehörigen Synchronität und der dafür genutzten Modalität (traditionell oder E-learning). Das ausführlichere Modell ist in Abbildung 2.5 dargestellt.

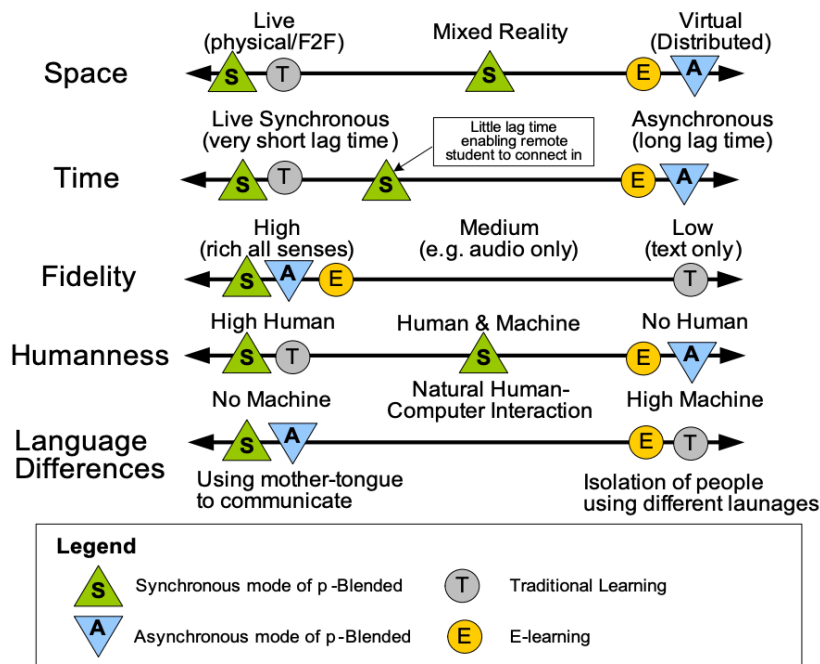


Abbildung 2.5.: Dimensionen des Blended Learnings, entnommen aus Suo and Shi [2008, S. 196]

**Miliszewska [2008]: Transnational education programs - Student reflections on a fully-online versus a hybrid model**

Miliszewska [2008, S. 82] ordnet die Ausprägungen von vier beschreibenden Dimensionen unterschiedlichen Lernstrategien bzw. Lernformen zu, welche selbst als fünfte Dimension dargestellt werden. Als Lernstrategien gelten für sie dabei die Laborarbeit, von Lehrenden angeleitetes Lernen, das Lernen durch ein Praktikum und das Selbststudium. Als beschreibende Dimensionen gilt dann zunächst der Übertragungsweg von Wissen (delivery mode), welcher in diesem Dokument stets als Modalität bezeichnet wird. Zugehörige Ausprägungen hierzu sind online und Face-to-Face. Weiterhin nennt sie die Zeit bzw. Synchronität. Die dritte Dimension bilden Lernobjekte, zuvor als Ressourcen bezeichnet, für welche die Ausprägungen Simulation, Video und Audio festgehalten sind. Letztlich erwähnt sie das Tempo bzw. die Steuerung des Lernens (Pace), welche selbstgesteuert oder gruppenorientiert sein kann.

**Reinmann [2021]: Hybride Lehre – Ein Begriff und seine Zukunft für Forschung und Praxis**

Den Aspekt der Steuerung erfasst auch Reinmann [2021, S. 5], nennt diesen jedoch *Aufteilung von Gruppenteilnehmern*. Als mögliche Unterformen dieser Dimension listet sie das Plenum ohne Aufteilung, die Aufteilung in eine Vor-Ort und eine Online-Gruppe, sowie die Aufteilung in zwei oder mehr Gruppen ohne Fixierung der Modalität auf. Weiterhin gilt für sie der Rythmus von Präsenzveranstaltungen, sowohl physisch als auch digital, als Dimension, welche die Ausprägungen vollständig und regelmäßig, reduziert und regelmäßig oder reduziert und punktuell einnehmen kann. Letztlich nennt sie die Einbindung von digitalen Lehr- / Lernmaterialien, welche optional als Ergänzung der Präsenz, teilweise als Ersatz der Präsenz oder vollständig als Ersatz der Präsenz stattfinden kann. So nimmt sie durch den letzten Aspekt wiederum Bezug zu dem Multi Access-Learning aus Abschnitt 1.5.2. Zusätzlich zu diesen didaktischen Dimensionen gelten jedoch auch für sie die Modalität und Zeit als weitere Dimensionen, wobei zeitliche Faktoren sowohl die Synchronität der Modalitäten, als auch Parallelität der Teilnehmenden beinhalten.

**Cunningham and Tabur [2012]: Learning Space Attributes - Reflections on Academic Library Design and Its Use**

Noch detaillierter gehen Cunningham and Tabur [2012] bei der Definition einer Dimensionspyramide für Lernräume vor (siehe Abbildung 2.6), welche auf den vier Charakte-

ristiken erfolgreicher öffentlicher Plätze von Kent and Myrick [2003] und der *hierarchy of needs* von Maslow [1943] aufbaut. Bei dieser Pyramide werden obere Schichten erst dann ermöglicht, wenn alle darunter liegenden Schichten die Bedürfnisse der Lernenden abdecken. Werden so alle Schichten berücksichtigt, ermöglicht dies die Erstellung eines idealen Lernraums [vgl. Cunningham and Tabur, 2012, S. 1].

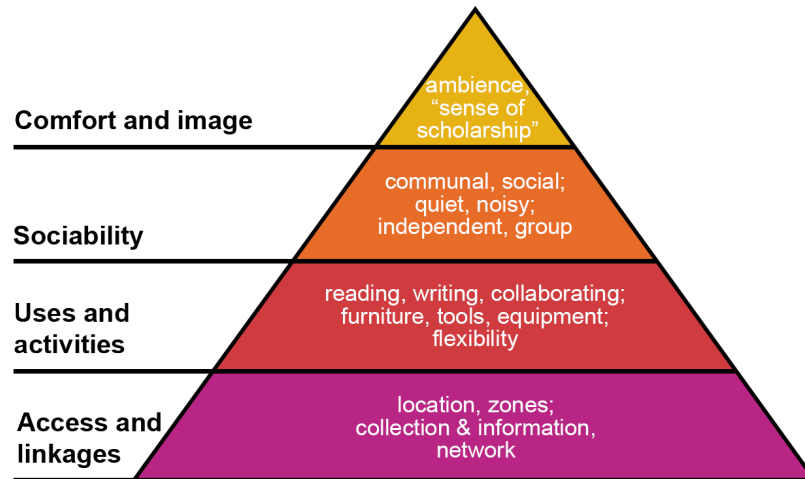


Abbildung 2.6.: Dimensionspyramide: Hierarchie von Lernraumattributen, inhaltlich entnommen aus Cunningham and Tabur [2012, S. 1]

Als unterste Dimension nennen sie den Zugang zu und die Verknüpfung von Lernräumen (*Access and linkages*). Neben dem Ort als physischem Zugang und Endgeräten als virtuellem Zugang liegt ein besonderer Fokus auf der Bereitstellung und Verknüpfung von Unterräumen / Zonen zur Unterstützung verschiedener Lernmethoden. Als Beispiele werden Räume für ruhiges und individuelles Lernen, für Gruppenarbeit oder für „geselliges Beisammensein“ genannt. Lernende müssen im Zuge aufeinander aufbauender Arbeitsschritte nahtlos zwischen Räumen wechseln können, wie es bereits in Abschnitt 1.5.5 mit dem Begriff der *Seamless Connectivity* angesprochen wurde. Lernressourcen müssen effektiv und effizient erreichbar sein, sowie frei kombinierbar. Dazu zählen andere Lernende, Informationstechnologien und Web-Inhalte, sowie flexible Raumkonfigurationen für Lernende und Lehrende [Cunningham and Tabur, 2012, S. 2]. Auch vielgenutzte externe Informationsquellen müssen erreichbar sein.

Die zweite Dimension repräsentieren Aktivitäten und ihre Nutzung im jeweiligen Kontext (*Uses and activities*) [Cunningham and Tabur, 2012, S. 2f.]. Für die Vielzahl unterschiedlicher Aktivitäten müssen entsprechende Räume, Ausstattungen und Anwendungen bereitgestellt werden. Laut Cunningham and Tabur [2012] werden Grup-

penarbeiten verstärkt eingesetzt, sodass kollaborative Räume notwendig sind. Doch auch stille und individuelle Räume sind weiterhin von Relevanz. Sowohl akademische Räume wie Klassenzimmer, als auch außerakademische Räume wie Cafés oder Museen werden genutzt. Ihren flexiblen und modularen Einsatz sehen die Autoren als zentral. Lehrenden und Lernenden sollte eine Umkonfiguration des Settings ad-hoc ermöglicht werden, um auch zukünftig wechselnden Nutzungen und Aktionen gerecht zu werden.

Die Ebene der Soziabilität (*Sociability*) umfasst sämtliche Interaktionen mit und Einbindungen von anderen Personen [Cunningham and Tabur, 2012, S. 3]. So müssen soziale Räume bereitgestellt werden, welche die Verflechtung von Wissensschaffung und Geselligkeit ermöglichen. Dennoch unterstreichen die Autoren, dass stets der Bedarf nach stillen Räumen für ein Selbststudium herrscht. So sollte wiederum ein Wechsel zwischen unabhängigen, individuellen Räumen und gruppenbasierten, sozialen Räumen möglich sein.

Letztlich wird die Dimension des Komforts und Images genannt (*Comfort and image*), welche schwer quanti- und qualifiziert werden kann, jedoch für die Motivation der Lernenden entscheidend ist [Cunningham and Tabur, 2012, S. 3f.]. Die Lernumgebung sollte komfortabel, aber auch wissenschaftlich sein. Die in Abschnitt 1.5.6 angesprochene Authentizität und Aktualität spielt dabei eine entscheidende Rolle. Bekannte Strukturen ermöglichen Komfort, sodass bereits genutzte Anwendungen und Informationsquellen, sowie authentische und kontextgerechte Arbeitsschritte einfließen sollten.

### **Zhang [2008]: Hybrid learning and ubiquitous learning**

Auch Zhang [2008, S. 253] unterscheidet bei der Charakterisierung von Blended Learning aufgrund verschiedener Dimensionen. Ebenso wie das Modell von Cunningham and Tabur [2012] beinhaltet auch sein Ansatz die Steuerbarkeit oder Kollaborationsabhängigkeit des Lernprozesses. Modalitätsunterschiede zwischen online und offline sieht er als weitere Dimension. Weiterhin unterscheidet er anhand der Anpassung von Lerninhalten (abgestimmter Inhalt und serienmäßiger Inhalt), der Kombination von Lernen und Arbeiten, sowie der Struktur der Lehre (strukturiert und unstrukturiert).

### Radcliffe et al. [2008]: Designing next generation places of learning - Collaboration at the pedagogy-space-technology nexus

Die Struktur der Lernumgebung ist auch im *Place for Learning - Spectrum* von Radcliffe et al. [2008, S. 12 ff.] eine entscheidende Dimension (siehe Abbildung 2.7). Den Raum sehen sie dabei als ein hochgradig verbundenes Netzwerk von Lernorten und nicht als eine Reihe von Lernsilos. In diesem Spektrum zeigen die Autoren die Vielfalt unterschiedlicher Lernraumtypen, welche im Hochschulökosystem zum Einsatz kommen oder dieses erweitern. So reichen sie von unstrukturierten Parks oder Cafés, über semi strukturierte Bibliotheken, bis hin zu vollstrukturierten Hörsälen. Die Strukturiertheit eines Raums erscheint dadurch als *Dimension höherer Ordnung*, welche mehrere Unterdimensionen wie Formalität oder Privatheit vereint und auch von anderen Dimensionen wie Authentizität beeinflusst wird.

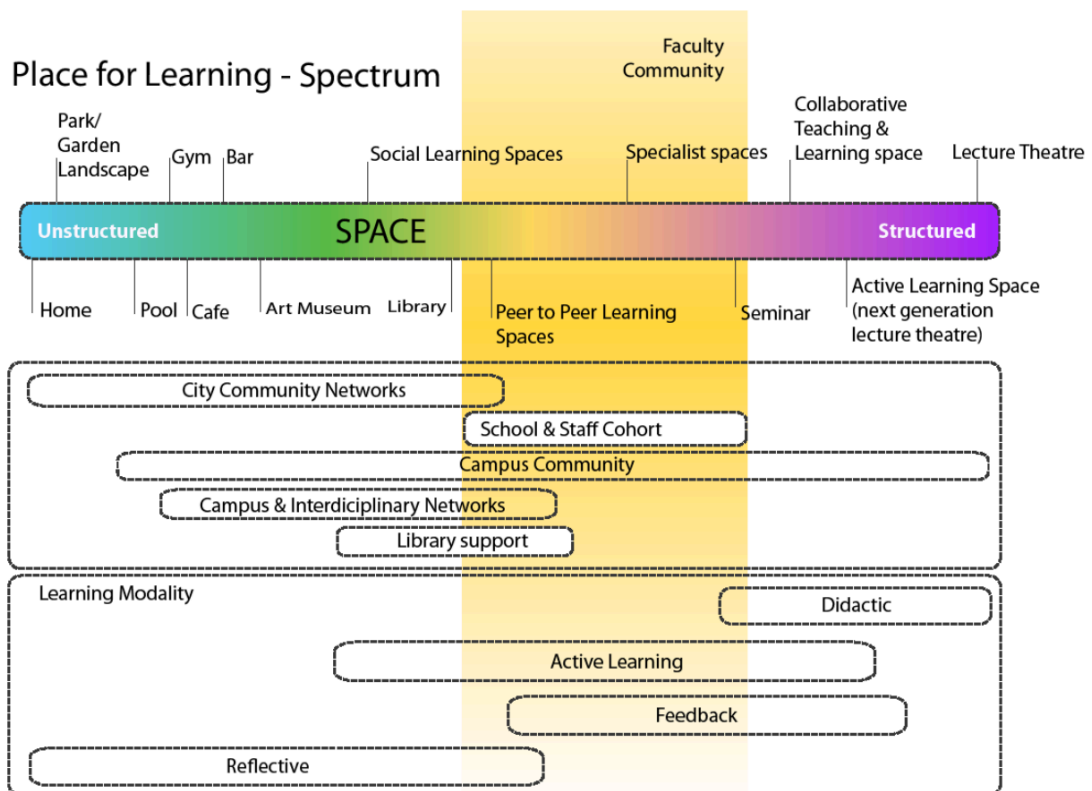


Abbildung 2.7.: Place for Learning Spectrum, entnommen aus Radcliffe et al. [2008, S. 13]

Die Autoren fokussieren in dem Modell zwei weitere Dimensionen, deren Ausprägungen sie jeweils mit der Strukturiertheit eines Raums in Verbindung setzen. Eine dieser Dimensionen ist die Lernmodalität. Dieser Begriff gilt hierbei jedoch nicht als Un-

terscheidung zwischen physischem und digitalen Lernen, wie es in diesem Dokument bisher der Fall war, sondern eher als Unterscheidung verschiedener Aktivitätstypen. Sie verbinden strukturierte Lernräume mit didaktischem Lernen, unstrukturierte Räume mit reflektivem Lernen und Zwischenstadien mit aktivem Lernen und Feedback.

Als zweite ergänzende Dimension präsentieren sie *Communities*. Während des Lernprozesses befinden sich Lernende in unterschiedlichen Communities, welche sie begleiten, beim jeweiligen Lernschritt unterstützen und allgemein ein Interesse am Erfolg des zugehörigen Lernraums aufweisen. Diese Communities gehen über fakultätseigene Gesellschaften hinaus, sodass auch Stadtgemeinschaften und persönliche Internet-Communities aus *Social Media* oder Foren einbezogen sind.

### **Mossavar-Rahmani and Larson-Daugherty [2007]: Supporting the hybrid learning model - A new proposition**

Mit Lern-Communities befassen sich auch Mossavar-Rahmani and Larson-Daugherty [2007] in ihrem *Online Hybrid Learning Model* (siehe Abbildung 2.8), in welchem sie die wichtigsten Teilnehmer solcher Communities sammeln.

Im Zentrum steht dabei der Lernende selbst, welcher im Laufe seines Lernprozesses Zugriff auf alle anderen Bestandteile der Community hat. Die *Faculty* stellt hierbei die Lehre mit bzw. den Kontakt zu Lehrenden dar, welcher Face-to-Face, über einen Live Chat oder asynchrone Wege wie E-Mails hergestellt werden kann. Im Zuge des Lernens erhalten Lernende zudem über virtuelle Bibliotheken oder ein virtuelles Tutoring Zugriff auf akademische Ressourcen. Ebenso wird Kursmaterial über File-Hosting- oder Cloud-Systeme für das Selbststudium oder Gruppenarbeiten bereitgestellt. Über das Assessment werden Lernfortschritte von Lehrenden verfolgt und Artefakte bewertet. Zusätzlich beinhaltet die Lern-Community administrative, systembedingte und technische Supportfunktionen (*Concierge*), sowie einen virtuellen Campus, welcher soziale Interaktionen abseits einer Gruppenarbeit bereitstellt. Für die Autoren ist die Bereitstellung solch einer effektiven Online-Community ein Schlüsselement des Erfolgs hybrider Lehre [vgl. Mossavar-Rahmani and Larson-Daugherty, 2007, S. 3]. Effektiv sind diese jedoch nur, wenn sie auf die Bedürfnisse der Lernenden abgestimmt werden.

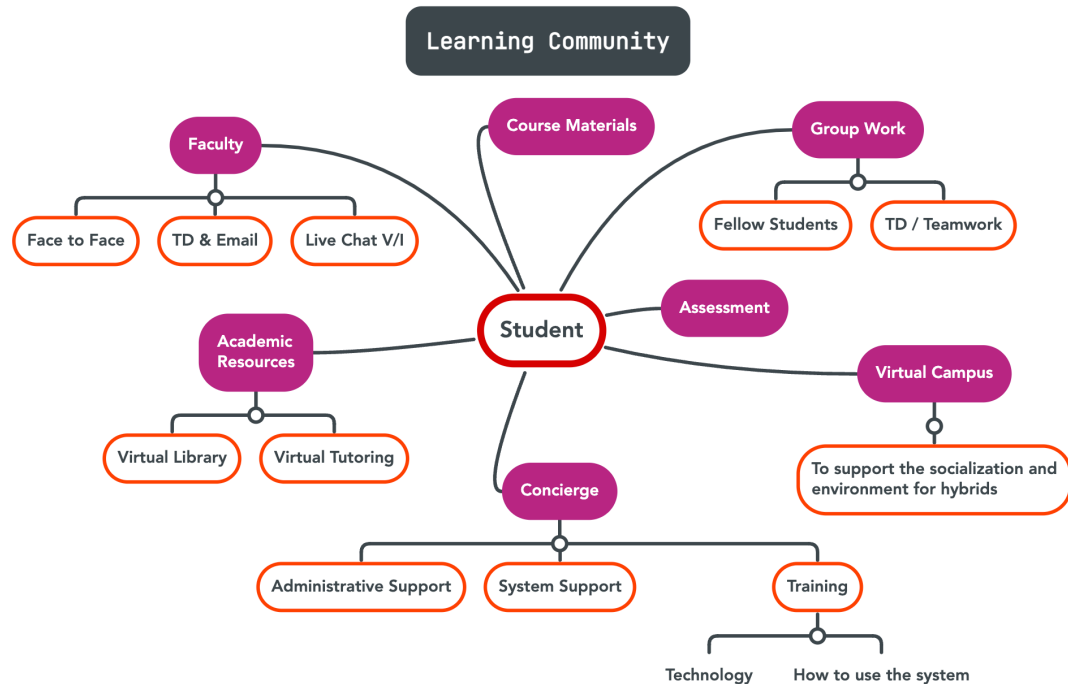


Abbildung 2.8.: Online Hybrid Learning Model, inhaltlich entnommen aus Mossavar-Rahmani and Larson-Daugherty [2007, S. 3]

### Wong and Looi [2011]: What seems do we remove in mobile-assisted seamless learning? A critical review of the literature

Abschließend müssen auch Modelle aus dem Bereich des Seamless Learnings bedacht werden. Wong and Looi [vgl. 2011, S. 8 ff.] identifizieren auch für diesen Bereich zehn verschiedene Dimensionen, welche für das Design einer effektiven Systemlösung notwendig sind. Laut ihrer ersten Dimension „Encompassing formal and informal learning“ müssen Lernende in der Lage sein, selbstständige und gemeinsame Erarbeitungen von Lerninhalten mit traditioneller Frontallehre zu verbinden. Mit „Encompassing personalized and social learning“ heben sie hervor, dass sowohl individuelle als auch kollaborative Artefakterstellungen ermöglicht werden und diese Prozesse stark miteinander verflochten sind. Weiterhin nennen die Autoren eine Orts- und Zeitunabhängigkeit. Lernräume werden dynamisch im alltäglichen Leben und Lernen erstellt und genutzt, sodass beispielsweise ein Café mithilfe mobiler Endgeräte ad-hoc zum hybriden Lernraum wird. Ebenso müssen Lernfortschritte, -ressourcen oder -artefakte beim Wechsel des Raumes erhalten bleiben und stets erreichbar sein („Ubiquitous access to learning resources“). Mit „Encompassing physical and digital worlds“ nennen die Autoren die Integration von physischen und digitalen Bestandteilen innerhalb hybrider Lehre. Phy-



sische und soziale Welten stehen in einer steten Interaktion, sodass alltägliche Erfahrungen in das Lernen einbezogen werden können. Laut der Dimension „Combined use of multiple devices“ ist für eine kollaborative Erarbeitung von Wissen die Konnektivität mehrerer Geräte notwendig, darunter mobile Geräte, Desktop Computer oder interaktive Whiteboards. Social Media stellt beispielsweise eine bereits vorhandene Möglichkeit dieser Verbindung dar, über welche ein Wissensaustausch ermöglicht wird. Auch die *Seamless Connectivity* ist wieder ein wichtiger Aspekt, welcher durch die achte Dimension „Seamless switching between multiple learning tasks“ widergespiegelt wird. Demnach soll ein nahtloser Wechsel von Arbeitsschritten und dafür genutzten Geräten möglich sein, sodass ein geschlossener, jedoch vielschichtiger und adaptiver Lernprozess möglich ist. Die Lernergebnisse müssen nach der Dimension „Knowledge synthesis“ zu jedem Punkt im Lernprozess vernetzt werden können. So sind Kombinationen des bestehenden und neuen Wissens, verschiedene Ebenen von Denkfähigkeiten und -abstraktionen, sowie multidisziplinäres Lernen zu unterstützen. Letztlich nennen die Autoren das Prinzip „Encompassing multiple pedagogical learning activity models“, welches wiederum die Vielschichtigkeit zu unterstützender Lernmethoden, sowie ihre flexible Kombination anspricht. Jeder Lernraum ermöglicht demnach die Erreichung multipler Learning Outcomes und kann hierzu auf unterschiedliche und angepasste Weise verwendet werden.

## **2.4. Identifikation gemeinsamer Kernaspekte**

Aufbauend auf den identifizierten Dimensionen der vorherigen Kapitel kann nun die Extraktion und Zusammenführung übereinstimmender sowie sich ergänzender Entitäten und Eigenschaften hybrider Lehre und Lernräume stattfinden. Diese werden in den folgenden Unterkapiteln separiert betrachtet und eine jeweilige Übersicht der Kernaspekte präsentiert.

### **2.4.1. Übersicht identifizierter Entitäten**

Ähnlich den generellen Modellen finden sich auch in den dimensionsbasierten Modellen einige Gemeinsamkeiten bezüglich der Bestandteile bzw. Entitäten hybrider Lehre. Diese werden folgend genannt, beschrieben und in Beziehung gesetzt. Nach einer groben Übersicht findet sich zudem eine detailliertere Darstellung der Gesamtübersicht, welche zur Verdeutlichung beispielhafte Ausprägungen jeder Komponente beinhaltet.

## Entitäten und ihre Beziehungen

Die analysierten Quellen zeigten eine große Varietät der Bestandteile hybrider Lehre. Trotz variierender Terminologien wurden bei der Analyse auch Gemeinsamkeiten ersichtlich. Folgend findet nun eine Auflistung und Beschreibung der wichtigsten Kerntitäten hybrider Lehre sowie ihrer internen Beziehungen statt.

Als wichtige, übergreifende Konzepte gelten zunächst die Lernerfahrungen bzw. der Lernprozess, der Lernraum, die Lernmethode, Lernmaterialien / Ressourcen, Artefakte und Personen. Von der Pädagogik werden dabei übergeordnete und didaktisch vorgegebene Learning Outcomes definiert, auf deren Basis geeignete Kombinationen von Lernformen, Lernmethoden und Aufgabenstellungen zu wählen sind. Nun beginnt der Lernprozess, in welchem Lernende unterschiedliche, aufeinander aufbauende Arbeitsschritte vollziehen und dadurch Lernerfahrungen sammeln. Für diese Arbeitsschritte stehen Lernenden unterschiedliche Arten von physischen und digitalen Ressourcen / Lernmaterialien zur Verfügung. So sind physische Bücher, Zettel oder Notizen als Ressource nutzbar, jedoch auch digitale Videos, Dokumente oder Websites. Einige Arbeitsschritte dienen zudem der Artefakt-Erstellung als Zwischenstand oder endgültige Ergebnis-Dokumentation, wobei auch diese wiederum physisch oder digital sein können und als Ressource für weitere Arbeitsschritte nutzbar sind (siehe Abbildung 2.9).

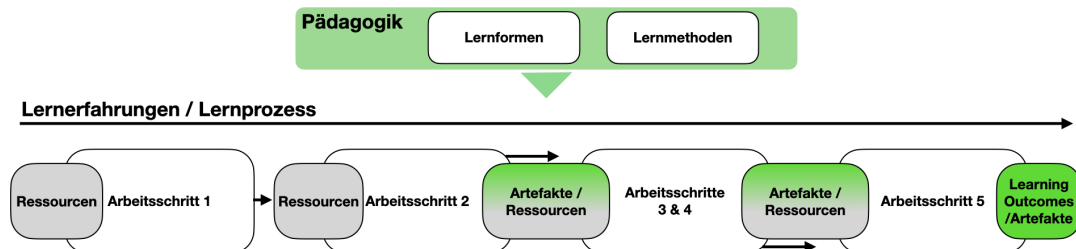


Abbildung 2.9.: Ablauf eines Lernprozesses

Im Zuge der Arbeitsschritte wechseln Lernende zudem zwischen unterschiedlichen Settings, sowie physischen und digitalen Lernräumen, welche für den jeweiligen Schritt kontextgerecht optimiert sind. Ihre Nutzung kann dabei parallel oder sequentiell geschehen (siehe Abbildung 2.10, Arbeitsschritte eins und zwei). Ebenso ist für Arbeitsschritte auch nur eine dieser Raumarten nutzbar und Lernräume sind ineinander verschachtelbar (siehe Abbildung 2.10, Arbeitsschritte drei, vier und fünf). Solche Verschachtelungen können dem Kontext entsprechend beliebig oft oder tief erfolgen.

Ein Beispiel solcher Unterraumbildungen ist eine physische Bibliothek, in welcher sowohl kleine Arbeitsnischen für individuelle Arbeiten als auch große Tischgruppen für kollaborative Tätigkeiten vorhanden sind, wobei sich auch diese wieder in Unterabschnitte aufteilen lassen. Ebenso könnte innerhalb einer digitalen Veranstaltung für unterschiedliche Gruppen ein vorbereiteter Unterraum bereitgestellt werden, um dort eine isolierte Erarbeitung von Aufgabenstellungen zu ermöglichen. Hierdurch würde auch der Lernprozess für jede Gruppe eine Abzweigung darstellen und bei einer späteren Zusammenführung der Ergebnisse wieder mit dem geteilten Prozess verschmelzen.

Ein wichtiger Aspekt ist zudem, dass sowohl physische, als auch digitale Räume wiederverwendet und unterschiedlich kombiniert werden können. So kann in verschiedenen physischen Räumen schnell eine ähnliche Lernumgebung geschaffen werden, welche verlässliche Ressourcen und Funktionen bereitstellt. Ein Beispiel hierfür ist die *Meeting Owl*<sup>1</sup>, durch welche jeder Raum für Online-Konferenzen optimiert wird. Anders herum ist jedoch auch derselbe physische Raum für unterschiedliche Arbeitsschritte nutzbar, indem er mit mindestens einem anderen und auf den Schritt optimierten digitalen Raum kombiniert wird.

Eine beispielhafte Nutzung unterschiedlicher Lernräume während des Lernprozesses ist in Abbildung 2.10 visualisiert.

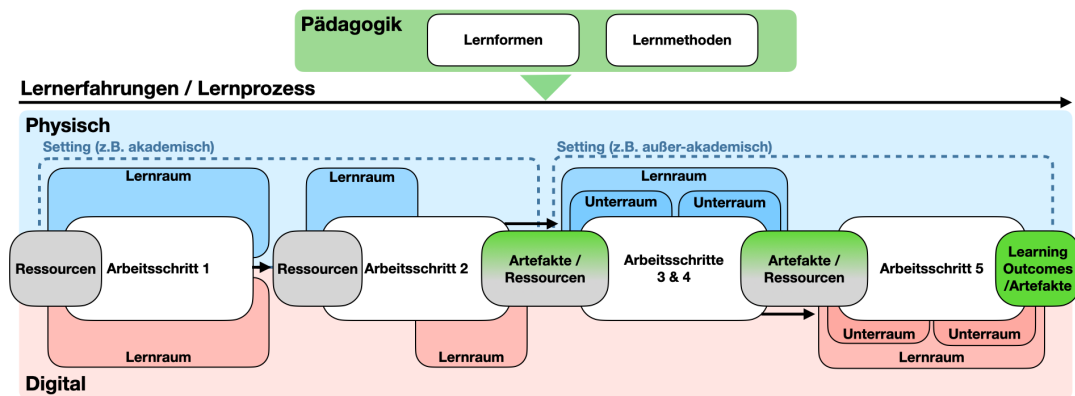


Abbildung 2.10.: Nutzung von Lernräumen innerhalb eines Lernprozesses

Je nach Raumart, physisch oder digital, finden sich zudem weitere beinhaltete und beschreibende Entitäten. Viele Quellen unterscheiden bezüglich der physischen Komponenten zwischen dem Begriff des Lernraums und Lernsettings. Während der Lern-

<sup>1</sup>Meeting Owl: Siehe <https://checkout.owllabs.com/products/meeting-owl-pro>, zuletzt gesichtet 23.08.2021

raum durch seine Infrastruktur (Ontologie und Topologie), beteiligte Personen und deren Rollen- / Aufgabenverteilungen (Agency) charakterisiert wird, befasst sich das Lernsetting mit dem Kontext in dem sich dieser Lernraum befindet, sowie dem damit verbundenen *Image*. So kann dieses beispielsweise schulisch, außerschulisch, wissenschaftlich oder gesellschaftlich wirken, wodurch Motivationen und Anreize zu unterschiedlichen Arbeitsschritten geschaffen werden.

Der digitale Lernraum wird wiederum durch beteiligte Personen und ihre Rollen / Aufgaben innerhalb des Raums beschrieben, wobei diese von denen im physischen Raum abweichen können. Als Erweiterung der menschlichen Komponente werden sowohl im physischen, als auch digitalen Kontext häufig Communities genannt, welche ebenso am Lernprozess teilnehmen können. Letztlich sind im digitalen Raum als Äquivalent zu der Infrastruktur einerseits beinhaltet Anwendungen, Tools bzw. Software-Komponenten eines Systems zu nennen und andererseits virtuelle Speicher und Quellen zu beachten, welche Ressourcen und Artefakte bereitstellen bzw. halten können.

Eine zusammenfassende Gesamtübersicht dieser Entitäten sowie ihrer Zusammenhänge wird in Abbildung 2.11 präsentiert.

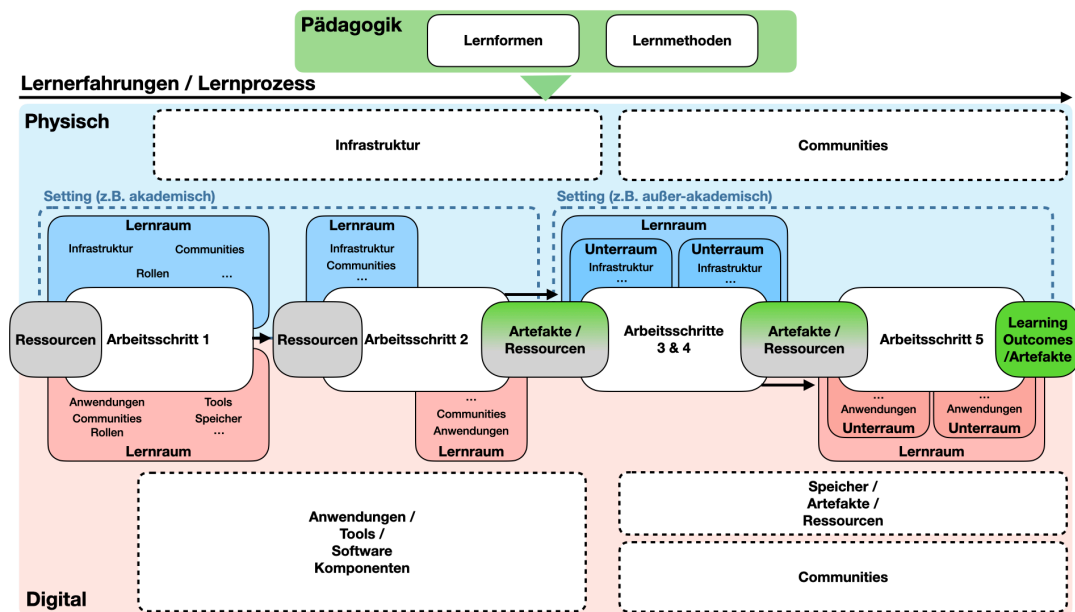


Abbildung 2.11.: Gesamtübersicht identifizierter Entitäten hybrider Lehre und ihre Zusammenhänge

## Mögliche Ausprägungen der Entitäten

Für einige der identifizierten Entitäten sind in den untersuchten Quellen auch spezifischere Ausprägungen aufgelistet, welche folgend beispielhaft beschrieben werden.

Zunächst wurden die Teilnehmenden einer Veranstaltung spezifiziert, allem voran die Communities. Diese können neben campuseigenen Personen und Gruppen auch externe Instanzen wie Unternehmen oder Social Media einbeziehen. Bezüglich digitaler Systeme bedeutet dies beispielsweise, dass ein System neben der registrierten Nutzung auch eine ad-hoc Einbindung anonymer Nutzer für einzelne Arbeitsschritte ermöglichen sollte.

Auch wurden genauere Informationen zur Infrastruktur physischer Lernräume gegeben, welche sich demnach selbst aus unterschiedlichen Komponenten zusammensetzt. So sind beim Design geeignete Garnituren auszuwählen und bereitgestellte Geräte zu beachten. Letztere ermöglichen erst den Zugang zu dem digitalen Raum, sodass ihre Auswahl eine große Auswirkung auf funktionale Möglichkeiten technischer Systeme, sowie auf ihre Vernetzung und Integration in den physischen Raum hat. Vorlesungsräume sind beispielsweise oft mit Beamern, Bildschirmen oder interaktiven Whiteboards ausgestattet, welche digitale Objekte im physischen Raum anzeigen können oder sogar eine Interaktion mit diesen ermöglichen. Zu der Infrastruktur physischer Räume zählen zudem sämtliche Objekte und Materialien, mit denen Lernende interagieren können oder welche zur Schaffung einer gewollten Atmosphäre (Setting) dienen. Diese sind nicht zuletzt interessant, da sie oft die Grundlage der Überführung zwischen physischen und digitalen Räumen darstellen. So sind auch besonders physische Ressourcen zu beachten, welche im Lernraum angeboten werden und für die eine digitale Repräsentation neue Möglichkeiten bietet. Notizen auf einem normalen Whiteboard müssen beispielsweise für eine Weiterverwendung zunächst abgeschrieben werden, während Notizen von einer virtuellen Fläche direkt kopiert und modifiziert werden können.

Für digitale Ressourcen und Artefakte wurden zudem unterschiedliche Formen genannt. So können diese unter anderem als Simulation, in Form eines Videos, als Audio-Datei oder in Textform vorliegen. Bei Ressourcen ist zudem zwischen allgemeinen Lernressourcen, ähnlich einer digitalen Bibliothek, und kursspezifischen Materialien zu unterscheiden. Zur Unterstützung der Lehre können beispielsweise Lernvideos bzw. Screencasts für vorlesungsspezifische Inhalte angeboten werden. Ebenso ist die Nutzung digitaler Skripte als Textdokument bzw. PDF eine gängige Praktik. Für allgemeine Informationen zur selbstständigen Erarbeitung hingegen werden häufig Bücher

empfohlen, welche in der campuseigenen Bibliothek in physischer oder digitaler Form bereit stehen. Sowohl physische, als auch digitale Ressourcen weisen somit unterschiedliche Anwendungszwecke auf und können sich innerhalb einer Veranstaltung ergänzen.

Bei digitalen Objekte sind zudem die unterschiedlichen Speichermöglichkeiten zu beachten. Laut dem Vergleich mit Medienökologien aus Abschnitt 1.6.2 sind dafür neben systemeigenen Datenspeichern auch externe, bereits etablierte File-Hosting Anbieter einzuplanen. Häufig verwendete Speicherdienste wie Dropbox oder Sciebo erleichtern die Verteilung von Dokumenten, da sie bereits in den Tagesablauf vieler Lernenden integriert sind und spezielle Software sowohl für Computer, als auch Smartphones vorhanden ist. Für die Verteilung von Screencasts oder anderen Videos eignet sich hingegen besonders eine bestehende Video-Plattform wie Youtube, welche ebenfalls einen täglichen Begleiter vieler Lernender darstellt. Je nach Inhalt und Format können also andere Dienste sinnvoll sein, wobei auch stets die bestehende Infrastruktur des Lehrinstituts, die Vorkenntnisse und Gewohnheiten der Lernenden und allgemeine rechtliche Einschränkungen zu beachten sind.

Bezüglich der Datenhaltung ist deshalb auch die Sichtbarkeit (privat vs. geteilt) der Ressourcen und Artefakte zu bedenken, da diese beispielsweise geschützte Daten beinhalten können oder nur von einem Kurs gesehen werden sollen. Ebenso ist die Verteilung an eine breitere Community denkbar, welcher die Objekte im digitalen, aber auch im physischen Raum bereitgestellt werden könnten, beispielsweise über öffentliche Monitore. Zudem müssen auch Lehrende im Zuge des Assessments auf diese Zugriff haben. Als Assessment wurden in den Quellen sämtliche Tätigkeiten gezählt, deren Ziel eine Prüfung oder Bewertung der Lernfortschritte und Artefakte ist, wodurch sie gerade für digitale Systeme eine besondere Form von Anwendungen darstellen. Auch Supportfunktionen wurden in diesem Zusammenhang genannt.

Planungen solcher und anderer Anwendungen eines Systems sollten ebenfalls das Prinzip der Medienökologien beachten. So sollten neben systemeigenen, dediziert für die Lehre geschaffenen Tools auch solche aus den privaten Medienökologien der Nutzer und Lehrinstitute vernetzt werden. Eine Kommunikation zwischen Lernenden und Lehrenden, zwischen Gruppenteilnehmern oder ganzen Communities kann beispielsweise über bekannte Social Media Kanäle geschehen. Grundlegende Dokumentation können von Gruppen in Online-Tools wie Google Docs erstellt werden, zu deren Bearbeitung auch Lehrende eingeladen werden können, sodass diese eine direkte Einsicht in Fortschritte und Ergebnisse haben. Ebenso können in einer Vorlesung bestehende Anwendungen wie Miro für ein gemeinsames Brainstorming genutzt werden und die geteilten Ergeb-

nisse anschließend als Grundlage für Einzelarbeiten dienen. Es finden sich viele Anwendungsfälle bereits etablierter Software-Produkte innerhalb eines Lehr- / Lernszenarios.

Dabei ist auch die datengenerierende Funktion hybrider Lernräume zu erwähnen, welche bereits in Abschnitt 1.6.2 angesprochen wurde. Alle Interaktionen innerhalb eines Raums generieren Metadaten, sei es der Zeitpunkt einer Tätigkeit, dessen Dauer, beteiligte Personen, Umstrukturierungen eines Raumes oder Iterationen / Versionen eines Artefakts. Diese Daten stellen selbst wiederum ein Ergebnis der Nutzung dar, aus welchem sowohl für eine akute Veranstaltung, beispielsweise in Form intelligenter tutorieller Systeme, als auch für weitere Designprozesse wichtige Erkenntnisse hervorgehen können. Somit finden sich neben aktiv geschaffenen Artefakten auch solche passiven Artefakte, in welchen automatisiert Nutzungsdaten gesammelt werden. Für die Sammlung und Auswertung der Daten, sowie den Aufbau geeigneter *Lerner-* und *Wissensmodelle* sind dann unter anderem Algorithmen und Entitäten aus dem Feld der *Learning Analytics* (siehe beispielsweise Slater et al. [2016]) notwendig, welche ebenfalls im Systemdesign ermöglicht werden sollten.

Eine spezifischere Visualisierung der identifizierten Bestandteile mit beispielhaften Ausprägungen ist nun abschließend in Abbildung 2.12 dargestellt.

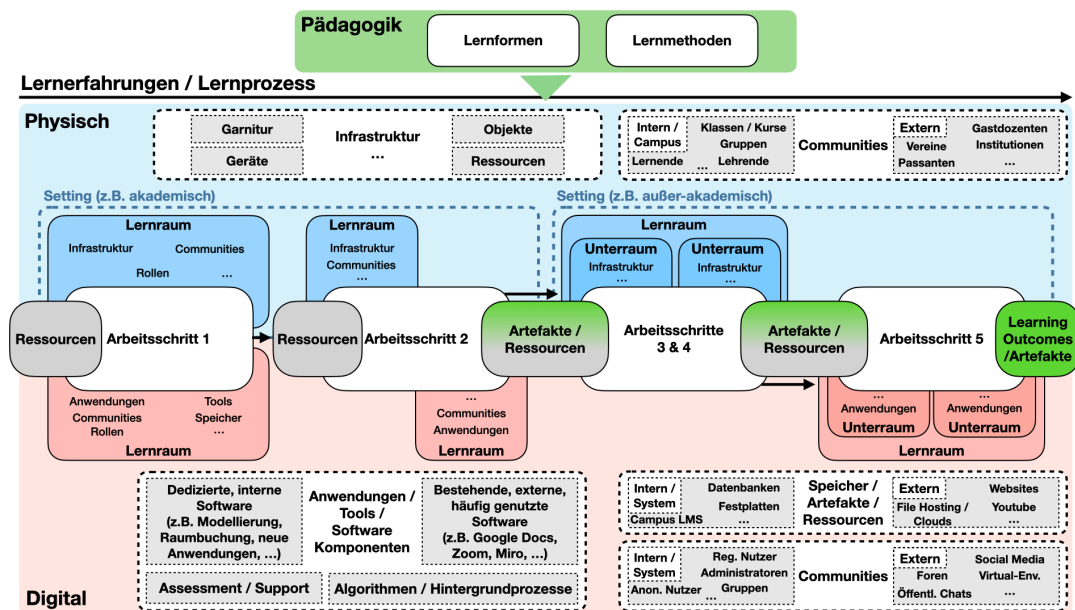


Abbildung 2.12.: Identifizierte Entitäten hybrider Lehre und ihre Zusammenhänge, mit Beispielen

### 2.4.2. Übersicht identifizierter Eigenschaften

Neben der Identifikation der Bestandteile hybrider Lehre ist das Ziel vieler Quellen vor allem die Definition typisierender bzw. kategorisierender Raum-Eigenschaften und Faktoren, um diese als Grundlage des Raumdesigns nutzen zu können.

Bezüglich dieser Eigenschaften wird in nahezu allen Modellen der Übertragungsweg von Wissen und Ressourcen bzw. deren Modalität genannt, welche physische und/oder digitale Ausprägungen annehmen kann. Bezüglich der Einbindung digitaler Lernmaterialien wird zudem zwischen dem optionalen Einsatz als Ergänzung der Präsenz, dem gelegentlichen Einsatz als dessen Ersatz und dem vollständigen Ersatz von Präsenz unterschieden. Weiterhin wird oft die Synchronität der Modalitäten genannt. In Bezug zur Zeit wird zudem die Parallelität der Präsenz von Teilnehmern einer Veranstaltung oder Gruppenmitgliedern erwähnt, welche bereits in der Definition hybrider Lehre beachtet wurde (siehe Abschnitt 1.5.6).

Die Authentizität bzw. Natürlichkeit eines hybriden Lernraums gilt als weitere Dimension und ist von einigen Faktoren abhängig. So ist die Repräsentationsform von Wissen als Ressourcen entscheidend, also wie natürlich diese wirken und wie viele Sinne sie ansprechen. Ebenso ist die Authentizität der Arbeitsschritte und Interaktionen im jeweiligen Kontext entscheidend, wozu auch die Integration aktueller, persönlicher und etablierter Anwendungen beiträgt. Somit ist auch die Nutzerorientierung bzw. Adaptivität charakterisierend, wobei Lerninhalte und Lernräume auf die Bedürfnisse einzelner Teilnehmer (*Seamless Adaptivity*, siehe Abschnitt 1.5.5), einer gezielten Gemeinschaft oder auf ein abstraktes Plenum abgestimmt werden können.

Damit verbunden sind Charakteristiken der Kollektivität. Zunächst ist zu wählen, ob eine individuelle Tätigkeit oder ein gruppenbasiertes Lernen stattfinden soll. Weiterhin sind hier die Interdisziplinarität der Lehre und dessen Offenheit gegenüber Neuerungen zu nennen, welche bereits in Kapitel 1.4 angesprochen wurden. Es können fächerübergreifende Themenbereiche in der Lehre behandelt werden. Auch die Zusammenarbeit unterschiedlicher Forschungsrichtungen bzw. Teams mit variierenden Hintergründen und Fachkenntnissen ist in diesem Zusammenhang möglich. Hierbei muss auch die Transparenz des Lernens bedacht werden. Zu unterscheiden ist, ob Ressourcen, Lernprozesse und Artefakte persönlich bleiben, sichtbar für weitere Personen gemacht werden oder sogar einer größeren Gemeinschaft bereitzustellen sind.



So zählen auch die Erreichbarkeit und Konnektivität als ausschlaggebende Charakteristiken. Es sollten kontextgerechte Zugangsmethoden und Verknüpfungen von Lernräumen vorhanden sein. Damit ist auch eine Orts- und Zeitunabhängigkeit des Zugriffs verbunden. Technisch ist zusätzlich zu klären, welchen Geräten ein Zugriff ermöglicht wird und wie sich diese ergänzen. Dabei ist wiederum die Seamless Connectivity zu nennen, derer nach Räume keine isolierten Silos sind, sondern in einem Netzwerk logische Verknüpfungen aufweisen. So müssen auch Geräte kombinierbar sein, um eine optimale Konnektivität zu gewährleisten. Auch für Lerninhalte, Ressourcen und resultierende Artefakte sollten solche Zusammenführungen bzw. Verknüpfungen ermöglicht werden. Sie müssen jederzeit erreichbar und zwischen den Räumen kompatibel sein, um einen fortlaufenden, sich ergänzenden Lernprozess zu ermöglichen.

Dem Ansatz folgend ist die Struktur bzw. der Strukturierungsgrad des Lernraums, des Lernprozesses und der Lerninhalte zu bedenken. Settings können von schulischen Vorlesungsräumen bis hin zu außerschulischen Cafés reichen und die Lehre innerhalb dieser Räume variiert von strukturiert bis unstrukturiert. Die unterschiedlichen Ausprägungen bevorzugen dann ebenso unterschiedliche Formen des Lernens, sei es formal oder informell. Die Strukturiertheit von Lernräumen wird auch durch die Steuerbarkeit des Lernens beeinflusst. Lernen kann dabei selbstgesteuert, gruppenorientiert oder geleitet erfolgen. Im Falle eines kollektiven Lernens in Gruppen ist auch die Gruppeneinteilung wichtig, welche laut Reinmann [2021] drei Ausprägungen einnehmen kann: Plenum ohne Aufteilung, Aufteilung in eine Vor-Ort- und eine Online-Gruppe, sowie Aufteilung in zwei oder mehr Gruppen ohne fixierte Modalität. Sie nennt zudem den Rhythmus von Veranstaltungen, welcher unterschiedliche Implikationen hat. Findet eine Veranstaltung vollständig und regelmäßig statt, so weist diese einen hohen Strukturierungsgrad des geleiteten Lernens auf, während eine reduzierte und punktuelle Planung von Präsenzveranstaltungen selbstgesteuertes Lernen fokussiert.

Ähnlich der Steuerbarkeit wird, auch speziell im Zusammenhang mit technischen Komponenten, die Flexibilität und Modalität eines Lernraums als Dimension genannt, welche in Abschnitt 1.6.2 bereits als wichtige Eigenschaft der Medienökologie erkannt wurde. Dabei steht die Möglichkeit zur spontanen Umkonfiguration des Raumes an oberster Stelle, welche jedem Teilnehmenden ermöglicht werden muss. Ebenso sind dynamische Auswahlen, Erweiterungen und Kombinationen von Räumen, Unterräumen, Interaktionen, Anwendungen und Personen bzw. Communities entscheidend.

Eine letzte, jedoch nicht minder wichtige Dimension hybrider Lernräume ist dessen Menschlichkeit bzw. Sozialität. Ein Bestandteil vieler vorgestellter Modelle ist der so-

ziale Raum für menschliche Kontakte, Interaktionen und Austausch abseits der Lehre. Einbindungen externer Personen und Institutionen ergänzen die campuseigene Community und auch Social Media wird häufig als wichtiger Kommunikationsweg erkannt. Durch solche Kanäle ist eine Verbindung von Wissensschaffung und Geselligkeit möglich. Bei der Planung hybrider Räume oder darin genutzter Technologie ist also auch die Gewichtung von Menschen- und Maschinenkontakt als Dimension zu beachten.

Die genannten Eigenschaften werden mitsamt zugehöriger Ausprägungen in Tabelle 2.1 zusammengefasst.

Tabelle 2.1.: Eigenschaften hybrider Lehre und deren identifizierte Ausprägungen

Eigenschaft	Ausprägungen
Modalität / Übertragungsweg	Physisch (Face-to-Face Unterricht, Physische Ressourcen - Bücher, Zettel, Whiteboards, etc.))
	Digital (Vorlesung als Lifestream, Digitale Ressourcen - Videos, Websites, PDFs, etc.)
Synchronität des Modalitätseinsatzes	Synchrone / Gleichzeitige Nutzung der Modalitäten
	Asynchrone / Zeitversetzte Nutzung der Modalitäten
Parallelität der Teilnahme	Synchrone / Gleichzeitige Anwesenheit Teilnehmern oder Gruppenmitgliedern
	Asynchrone / Zeitversetzte Anwesenheit von Teilnehmern oder Gruppenmitgliedern
Authentizität / Natürlichkeit / Nutzerzentriertheit	Unterschiedliche Repräsentationen von Wissen (von Ressourcen angesprochene Sinne)
	Authentizität bzw. Praxisnähe der Aufgabenstellungen und Arbeitsschritte
	Integration aktueller, persönlicher und vielgenutzter Software sowie Informationsquellen
	Adaptivität bzw. Anpassung von Lernprozessen, Lernräumen und Lerninhalten auf Bedürfnisse der Lernenden und Lehrenden (abgestimmt vs. serienmäßig)
Interdisziplinarität / Offenheit	Fächerübergreifende Lehrinhalte und Lehrmethoden
	Zusammenarbeit unterschiedlicher Forschungsrichtungen
	Offenheit gegenüber neuen Methoden, Technologien und Inhalten

Eigenschaft	Ausprägungen
Flexibilität / Modularität	Flexible ad-hoc Umkonfiguration des Lernraums
	Flexible Wahl und Kombination von Lernmethoden, Arbeitsschritten, Anwendungen sowie Funktionen
	Flexibler Wechsel zwischen Lernräumen, Gruppen und Communities
	Flexible Erstellung und Verwaltung von Gruppen, Gruppenmitgliedern und integrierten Communities
	Flexible Erweiterbarkeit des System, um auf wechselnde didaktische oder technische Bedingungen einzugehen
	Flexible Erstellung und Verwaltung von Räumen und Unterräumen (auch Duplizierung bestehender Räume)
Menschlichkeit / Sozialität / Kollektivität	Menge des Menschenkontaktes (vs. Maschinenkontakt, Verflechtung von Wissensschaffung und Geselligkeit)
	Individuelle Lernprozesse (Selbststudium) vs. kollaborative Lernprozesse (soziale Räume und Aktivitäten)
	Transparenz / Sichtbarkeit von Prozessen, Räumen, Ressourcen und Artefakten (Personal Space vs. Open Space)
	Interaktionen mit und Einbindungen von anderen Personen, Instanzen und Communities (Campus interne Personen, externe Institute, Museen, Social Media, Foren, etc.)
Erreichbarkeit / Seamless Connectivity	Kontextgerechte und zeitgemäße Zugangsmethoden zu technischen Systemen (z.B. single-sign-on)
	Verknüpfung und Kompatibilität von Lernräumen, Ressourcen und Artefakten (Zusammenführung und Kombination von Wissen)
	Erreichbarkeit und Persistenz von Lernressourcen und Artefakten (zeitbasierte vs. stetige Erreichbarkeit)
	Kompatibilität und Verknüpfung von bereitstehenden Geräten (geräteunabhängige Nutzung, Übertragbarkeit und Fortführung des Lernprozesses)
	Orts- und Zeitunabhängigkeit des Zugriffs auf den Lernraum / den Lernprozess und alle beteiligten Aktivitäten / Funktionen
	Übertragung von Objekten, Ressourcen und Artefakten aus dem Physischen in eine digitale Form

Eigenschaft	Ausprägungen
Strukturierungsgrad	Struktur des Raumes / Settings (z.B. akademisch vs. außer-akademisch)
	Struktur der Lehre (strukturiert vs. unstrukturiert / offen)
	Formalität von Unterrichtsmethoden
	Rythmus einer Veranstaltung und beteiligter Arbeitsschritte (vollständig und regelmäßig, reduziert und punktuell, etc.)
	Steuerbarkeit des Lernprozesses (selbstgesteuert vs. gruppenorientiert vs. geleitet, auch Gruppeneinteilungen, Methodenwahl, Zeiteinteilung / -planung, etc.)
	Einsatz digitaler Lernmaterialien (optional als Ergänzung der Präsenz, gelegentlicher Einsatz als Ersatz der Präsenz, vollständiger Ersatz der Präsenz)
	Aufteilung der Teilnehmer in Gruppen (Plenum ohne Aufteilung, Aufteilung in eine Vor-Ort und eine Online-Gruppe, Aufteilung in zwei oder mehr Gruppen ohne fixierte Modalität, etc.)

## 3. Framework zur Konzeption hybrider Systeme

Nachdem nun notwendige Bestandteile und Eigenschaften hybrider Lehre und Lernräume identifiziert wurden, sind geeignete Schritte der Planung und des Designs eines hybriden Systems zu thematisieren. Durch vorangegangene Abschnitte wurde deutlich, dass die Planung auf Basis charakterisierender Dimensionen ein umfassendes Vorgehen darstellt. Hybridität bedeutet die Ermöglichung von Mischformen, welche ad-hoc auf die Bedürfnisse der Nutzer und andere kontextbedingte Anforderungen abzustimmen sind. Bedenkt man nun bei der Konzeption eines Systems alle möglichen Kombinationen von Dimensionen und zugehörigen Ausprägungen, so ergeben sich gefühlt unendlich viele verschiedene Raumtypen. Diese Raumtypen separiert zu betrachten und in das System zu integrieren würde demnach eine unverhältnismäßig lange Zeit in Anspruch nehmen. Weiterhin sind bei der Planung didaktische Methoden und Lernformen zu berücksichtigen, welche auch ein interdisziplinäres Lernen ermöglichen. Hierdurch entstehen zusätzliche Differenzierungen zwischen notwendigen Räumen, weshalb der Planungsaufwand weiterhin gesteigert wird. Demnach herrscht der Bedarf nach einer flexibleren und praxisorientierteren Methode der Systemplanung, deren Erforschung sich dieses Kapitel widmet.

### 3.1. Aktivitäten als Zentrum der Raumplanung

Goodyear and Ellis [2008] beschäftigen sich mit solch einem komplexen Design von Lernräumen, wobei sie vor allem das Verhalten von Lernenden im Lernprozess betrachten. Ihre Ergebnisse schlagen bei der Planung von Lernräumen einen Ablauf vor, dessen zentraler Aspekt die Identifikation notwendiger und kontextgerechter Lernaktivitäten darstellt:

„Again, observing what students actually do—how they move in, inhabit and reconfigure space, how they create congenial learning places, how they assemble tools and other artefacts in their work as students—is the best way of gaining insights [...]“[Ellis and Goodyear, 2016, S. 181]

Um diesen greifbar zu machen, verweisen sie auf ein *activity-centred model*, welches in Abbildung 3.1 visualisiert ist.

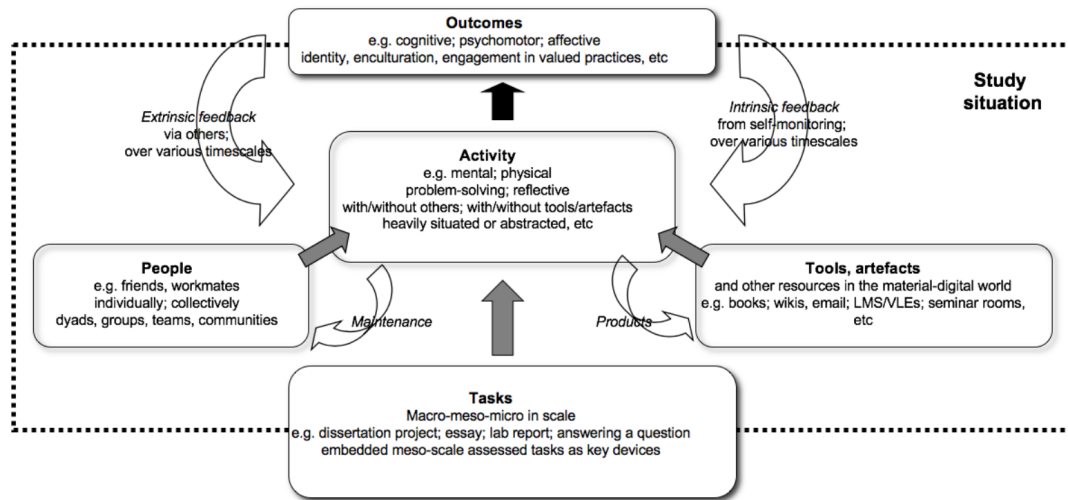


Abbildung 3.1.: Activity-centred model, entnommen aus Goodyear and Ellis [2008, S. 24]

Zum Anstoß eines Lernprozesses wird darin stets ein Task bzw. eine Aufgabenstellung mit variierendem Umfang benötigt. Dies kann die Erstellung einer Dissertation sein, jedoch auch die Beantwortung einer einfachen Frage. Vergleicht man dieses Element mit den zuvor identifizierten Entitäten hybrider Lernräume, so ist eine Verbindung zu Lernmethoden und Lernformen ersichtlich, welche von der pädagogischen Komponente des hybriden Paradigmas abhängen.

Zur Bewältigung einer Aufgabenstellung wählen Lernende für sie optimale Lernaktivitäten. Diese können sowohl mental als auch physisch sein, problemlösend (aktiv) oder reflektiv, sowie situationsabhängig oder generalisiert. Die Auswahl der Aktivitäten als Zentrum ist dabei bewusst gewählt. Zwar hängen diese von der jeweiligen Aufgabenstellung ab, jedoch entscheiden Lernende selbst, welche Aktivitäten für sie in dem jeweiligen Kontext angemessen sind. Zudem kann dieselbe Aktivität zur Erfüllung mehrerer Aufgabenstellungen dienen.

Die Wahl einer Aktivität ist jedoch auch davon abhängig, welche Personen, Tools oder Artefakte (Ressourcen) im Lernprozess vorhanden sind. Zu den Teilnehmenden an einer Aktivität zählen die Autoren unter anderem Freunde der Lernenden, deren Arbeitskollegen, Team- und Gruppenmitglieder, sowie interne und externe Communities.

Dabei können die Teilnehmenden individuell vorgehen, beispielsweise indem sie verschiedenen Abschnitten zugeordnet werden, oder kollektiv eine Thematik erarbeiten. Als Tools und Artefakte verstehen die Autoren sämtliche physischen und digitalen Ressourcen, welche innerhalb der Aktivität als Lernmaterial bereitstehen. Hierzu zählen beispielsweise Bücher, Emails, Webseiten oder Learning Management Systems. Auch die Ontologie der genutzten Räumlichkeiten und deren Möglichkeiten nennen sie in diesem Zusammenhang.

Die Aktivität gilt demnach als Vermittler zwischen der Aufgabenstellung, sowie den physischen / digitalen und sozialen Bestandteilen. Das Ziel dieser Interaktionen sind abschließend Learning Outcomes, zu denen die in vorherigen Abschnitten als Artefakt bezeichneten, unterschiedlichen physischen und digitalen Formen der Wissenspräsentation gehören. Ebenso zählen persönliche Erfahrungen, Identitätsbildungen und andere kognitive oder (psycho-) motorische Entwicklungsprozesse sowie Kompetenzen als Learning Outcomes.

Doch stellen die Verbindungen zwischen den einzelnen Bestandteilen keinen unidirektionalen Einfluss dar. Angestrebte Learning Outcomes können auch Aktivitäten beeinflussen und entstandene Produkte / Artefakte als neue Ressource ein intrinsisches oder extrinsisches Feedback ermöglichen, was in sich wiederum eine neue Aktivität darstellt. Ebenso können abhängig von der Aktivität gezielt physische / digitale Umgebungen ausgewählt werden, und entstehende Produkte / Artefakte wiederum diese Umgebung anreichern. Auch die Wahl der Teilnehmenden kann aktivitätsgesteuert erfolgen oder durch frühere Aktivitäten beeinflusst werden. So sind Aktivitäten keine stringenten Prozesse, sondern schleifenartige Gebilde, welche ihre Umgebung formen und ebenso von ihr geformt und modifiziert werden [vgl. Ellis and Goodyear, 2016, S. 179 f.].

Als Erweiterung dieses Modells veröffentlichten Goodyear et al. [2021] Orientierungshilfen für ein strukturiertes Design. Dafür unterteilten sie zunächst den Designprozess in drei Unterabschnitte [vgl. Goodyear et al., 2021, S. 447 ff.]:

1. **Epistemisches-Design** der Aufgabenstellungen
2. **Set-Design** der physischen / digitalen Ressourcen, Tools und Räume
3. **Social-Design** der Teilnehmenden, Gruppen und Rollen

Bei der Planung sind ihrer Aussage zufolge nun vier wichtige Fragen zu beantworten. Diese präsentieren sie in Form der verschiedenfarbigen *ACAD-Cards* (Activity-Centred

Analysis and Design), welche in Tabelle 3.1 kurz erläutert sind.

Tabelle 3.1.: ACAD-Cards (Activity-Centred Analysis and Design), inhaltlich entnommen aus Goodyear et al. [2021, S. 456]

<b>Karten</b>	<b>Frage</b>
<b>Blue cards</b> (philosophy / pedagogy)	<b>What learning theory underpins your design?</b> e.g. constructivism -- learning as a process of actively building knowledge.
<b>Yellow cards</b> (epistemic design)	<b>How will you structure and pace tasks?</b> e.g. lecture, case studies, model building etc. and forms of assessment.
<b>Green cards</b> (set design)	<b>What resources will you use?</b> e.g. smartphone, pen & paper, LMS, outside, collaborative learning, studio, seminar room.
<b>Orange cards</b> (social design)	<b>How will you group learners?</b> e.g. groups, pairs, scripted roles, note-taker, mentor.

Bezüglich der blauen Karten grenzen die Autoren die Planung auf eine Macro-, eine Meso- und eine Micro-Ebene ein. Auf der globalen Macro-Ebene ist zu beschreiben, welche globalen Designentscheidungen zu wählen sind. Dazu gehört bezüglich des Set-Designs die Wahl der digitalen und physischen Infrastruktur, also des Orts, und der Technologie. Epistemisch ist auf dieser Ebene die gewünschte Form der Wissensproduktion und -verteilung zu klären. Letztlich befasst sich das soziale Design hier vor allem mit Organisationsstrukturen.

Auf der Meso-Ebene werden lokale Faktoren eines Lernprozesses betrachtet. Dazu gehört die gezielte Zuteilung und Nutzung von Räumen und Technologie, wobei speziell deren Verfügbarkeit und Zugangsmöglichkeiten zu planen sind. Zudem müssen Anforderungen des Curriculums beachtet werden und eine genaue Eingrenzung beteiligter Communities stattfinden.

Die detaillierteste Planung findet auf der Micro-Ebene statt, auf welcher sowohl die Artefakte, Tools und Ressourcen zu wählen sind, als auch die Auswahl, Abfolge und das Tempo von Aufgabenstellungen oder Aktivitäten. Letztlich ist hier auch eine Rollenverteilung und Arbeitsaufteilung innerhalb einer Gruppe zu bestimmen. Die genannten Zusammenhänge sind in Tabelle 3.2 zusammengefasst.



Tabelle 3.2.: ACAD-Wireframe (Activity-Centred Analysis and Design), inhaltlich entnommen aus Goodyear et al. [2021, S. 457]

Philosophy or high-level pedagogy	<b>SET DESIGN</b> Learning is...	<b>EPISTEMIC DESIGN</b> Learning is...	<b>SOCIAL DESIGN</b> Learning is...
<b>MACRO</b> The global Level 1 patterns	<b>Building &amp; technology</b> e.g. digital & physical infrastructure	<b>Stakeholder values</b> e.g. forms of knowledge production & sharing	<b>Organisational structures</b> e.g. hierarchal or networked
<b>MESO</b> The local Level 2 patterns	<b>Allocation or use of space &amp; technology</b> e.g. availability & access	<b>Curriculum</b> e.g. unit of study, program or degree	<b>Community</b> e.g. school, faculty, cohort or club
<b>MICRO</b> The detail Level 3 patterns	<b>Artifacts, tools &amp; resources</b> e.g. clock, BYOD & furnishings	<b>Selection, sequence &amp; pace</b> e.g. content & timing of tasks	<b>Roles &amp; divisions of labour</b> e.g. facilitator, groups & scripts

Für die Planung eines Systems zur Unterstützung möglichst vieler Aufgabentypen bedeutet solch eine aktivitätsbasierte Herangehensweise nun, dass zunächst die wichtigsten epistemischen Lernformen identifiziert werden müssen, da diese eine strukturierte Eingrenzung zu unterstützender Methoden und darin benötigter Aktivitäten ermöglichen. Ebenso stellen Set- und Social-Design auf der Makro-Ebene, sowie auf Teilen der Meso-Ebene allgemeine Anforderungen an ein System. Erst nachdem diese Aspekte im Systemdesign bzw. grundlegenden Framework bedacht wurden, kann eine umfassende Spezifikation und Einbettung benötigter Aktivitäten mit deren Personen, Artefakten, Ressourcen und Tools stattfinden. Aufgrund dieser Einbettungen kann bei der späteren Erstellung von Lernräumen eine kontextspezifische Auswahl benötigter Komponenten erfolgen, welche den zuvor individuellen Ursprung überbrückt, Mischformen ermöglicht und so den hybriden Gedanken realisiert.

### 3.2. Detaillierte Betrachtung der Lehrformen, Methoden und Aktivitäten

Jegliche Planung von Lehr- oder Lernveranstaltungen beginnt bei der Betrachtung des jeweiligen Lehr- / Lernkontextes, welcher bei der Beschreibung eines Lehrszenarios eine erhebliche Rolle einnimmt. Dabei sind viele Faktoren zu beachten, beispielsweise örtliche, zeitliche, physische oder digitale Gegebenheiten, sowie bestehende Probleme in der Lehre und die Anforderungen eines Curriculums (siehe Tabelle 3.2). Abgestimmt

darauf findet anschließend die Auswahl geeigneter Lehr- / Lernformen bzw. Arbeitsformen oder Unterrichtsformen statt, welche ebenfalls im Szenario festgehalten werden.

Zur Ermittlung der Lehr- / Lernformen bzw. Unterrichtsformen ist zunächst zu untersuchen, welche Kategorien von Lehrveranstaltungen existieren. Diese Kategorien sind laut Meyer [2009] von der jeweiligen Sozialform abhängig, welche er wie folgt definiert:

„Sozialformen geben an, mit wem ein Schüler in einer bestimmten Unterrichtsphase zusammenarbeitet. Sie regeln also die Beziehungs- und Kommunikationsstruktur des Unterrichts.“[Meyer, 2009, S. 136]

Dabei sind vier mögliche Sozialformen zu unterscheiden [Meyer, 2009, S. 136 ff.]:

1. Klassenunterricht / Frontalunterricht / Plenum
2. Einzelarbeit / Stillarbeit
3. Partnerarbeit
4. Gruppenarbeit

Eine grobe Kategorisierung könnte nun anhand dieser Sozialformen geschehen. Da sich die Partnerarbeit und Gruppenarbeit vor allem anhand der Anzahl von Personen unterscheidet, können diese zur *kollaborativen Arbeit* oder *Teamarbeit* zusammengefasst werden, wie es beispielsweise von Euler et al. [2006] oder Rösler [2012] gemacht wird. Diese Sozialformen klären nun die Frage, mit wem zusammen gelernt wird. Es fehlt also nun noch die Entscheidung, wie zu lernen / lehren ist. Hierbei finden sich unterschiedliche Auflistungen möglicher Arbeitsformen, welche nicht immer klar von den Sozialformen trennbar sind bzw. von diesen getrennt werden. So wird als zusätzliche Kategorie häufig das *informelle Lernen* bzw. die *praktische* oder *offene Arbeit* genannt (siehe Peschel [2003] oder Düsseldorf [2021]), welche im Gegensatz zu den anderen Typen keine Strukturierung und Eingrenzung durch Lehrende erfährt und Lernenden eine vollständige Selbstbestimmtheit des Lernprozesses ermöglicht. Rösler [vgl. 2012, S. 104 ff.] teilt Arbeitsformen in Übungen und Aufgaben, Projekte, sowie Spiele ein. Dabei nennt er jedoch, dass diese Arbeitsformen stark von der Domäne abhängig sind. Aebli [2006] führten eine stärkere Differenzierung anhand von zwölf Grundformen ein. Diese zwölf Grundformen lauten:

- Erzählen und Referieren
- Vorzeigen – Vormachen
- Anschauen und Beobachten

- mit Texten umgehen
- Texte schreiben
- Handlungsablauf erarbeiten
- Begriffe bilden
- Problemlösungen aufbauen
- Durcharbeiten
- Üben und Wiederholen
- Anwenden
- selbstständig entdeckendes Lernen

Weiterhin wird auf [e-teaching.org](https://www.e-teaching.org) [2021a] beschrieben, dass gerade durch einen Medieneinsatz die traditionellen Kategorien aufgebrochen werden können, um hybride Mischformen zu ermöglichen. All diese Aspekte sind bei der Planung einer Unterrichtsform zu bedenken, für die anschließend geeignete Lernmethoden gewählt werden müssen, welche wiederum mögliche Aktivitäten zur Aufgabenbewältigung eingrenzen.

Bei der Auswahl von Lernmethoden ist nun zunächst zu bedenken, welche Unterrichtsformen, also Sozial- und Arbeitsformen sinnvoll sind und wie sie kombiniert werden können. So weisen auch Lernmethoden viele Überschneidungen zu den zuvor genannten Kategorisierungen auf. Bei der Planung ist weiterhin der Strukturierungsgrad und Aktivierungsgrad einer Methode entscheidend. Hierbei nennt Einsiedler [1981] eine grundlegende Einteilung in die Aktionsformen darbietend, erarbeitend und explorativ. Laut [e-teaching.org](https://www.e-teaching.org) [2017] sind darbietende Methoden stark von den Lehrenden strukturiert und angeleitet, wodurch sie sich für die Einführung eines Themas oder dessen Zusammenfassung eignen. Hierzu gehören beispielsweise Vorträge, Demonstrationen oder das Vormachen. Erarbeitende Methodiken sind nur halbstrukturiert und Lehrende bilden nicht mehr den Mittelpunkt. So ist auch der Unterricht nur teilweise geplant und beinhaltet Phasen für eine selbstgesteuerte und offene Gestaltung durch Lernende. Als erarbeitende Methoden gelten unter anderem angeleitete Übungen oder der Dialog zwischen Lehrenden und Lernenden. Explorative Methoden zeichnen sich nun durch einen geringen Strukturierungsgrad aus und fordern eine hohe Eigenaktivität der Lernenden. Hierzu zählen das Brainstorming, Fallstudien oder Planspiele. Wie in Abschnitt 3.1 und Tabelle 3.2 aufgezeigt, sind Lernmethoden im epistemischen

Design auf der Macro-Ebene abhängig von den gewünschten Formen der Wissensproduktion und auf der Meso-Ebene abhängig von dem Curriculum. Demnach finden sich auch fach- und domänenspezifische Methoden und Methodenkombinationen, wie beispielsweise das *Live-Coding* in Studiengängen der Informatik.<sup>1</sup> Durch die erweiterte Charakterisierung anhand des Curriculums ist eine vollständige Sichtung aller Methodiken innerhalb einer Ausarbeitung jedoch unwahrscheinlich. Umso wichtiger ist es, dass das geplante Modell und darauf aufbauende System eine hohe Flexibilität und Erweiterbarkeit bezüglich der Methodenrealisierung aufweisen.

Eine Möglichkeit der Erweiterbarkeit findet sich in der Abstraktion. Auch wenn sich eine Methode aus mehreren Aktivitäten zusammen setzt, finden dennoch dieselben Aktivitäten innerhalb mehrerer Methoden Anwendung. Dabei sind abstrahierte Einsatzbereiche von Aktivitäten erkennbar, welche wiederum für eine flexiblere Planung und Erweiterbarkeit berücksichtigt werden sollten. Ebenso ist eine Definition unterschiedlicher abstrakter Aktivitätstypen hilfreich bei der Erforschung und Identifikation konkret benötigter Funktionen. Conole [vgl. 2007, S. 84 ff.] beschreibt diesbezüglich eine Taxonomie mit sechs Aktivitätstypen, welche als Vorlage für Tabelle 3.3 dient.

Tabelle 3.3.: Taxonomie mit sechs Aktivitätstypen nach Conole [vgl. 2007, S. 84 ff.]

Typ	Beschreibung
Assimilativ (assimilative)	Passive Aktivitäten der Wissensaufnahme wie Lesen, Zusehen oder Zuhören
Informations- verarbeitend (information handling)	Aktivitäten in denen Ressourcen aktiv gesammelt und zur Schaffung neuen Wissens verarbeitet werden
Adaptiv (adaptive)	Aktivitäten, welche mithilfe von Modellierungs- und Simulationssoftware der Überführung oder praktischen Erprobung von Wissen dienen
Kommunikativ (communicative)	Sämtliche kommunikationsbezogenen Aktivitäten
Produktiv (productive)	Aktivitäten in denen die aktive Schaffung von Artefakten angestrebt wird
Erlebnisorientiert (experiential)	Aktivitäten in denen Fähigkeiten im Feld erprobt oder ein Feld investigativ erforscht wird

<sup>1</sup>Umfassende Auflistungen etablierter Methodiken bieten beispielsweise Reich [2017], NRW [2018], methopedia.eu [2017], Düsseldorf [2021] und e-teaching.org [2017]

Nun können aus Sicht eines Systems Anpassungen der Aktivitätstypen sinnvoll sein. Methoden bilden sich aus Aktivitäten und sind deshalb selbst als komplexe Aktivität darstellbar. Auch die genannten Aktionsformen der Methoden weisen Überschneidungen mit den Aktivitätstypen auf, weshalb ihre Vereinheitlichung sinnvoll erscheint. *Erlebnisorientierte Aktivitäten* und *explorative Methoden* sind sich nicht nur begrifflich, sondern auch inhaltlich ähnlich, ebenso wie *informationsverarbeitende Aktivitäten* und *erarbeitende Methoden*. Im folgenden Dokument werden deshalb stets die Begriffe *explorativ* und *erarbeitend* genutzt. Einzig für *darbietende Methoden* findet sich kein direktes Äquivalent. Es könnte als Mischform von *assimilativen Aktivitäten*, welche die passive Komponente der darbietenden Methode beleuchten, und *kommunikativen Aktivitäten* gesehen werden, da auch Präsentationen und vergleichbare Tätigkeiten als Form der Kommunikation gelten. Um Vermischungen und Einordnungsproblemen vorzuzugreifen, werden darbietende Aktivitäten als zusätzliche Kategorie eingeführt, sodass sich kommunikative Aktivitäten ausschließlich auf die Kontaktaufnahme, Dialoge und andere Handlungen der sozialen Strukturierung beschränken.

Weiterhin erscheint die Einteilung unter Anbetracht der zuvor definierten Eigenschaften hybrider Lernumgebungen und der revidierten Bildungsziel-Taxonomie von Anderson und Krathwohl [2002], welche die bekannte Taxonomie von Bloom et al. [1984] erweitert, als noch unvollständig. Die Steuerbarkeit des Lernprozesses durch Lernende selbst wurde als wichtige Eigenschaft genannt. Diese bezog die Erstellung und Verwaltung von Gruppen, die Zusammenstellung und Strukturierung des eigenen Lernraums und die selbstorganisierte Abfolge benötigter Arbeitsschritte ein. Auch auf der obersten Ebene kognitiver Prozesse, dem *Kreieren*, wurde von Krathwohl [vgl. 2002, S. 215] eine selbstständige Planung als wichtige Komponente genannt. Demnach sind administrative und organisatorische Aktivitäten ebenso notwendig, wie die bereits genannten Aktivitätstypen. Auch das Assessment galt in vorherigen Abschnitten als zentraler Bestandteil von hybriden Lernräumen. Nicht nur Bewertungen und Feedback von Lehrenden, sondern auch von den Lernenden selbst sind in unterschiedlichen Lehr- / Lernkontexten notwendig. Diese könnten zwar als kommunikative Aktivitäten gesehen werden, doch sind auch reflektive Tätigkeiten wie Evaluationen mit einem Assessment verbunden, welche nicht notwendig auf einer direkten Kommunikation aufbauen müssen und ebenso assimilative, informationsverarbeitend oder produktive Bestandteile aufweisen können. Ähnlich dem *Kreieren* und damit verbundenen Planungsprozessen findet sich nun auch wieder das *Evaluieren* als eigenständige Ebene der überarbeiteten Bildungsziel-Taxonomie von Krathwohl [vgl. 2002, S. 215]. Da es somit im Lernprozess eine gravierende Rolle einnimmt oder sogar die zentrale Aufgabe darstellt, ist es bei der Definition eigenständiger Aktivitätstypen zu beachten. Ebenso verhält es sich

mit dem in Abbildung 2.8 dargestellten Support, welcher als begleitende Aktivität im Lernprozess anzusehen ist.

Somit ergibt sich letztlich eine mögliche Aufteilung von Aktivitäten in elf Kategorien, welche in Tabelle 3.4 aufgelistet sind.

Tabelle 3.4.: Liste identifizierter Aktivitätstypen

Typ	Beschreibung
Darbietend	Aktivitäten, welche der Präsentation / Vermittlung erarbeiteten Wissens dienen
Assimilativ	Passive Aktivitäten der Wissensaufnahme aus bestehenden Ressourcen (z.B. Lesen, Zusehen, Zuhören oder Auswendiglernen)
Erarbeitend	Aktivitäten in denen Ressourcen aktiv aus unterschiedlichen Quellen gesammelt und zur Schaffung neuen Wissens verarbeitet werden (z.B. Recherche auf Webseiten, Integration von Foren / Social Media als Informationsquelle, Sortierung von Quellen, Analysen oder Klassifizierungen)
Adaptiv	Aktivitäten, welche der Überführung oder praktischen Erprobung von Wissen dienen (z.B. Simulation oder Überführung von physischen Objekten ins Digitale)
Kommunikativ / Sozial	Sämtliche sozialen und kommunikationsbezogenen Aktivitäten (z.B. Chats, Video-Konferenz oder aktives Teilen von Inhalten in Social Media)
Produktiv	Aktivitäten in denen die aktive Schaffung von Artefakten angestrebt wird (z.B. Dokumenterstellung oder Videoproduktion)
Explorativ	Experimentelle Aktivitäten in denen Fähigkeiten im Feld erarbeitet oder ein Feld investigativ erforscht wird (z.B. Interviews oder Nutzung von Innovationskarten)
Organisatorisch / Administrativ	Planende Aktivitäten zur Strukturierung des Lernprozesses (z.B. Gruppeneinteilung, Definition von Arbeitsschritten, Aufgabenverteilung oder Definition fester Termine)
Begleitend / Support	Aktivitäten, in denen Lehrende oder Lernende beim Lernprozess unterstützen

Typ	Beschreibung
Reflektiv / Evaluierend	Bewertende Aktivitäten auf Basis zuvor erstellter Artefakte oder bereitgestellter Ressourcen
Prüfend	Spezielle Form der reflektiven Aktivität bei der Wissen abgefragt / abgerufen werden muss und welche oft in einer Benotung endet

Zu beachten ist, dass diese Aktivitätstypen zwar unterschiedliche Schwerpunkte darstellen, eine Aktivität jedoch in den seltensten Fällen nur einer dieser Kategorien gerecht wird. Sie stellen deshalb grobe Richtlinien bei der Planung zielgewandter Funktionen eines Systems dar, welche ihre Identifikation innerhalb des jeweiligen Kontextes vereinfachen und beschleunigen sollen. Dabei sind unterschiedliche Fragen sinnvoll, welche beispielhaft in Tabelle B.1 des Anhangs für jeden Typen aufgelistet sind.

### 3.3. Übersichtsmodell notwendiger Designschritte

In diesem Kapitel wurden Lernaktivitäten als Zentrum eines strukturierten Designprozesses erkannt und unterschiedliche Aktivitätstypen für die Ergründung benötigter Funktionalitäten eines Systems beschrieben. Durch die Abhängigkeit der Aktivitäten von Lernmethoden, dem Lehrkontext, angestrebten Unterrichtsformen und Learning Outcomes, ist für jeden Kontext eine domänenspezifische Analyse notwendig. Aufbauend auf den in Abschnitt 2.4 identifizierten Kernbestandteilen und -eigenschaften hybrider Lehre / Lernräume, sowie den in diesem Kapitel erarbeiteten didaktischen Strukturen wird nun ein Framework vorgestellt, welches als Grundlage strukturierter Analysen bestehender oder zukünftiger hybrider Systeme nutzbar ist. Dieses in Abbildung 3.2 dargestellte Modell bietet eine Übersicht notwendiger Design-Schritte, welche sequentiell in den folgenden Unterabschnitten erläutert werden. Es erhebt jedoch keinen Anspruch auf Vollständigkeit und stellt demnach vor allem die wichtigsten Bestandteile und ihre Zusammenhänge dar. So können bei dessen Anwendung domänenspezifische Erweiterungen präsentierter Schritte notwendig oder vorteilhaft sein. Allgemein sollten in jedem Schritt der Analyse zudem die Übersichtsmodelle der Entitäten und Dimensionen betrachtet werden (siehe Abschnitt 2.4), welche bei einer kontextgerechten Auswahl helfen können.

#### 3.3.1. Lehr- / Lernszenario definieren

Als Ausgangspunkt der Planung hybrider Systeme gelten die didaktischen Bedingungen des Lehr- / Lernszenarios bzw. eines Kurses, welche durch die blaue ACAD-Card

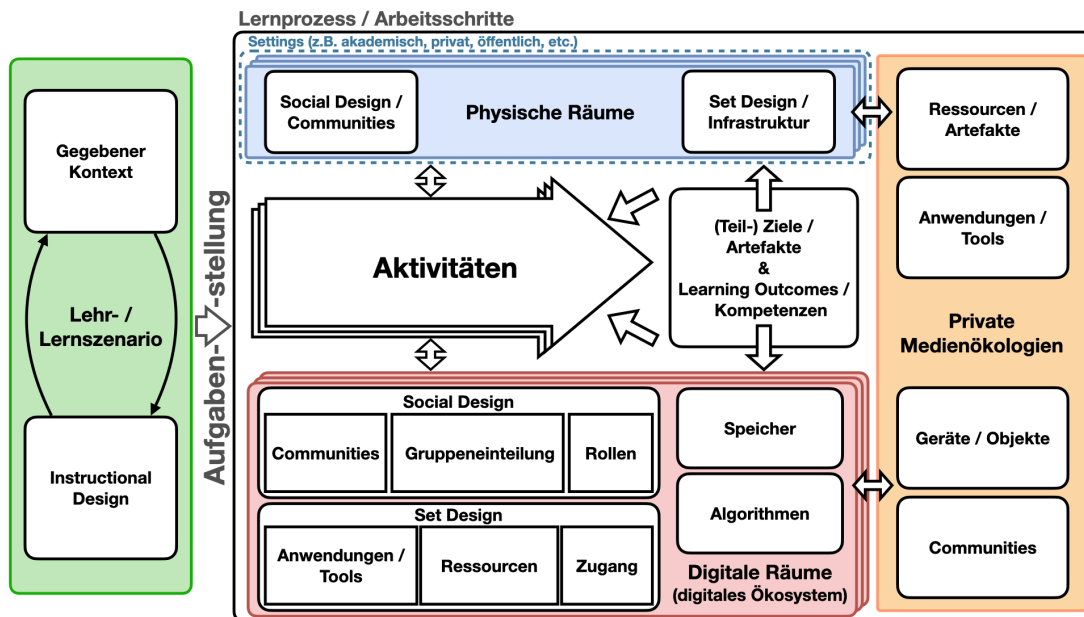


Abbildung 3.2.: Aktivitätszentriertes Framework als Grundlage des Designs hybrider Systeme

widergespiegelt sind (siehe Tabelle 3.1). Dabei sind sowohl die Gegebenheiten des Kontextes zu betrachten, als auch die Aspekte des *Instructional Designs*<sup>2</sup>. Als Kontext gelten zunächst die Anforderungen und Restriktionen des Curriculums, welche bereits eine Eingrenzung möglicher Unterrichtsformen und -methoden darstellen. Ebenso sind vorhandene Räume des Campus und darin beinhaltete Geräte und Garnituren zu betrachten, sowie die Personen, welche allgemein an dem Lehr- / Lernszenario beteiligt sind. Die örtlichen und zeitlichen Gegebenheiten des Zugangs der Personen zur bereitgestellten Infrastruktur grenzen das Instructional Design weiter ein, sodass auch die Nutzungs- und Bereitstellungsmöglichkeiten von Ressourcen und anderen Objekten einzuplanen sind. Abhängig von diesen Kontextbedingungen kann nun ein geeignetes Instructional Design gewählt werden. Dabei sind zunächst die Learning Outcomes zu definieren, welche zu Teilen von dem Curriculum abhängen. Zur Erreichung der Learning Outcomes sind anschließend geeignete Sozial- und Arbeitsformen auszuwählen und darauf abgestimmte Lehr- / Lernmethoden festzulegen (siehe Abschnitt 3.2). Doch ist die Beziehung zwischen dem Kontext und dem Instructional Design nicht nur unidirektional. Sind besondere Lernmethoden vorgesehen, können auch die Kontextbedingungen angepasst werden, beispielsweise durch Anschaffungen neuer Geräte, Garnituren oder Ressourcen. Abhängig vom gesamten Lehr- / Lernszenario werden für Lernende anschließend geeignete Aufgaben definiert. Die Planung eines Systems muss

<sup>2</sup>Instructional Design: Siehe Smith and Ragan [2004, S. 3 ff.]



jedoch nicht zwingend beim Kontext beginnen, da in den meisten Fällen für bereits vorhandene Methoden und Aufgaben neue Umsetzungsmöglichkeiten gesucht werden. In diesem Falle ist das Lehr- / Lernszenario bereits definiert und es wird direkt mit der Planung von Lernprozessen und Arbeitsschritten begonnen.

### **3.3.2. Planung des Lernprozesses, benötigter Arbeitsschritte und beteiligter Aktivitäten**

Im Kern der Systemplanung steht die Bereitstellung geeigneter physischer und digitaler Räume. Zur Identifikation der Bestandteile dieser Räume muss zunächst der Lernprozess spezifiziert werden. Die Planung des Lernprozesses beginnt mit der Definition einzelner Arbeitsschritte, welche zur Erreichung des Learning Outcomes / einer Kompetenz und zur Bewältigung bestimmter Aufgabenstellungen notwendig sind. Es wird also entsprechend der gelben ACAD-Card (siehe Tabelle 3.1) die Struktur und Reihenfolge der Aufgabenbewältigung untersucht. Stehen diese fest, ist die Ermittlung der Aktivitäten der wichtigste Schritt des Systemdesigns. Jeder Arbeitsschritt besteht aus unterschiedlichen Aktivitäten, welche mithilfe der Aktivitätstypen (siehe Tabelle 3.4) und den zugehörigen Fragen (siehe Tabelle B.1) ermittelt werden können. Da innerhalb eines Arbeitsschrittes mehrere, aufeinander aufbauende Teilziele oder Kompetenzen notwendig sein können, werden die Aktivitäten anhand ihrer gemeinsamen Zielstellung gruppiert und bilden somit unterschiedliche Teilräume, welche sequentiell oder parallel nutzbar sind. So können innerhalb einzelner Arbeitsschritte zunächst individuelle Recherche-Räume notwendig sein, welche innerhalb eines kollaborativen Brainstorming-Raums eingebettet sind. Neben der Gruppierung von Aktivitäten zu Unterräumen dienen die Teilziele auch der Identifikation notwendiger Artefakte. Es ist zu untersuchen, welche Artefakttypen zur Persistierung des Lernfortschritts oder der Ergebnisse geeignet sind, da dies auch Auswirkungen auf Social- und Set-Design hat, welches anschließend pro Aktivität in geeigneter Form gewählt werden muss (siehe grüne und orangene ACAD-Cards aus Tabelle 3.1). Dabei ist abhängig vom Kontext zu entscheiden, ob zunächst der digitale oder der physische Raum betrachtet werden soll. Sind Anpassungen und Erweiterungen im physischen Raum nur begrenzt oder gar nicht möglich, so bietet sich dieser als Ausgangspunkt an, da er die größten Restriktionen präsentiert. Ähnlich ist der digitale Raum als Erstes zu fokussieren, wenn dieser die meisten Hindernisse aufweist. Auch das ursprüngliche Ziel bzw. der Grund für ein neues Design ist entscheidend. So kann es gewollt sein, einen der Räume gezielt zu verbessern, beispielsweise wenn in diesem starke Defizite wahrgenommen wurden. Diese werden dann natürlich fokussiert, wobei das jeweilige physische oder digitale Äquivalent bei der Planung nicht vernachlässigt werden darf. Finden sich allgemein

keine starken Einschränkungen oder Fokussierungen, so lohnt es sich stets beim digitalen Raum zu beginnen, da dieser die größte Entitätsvielfalt und somit die meisten Anpassungsmöglichkeiten bietet. Viele Bestandteile im physischen Raum können die Nutzung unterschiedlicher, digitaler Elemente ermöglichen, jedoch erfordern einige dieser digitalen Elemente spezifische, physische Geräte und Utensilien. Werden also zunächst digitale Bestandteile gewählt, kann hierdurch eine schnellere Eingrenzung des Gesamtkonzepts erreicht werden. Somit wird auch in den folgenden Abschnitten zunächst auf die Wahl digitaler Elemente eingegangen.

### **3.3.3. Erkundung digitaler Räume**

Bezüglich des digitalen Social-Designs ist zunächst zu klären, welche digitalen Communities an der jeweiligen Aktivität beteiligt sind. Dabei können einzelne Personen des Campus als registrierte und feste, oder anonyme und kurzfristig hinzugefügte Teilnehmende gelten. Ebenso sind unterschiedliche Gruppen einzuplanen, sei es ein zufällig oder gezielt bestimmtes Team aus Kursteilnehmern oder andere Gruppen, welche über den Kurskontext hinaus definiert sind, beispielsweise Klassen, Studiengänge oder Lerngruppen. So ist besonders die Gruppeneinteilung zu beachten. Es muss also zunächst geklärt sein, ob innerhalb der Aktivität Aufgaben an Einzelpersonen oder Teams vergeben werden bzw. ob allgemein Einzelarbeiten und kollaborative Tätigkeiten als Bestandteil gelten. Im Falle von Teameinteilungen müssen benötigte Interaktionen eingeplant werden, darunter auch, welche Teilnehmenden zu welcher Zeit und wie an der Aktivität partizipieren. Im Falle einer Vorlesung können beispielsweise einzelne Teams Vor-Ort und andere remote in digitaler Präsenz teilnehmen, wofür unterschiedliche Übertragungswege, Teilnahmemöglichkeiten und allgemeine Kontextbedingungen herrschen. Letztlich ist die Aufgabenverteilung innerhalb einer Aktivität zu betrachten, also welche Rollen Teilnehmende bei der Aufgabenbewältigung einnehmen und welche Rechte sie dadurch erhalten. So könnten innerhalb eines Brainstormings eine Person als Moderator, eine Person für die Dokumentation und die anderen Personen als Diskutierende eingeteilt sein. Der Moderator bräuchte nun weitere Rechte zur Leitung des Gesprächs, beispielsweise durch eine gezielte Stummschaltung einzelner Personen oder die Nutzung einer Stoppuhr. Diskutierende Personen bräuchten die Rechte, den Sprachchat zu nutzen und parallel gemeinsame Notizen oder andere Quellen einzusehen, welche jedoch nur von der dokumentierenden Person bearbeitet werden dürfen.

Durch das Social-Design können also auch weitere notwendige Teilaktivitäten und dadurch Funktionen des Systems ermittelt werden, welche mithilfe des Set-Designs zu

ermöglichen sind. Dabei sind zunächst Anwendungen und Tools zu wählen, welche für die Aktivität und Teilaktivitäten notwendig sind. Diese können einem gesamten Raum oder einzelnen Unterräumen zugeordnet werden. So werden unterschiedliche Anwendungen für die Darstellung und Erstellung digitaler Artefakte, kollaborative Funktionen wie Sprachchats und viele andere Tools benötigt, beispielsweise für die Durchführung digitaler Umfragen. Neben den Anwendungen sind auch digitale Ressourcen zu beachten, welche vor oder während einer Aktivität bereitgestellt werden sollten. Wichtig ist zudem die Planung des Zugangs zu dem digitalen Raum und seinen beinhalteten Unterräumen, Anwendungen und Ressourcen. Dabei ist zu entscheiden, wer diesen sehen und nutzen kann. Auch ist zu bestimmen, ob der digitale Raum nur an einem bestimmten physischen Ort oder über eine bestimmte Plattform erreichbar ist, beispielsweise nur bei einem Museumsbesuch oder innerhalb eines Hochschulnetzes. So müssen auch unterschiedliche Zugangsmethoden und Zugangsdaten für Accounts in die Planung einbezogen werden. Ebenso sind zeitliche Bedingungen zu klären, also wie und zu welcher Tageszeit ein Zugang oder eine Nutzung möglich ist, sowie bis zu welchem Datum dies der Fall ist.

In Abhängigkeit von dem Set-Design und den Artefakten sind zudem weitere Systembestandteile zu berücksichtigen. So müssen beispielsweise geeignete Speicherorte für Ressourcen und Artefakte eingeplant werden. Auch ist zu entscheiden, welche Hintergrundprozesse allgemein vom System automatisiert wurden, in den Raum einbezogen werden können oder allgemein für diesen Raum neu geschaffen werden müssen. Dabei ist die Sammlung von Nutzungsdaten für die Erstellung kontextgerechter Wissens- und Lernermodelle denkbar, aufgrund welcher dann weitere Arbeitsschritte optimiert oder auf Lernende angepasst werden können. Diese zusätzlichen Systembestandteile sind dann kontextspezifisch zu identifizieren.

#### **3.3.4. Betrachtung physischer Räume**

Physische und digitale Räume können als sich gegenseitig ergänzendes Pendant gesehen werden, da die jeweilige Infrastruktur, die zugehörigen Communities und andere beinhaltete Strukturen erst als Symbiose den hybriden Lernraum aufspannen. Bei der Planung eines digitalen Raums sind demnach physische Gegebenheiten zu beachten, welche dessen Nutzung erst ermöglichen oder anreichern. Vorhandene Garnituren entscheiden zu großen Teilen darüber, ob und wie eine spezielle Aktivität stattfinden kann. Findet sich in einem physischen Raum genügend Platz für eine große Sitzcke, so können Gespräche vor Ort stattfinden und müssen nicht unbedingt über Sprachchats vollzogen werden. Ebenso ist der Zugang zum System und dessen Nutzung abhängig

von den vorhandenen Geräten, sodass beispielsweise ein großes Smartboard kollaborative Aktivitäten vereinfacht, bei denen alle Teilnehmenden an einem Ort versammelt sind. Physische Unterräume ermöglichen es zudem, Teilgruppen zu bilden, welche parallel dieselbe Aufgabe erledigen können, ohne sich gegenseitig zu beeinflussen.

Dabei ist auch wichtig, in welchen Settings einzelne Schritte stattfinden können, da diese unterschiedliche Nutzungen der Infrastruktur ermöglichen bzw. allgemein unterschiedliche Infrastruktur anbieten. So werden in einem privaten Umfeld seltener große Bildschirme vorhanden sein als in einem Campus-Setting, während gemütliche Sitzecken für eine Gruppendiskussion auch in privaten oder öffentlichen Settings vorhanden sein können. Bezüglich der Möglichkeiten eines physischen Raums ist demnach zu entscheiden, welche Arbeitsschritte innerhalb des Schulsettings und welche darüber hinaus stattfinden sollten. Außerschulische Settings sind dabei besonders schwer zu identifizieren, da diese für jeden Lernenden variieren können. Hierbei gilt es allgemeine Gesellschaftsmuster und Lernermodelle zu berücksichtigen, oder mit Lernenden selbst in Kontakt zu treten.

Weiterhin werden Ressourcen und Communities des physischen Raums von dem Setting beeinflusst. Ressourcen sollten identifiziert werden, da vorhandene physische Ressourcen nicht zwingend ein digitales Ebenbild erhalten müssen. Die Communities des physischen Raums können im Vergleich zum digitalen Raum ebenfalls variieren. So sind unter Umständen über physische und digitale Kommunikationskanäle andere Personen und Gruppierungen erreichbar. Bei Umfragen sind beispielsweise alle Teammitglieder im digitalen Raum vertreten, während auf der Straße befragte Passanten nur über den physischen Weg teilnehmen können. Diese Aspekte sind bei der Planung zu berücksichtigen, da sie die Art und Weise des Systemgebrauchs stark beeinflussen.

Ebenso kann jedoch auch der digitale Raum als Ergänzung zum physischen Kontext genutzt werden. Nicht vorhandene physische Ressourcen werden digital bereitgestellt, Artefakte sind digital einfacher zu übertragen als in physischer Form, zeitliche Begrenzungen des physischen Raums werden vom digitalen Raum überbrückt und vieles mehr. Bei der Planung eines Systems sind also neben digitalen Möglichkeiten auch vor allem physische Möglichkeiten, Anforderungen, Restriktionen und Chancen zu betrachten, um eine kontextgerechte Mischform zu erstellen. Dabei können auch Situationen auftreten, in welchen ausschließlich einer der Räume wirklich für das Lernen genutzt wird, auch wenn für die Nutzung digitaler Räume stets ein geeignetes Gerät bereitstehen muss und sich Personen immer in einem physischen Raum befinden.

### 3.3.5. Integration von Ökosystemen und privaten Medienökologien

Letztlich sind die Prinzipien der Ökosysteme und privaten Medienökologien zu berücksichtigen, wobei zunächst die variierenden Begriffsverständnisse noch einmal kurz verdeutlicht werden. Private Medienökologien bilden sich aus unterschiedlichsten Anwendungen, Geräten, digitalen Ressourcen und digitalen Communities, welche von einem Individuum häufig zur Aufgabenbewältigung eingesetzt und zielorientiert kombiniert werden. Somit stellt jeder Bestandteil der Ökologie einen wiederkehrenden und hohen Wert für dieses Individuum dar, wobei sie meist von unterschiedlichen Anbietern stammen und somit keine direkte Kompatibilität aufweisen müssen. So ist Youtube eine häufige Informationsquelle, während Google Docs ein etabliertes Dokumentations-Tool darstellt. Diese werden beim Lernen oft in Kombination genutzt, jedoch ist nativ keine direkte Überführung der Video-Inhalte in ein Google Docs Dokument möglich. Aufgrund der Ungebundenheit sind Medienökologien jedoch flexibel und erweiterbar. Die freie Wahl nach eigenen Präferenzen führt zudem zu einer hohen Anpassungsfähigkeit und Nutzerzentriertheit. Ein Ökosystem hingegen beschreibt eine Ansammlung von Systemen oder System-Komponenten, welche von demselben Anbieter angeboten werden oder zumindest auf einer gemeinsamen Basis aufbauen. Jeder Bestandteil erfüllt eine unterschiedliche Aufgabenstellung, jedoch sind sie als sich ergänzende Komponenten anzusehen, welche eine hohe Kompatibilität aufweisen. Durch die gemeinsame Basis ist ihre Anpassungsfähigkeit und Flexibilität jedoch oft eingeschränkt, sodass sie nicht die Präferenzen jedes Nutzers abdecken können. Beispiele für Ökosysteme sind Produkte und Geräte der Apple-Familie sowie die *Microsoft 365-Suite*<sup>3</sup>.

Im besten Falle werden nun die Bestandteile der bereitgestellten digitalen Räume als digitales Ökosystem konzipiert, welches ebenso eine private Medienökologie repräsentiert, wodurch sie sowohl flexibel und anpassungsfähig, als auch untereinander kompatibel sind. Um dieses möglich zu machen, sollte bei der Planung der Systembestandteile eine einheitliche Basis vorliegen (siehe Abschnitt 4) und bestehende Medienökologien bedacht sowie bestmöglich integriert bzw. unterstützt werden. Zur Bewältigung unterschiedlicher Aufgabentypen haben sich individuelle Praktiken gefestigt, welche auf der Nutzung alltäglicher Informationsquellen, Anwendungen, Geräte und Communities aufbauen. Die Einbindung dieser Bestandteile in die hybride Lehre und somit auch in das hybride System vereinfacht den Nutzern den Lernprozess, macht diesen adaptierbar auf individuelle Bedürfnisse und optimiert dadurch die Zielerreichung. Wurden also wichtige Bestandteile im Designprozess erkannt, so sollte zunächst bedacht werden, ob

---

<sup>3</sup>Microsoft 365: Siehe <https://www.microsoft.com/en-ww/microsoft-365/>, zuletzt gesichtet 23.08.2021

eine Neuentwicklung wirklich von Nöten ist, oder ob bereits vorhandene Anwendungen nutzbar sind. Zu berücksichtigen sind bei dieser Betrachtung allgemeine Informationsquellen und Ressourcen- oder Artefaktspeicher, beispielsweise Plattformen wie Youtube oder Google Drive. Ebenso sind vielgenutzte Anwendungen zu identifizieren, welche bereits eine geplante Aktivität ermöglichen können und entsprechende Integrationsmöglichkeiten für das eigene System bereitstellen. Viele dieser Anwendungen sind bereits Bestandteil einer größeren Infrastruktur, wodurch ihre Integrationen in das eigene System ebenso Chancen für folgende Erweiterungen und eine höhere Kompatibilität darstellen. Weiterhin sind alltägliche Geräte potentieller Nutzer zu unterstützen, damit eine möglichst verbreitete und flexible Nutzung des Systems möglich ist. Letztlich sollten Integrationsmöglichkeiten persönlicher Communities abgewogen werden, wobei typische Kommunikationswege eine erhebliche Rolle einnehmen. Durch die Verbindung des Lernens mit alltäglichen Bestandteilen des Lebens kann eine Übernahme des hybriden Systems in die persönlichen Medienökologien erfolgen.

### 3.3.6. Vollständiges Modell & hilfreiche Fragen beim Designprozess

Unter Berücksichtigung der genannten Planungsdetails lässt sich das, in Abbildung 3.2 dargestellte Übersichtsmodell weiter spezifizieren. In Abbildung 3.3 wird dieses nun mit möglichen Ausprägungen angereichert, welche im Planungsprozess als Hilfestellung dienen können.

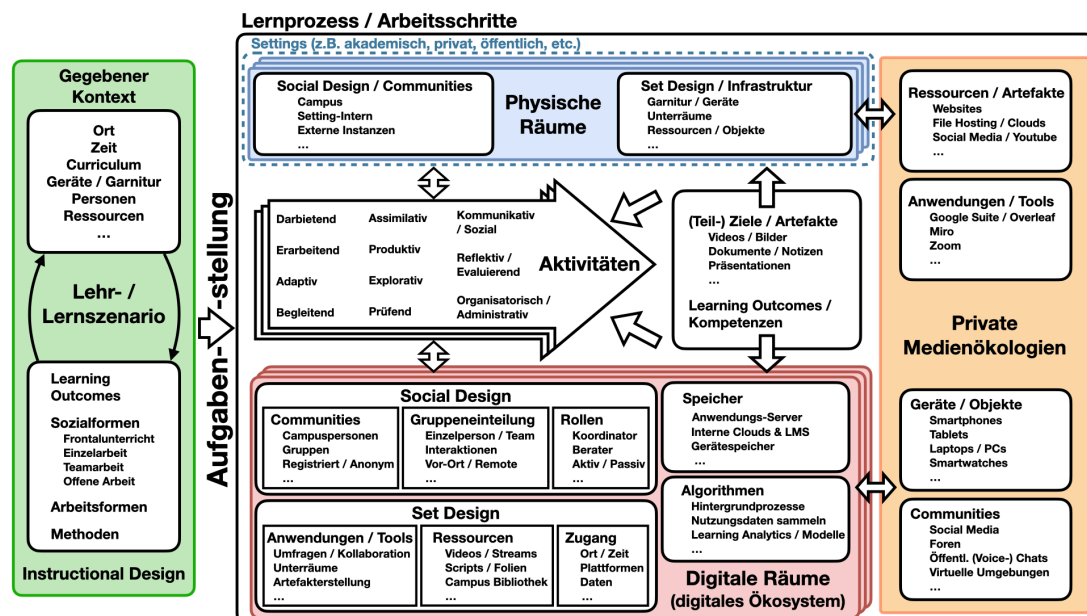


Abbildung 3.3.: Erweitertes Übersichtsmodell für das Design hybrider Systeme

Zur Identifizierung der beschriebenen Bestandteile ist zudem die Nutzung einer zusammenfassenden Anleitung sinnvoll, mithilfe derer gezielt die Planungsschritte und die zugehörigen Entitäten abgearbeitet werden können. Hierfür wurde eine Tabelle mit den wichtigsten Fragen vorbereitet, welche in Anhang C aufzufinden ist.

## 4. Entwicklung eines hybriden Ökosystems

In Kapitel 3 wurde ein Ansatz zur zielgerechten Konzeption hybrider Systeme entworfen, welcher die Ermittlung von Lernaktivitäten als zentralen Aspekt ansieht. Aktivitäten gelten darin als stark domänenspezifisch, sodass gerade diese in unterschiedlichen Lehrszenarien variieren. Es werden jedoch auch feste Bestandteile genannt, welche in jeder hybriden Anwendung vorhanden sein müssen und welche bestenfalls ein Ökosystem bilden sollten. Für diese übergreifenden Entitäten kann demnach ein zentrales System gestaltet werden, welches anschließend als Ausgangspunkt weiterer Entwicklungen nutzbar ist. So befasst sich dieses Kapitel mit der zentralen Forschungsfrage, welche Architektur solch ein grundlegendes System aufweisen muss, um eine flexible Erstellung hybrider Lernräume zu ermöglichen. Die dadurch entstehende Architektur soll zusammen mit dem zuvor definierten Design-Framework die Erstellung kontextgerechter Hybrid-Anwendungen erleichtern, beschleunigen und qualitativ aufwerten. Neben einer hohen Stabilität muss die Architektur dafür besonders eine hohe Flexibilität und Erweiterbarkeit aufweisen, sodass eine Adaption und Anpassung auf neue Kontexte gewährleistet wird. Erst hierdurch kann das System stetig wachsen, den wechselnden Anforderungen der Umwelt gerecht werden und schließlich einer vielseitigen Medienökologie mit umfassenden Angeboten entsprechen.

So werden zunächst die bereits identifizierten Entitäten und Eigenschaften in ein technisches Übersichtsdiagramm überführt, in welchem ihre gegenseitigen Abhängigkeiten und Klassenstrukturen ersichtlich sind. Daraufhin findet die Auswahl eines geeigneten Anwendungstypen und Architekturstils statt. Notwendige Bestandteile des ausgewählten Architekturstils werden anschließend detailliert erläutert und dafür vorteilhafte Technologien aufgelistet. Auf diesen Wahlen aufbauend findet abschließend die Präsentation und Begründung der ermittelten Systemarchitektur statt.

### 4.1. Domänen-Modell der Bestandteile

In vorherigen Abschnitten wurden bereits die wichtigsten Bestandteile und Eigenschaften hybrider Lehr- / Lernsysteme herausgearbeitet und beschrieben, welche es nun bei einem Architekturdesign zu berücksichtigen gilt. Abschnitt 2.4 fokussierte allgemeine



---

Bestandteile hybrider Lehre, während Abschnitt 3.3 Entitäten für eine kontextgerechte Identifikation geeigneter Lernaktivitäten nannte. Bei der Überführung auf technische Systemkomponenten sind jedoch ebenso technische Möglichkeiten und Standards zu bedenken, weshalb variierende Konzepte auftreten können. Die Erstellung eines abstrakten Klassendiagramms ist deshalb nach der Domänenanalyse ein hilfreicher, erster Schritt für die Ermittlung technischer Anforderungen des Themenfeldes. In Abbildung 4.1 ist nun solch ein abstraktes Klassendiagramm für die Kernbestandteile hybrider Lernraumsysteme aufgeführt. Es beinhaltet vier logisch separierbare Abschnitte (A, B, C und D), welche in den folgenden Unterkapiteln erläutert werden. Zu beachten ist dabei, dass diese Form der Visualisierung noch keine vollständig pragmatische Architektur darstellt und vorwiegend der Identifikation von Abhängigkeiten und Abläufen dient. So wird darin lediglich zwischen konkreten Klassen und Abstraktionen (Enumerationen und Interfaces) unterschieden. Für diese Abstraktionen werden zwar beispielhaft Implementierungen genannt, Klassen allgemein jedoch nur namentlich und ohne zugehörige Attribute oder Methoden aufgeführt. Die Beziehungen zwischen diesen Bestandteilen sind weiterhin abstrahiert, sodass lediglich zwischen Vererbungen / Implementierungen (gestrichelte Linie, leere Pfeilspitze) und unidirektionalen Relationen (durchgezogene Linie, gefüllte Pfeilspitze, Multiplizität) unterschieden wird. Bei der Implementierung des Ansatzes sind die aufgeführten Bestandteile demnach als Oberkategorien anzusehen, zu deren Umsetzung viele Teilentitäten notwendig sind.

#### 4.1.1. A: Akteurverwaltung

Im orangenen Abschnitt *A* befinden sich die wichtigsten Klassen zur Darstellung von Nutzern des Systems. Nutzer selbst sind unter dem Interface *User* zusammengefasst, welches zwei Implementierungen aufweist. Zunächst müssen sich Nutzer in diesem System registrieren können, um so eine Speicherung und Fortsetzung des Lernprozesses zu ermöglichen. Ebenso ist in Abschnitt 2.4 auch angesprochen worden, dass eine ad-hoc Verwendung des Systems, sowie Einbindung weiterer Nutzer möglich sein sollen, wofür neben den *Registered*-Nutzern auch *Anonymous*-Nutzer vorgesehen sind. Dabei kann es sich um feste externe Parteien handeln, welche an ganzen Lehrveranstaltungen, mehreren Arbeitsschritten, individuellen Lernräumen oder einzelnen Aktivitäten partizipieren. Es können jedoch auch Lernende zunächst als anonyme Nutzer einen Lernprozess planen oder diesem beiwohnen und erst im späteren Verlauf einen Account erstellen. Die Einbindung anonymer Nutzer ist jedoch vor allem für eine kurzfristige Einbindung externer Personen vorgesehen. So ist es beispielsweise für interdisziplinäre Veranstaltungen möglich, Gastdozenten dynamisch für eine Vorlesung einem Kurs hinzuzufügen. Ebenso könnten Umfrageteilnehmer kurzfristig als anonyme Nutzer für eine Aktivität

hinzugefügt werden, beispielsweise mithilfe von QR-Codes oder Links.

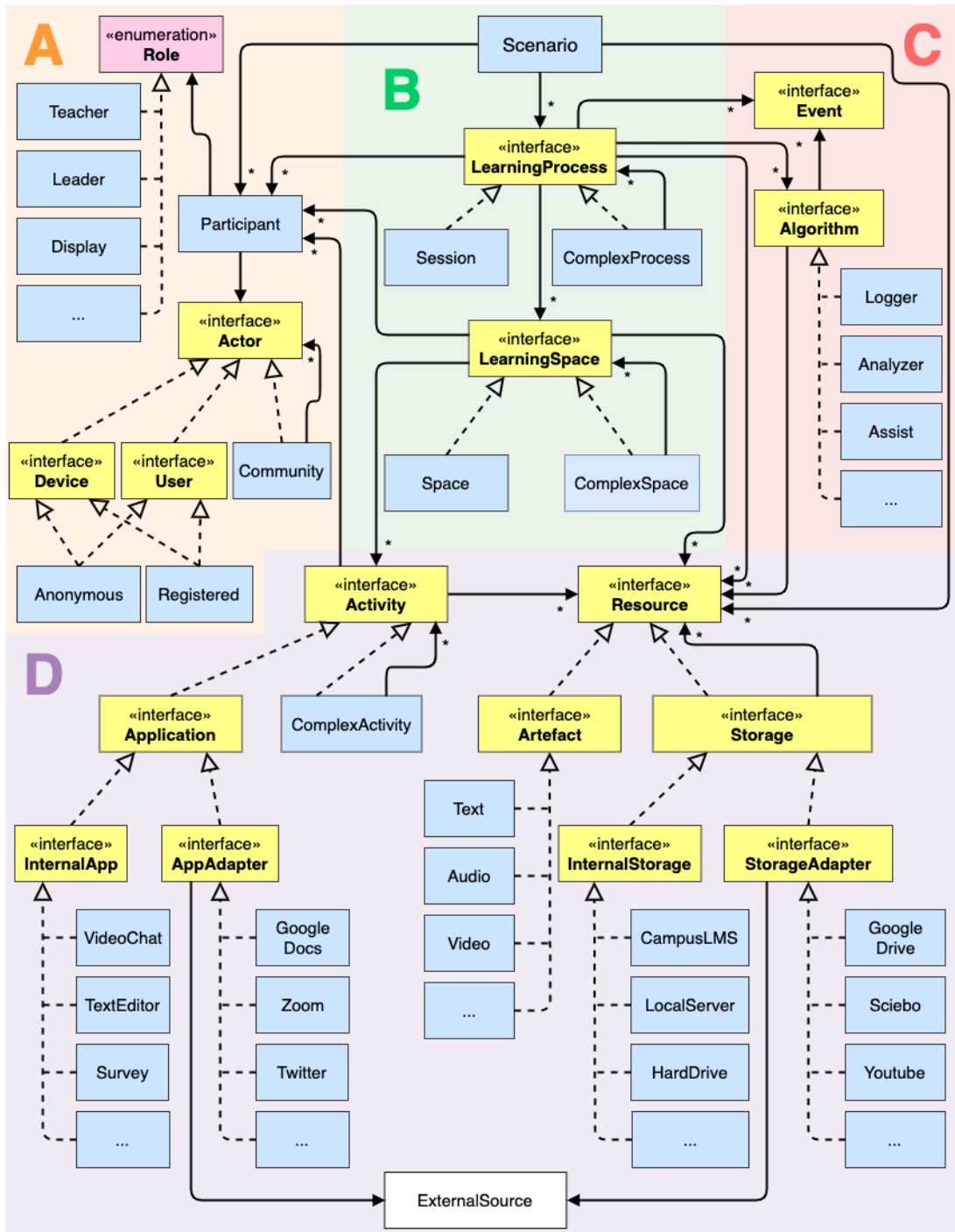


Abbildung 4.1.: Abstraktes Klassendiagramm beinhalteneter Entitäten

---

Doch nicht nur einzelne Nutzende, sondern auch Geräte (*Device*) sollten vom System erfassbar und nutzbar sein. Auch diese können anonym und kurzfristig hinzugefügt werden, beispielsweise ein Tablet, welches ad-hoc als Präsentationsmedium dient, oder aber langfristig im System registriert sein, was beispielsweise für fest montierte Bildschirme, Beamer oder Kameras eines Vorlesungsraums nützlich ist. Ebenso sollten ganze Communities darstellbar sein. Diese bestehen aus mehreren Nutzern und Geräten, können jedoch auch Teil-Communities beinhalten. So findet sich beispielsweise die Campus-Community mit allen Lernenden, welche sich selbst aus verschiedenen Studienrichtungen bzw. deren Communities zusammensetzt, zu denen wiederum unterschiedlichen Jahrgangs- / Semester-Communities gehören. Weiterhin können auch Kurse, Teams oder Lerngruppen als kleine Communities dargestellt werden, welche wiederum unterschiedliche Nutzer beinhalten und auf fest montierte Geräte zugreifen können. So wurde *Actor* als ein gemeinsames Interface für *User*, *Device* und *Community* eingeführt, da all diese Entitäten aktiv im System agieren. Communities gelten dabei als zugehöriges *Kompositum*<sup>1</sup>, um die Verschachtelung darzustellen.

Letztlich wurde in den Kapiteln 2.4 und 3.3 die Rolleneinteilung angesprochen, welche innerhalb ganzer Veranstaltungen, unterschiedlicher Arbeitsschritte oder einzelner Aktivitäten vorgenommen wird. Derselbe Nutzer kann innerhalb eines Lernszenarios sequentiell oder parallel unterschiedliche Rollen einnehmen, welche dann mit ebenso variierenden Berechtigungen verbunden sind. Als Beispiel gilt ein Nutzer, welcher innerhalb eines Kurses als Lernender eingeteilt ist und somit ausschließlich nutzende, jedoch keine administrativen Rechte erhält. Werden jedoch Gruppenarbeiten in diesem Kurs vollzogen, dafür zugehörige Lernräume erstellt und dieser Nutzer in einem dieser Lernräume als Teammitglied oder Gruppenleiter bestimmt, so könnten ihm innerhalb des Raumes und der Unterräume administrative Funktionen zugeteilt werden. Rollen gelten also immer für die Ebene, auf welcher sie vergeben wurden und für alle darunter liegenden Ebenen / Verschachtelungen, außer es werden für diese unteren Ebenen neue und stärkere Rechte verteilt. So lassen sich über dieses Rollensystem auch die Dimensionen der Strukturierung und Flexibilität für gezielte Personen, Prozesse und Räume beeinflussen. Um für bestimmte Situationen neue Rechte vergeben zu können, wurde die Zwischenklasse *Participant* als Bindeglied zwischen der *Role* und dem *Actor* eingeführt. Durch diese Konstellation kann auch ganzen Gruppen / Communities eine einheitliche Rolle zugewiesen werden, beispielsweise die Rolle der *Lernenden* innerhalb einer Veranstaltung.

---

<sup>1</sup>Für eine Erläuterung des Kompositum-Musters kann Freeman et al. [2008, S. 356 ff.] betrachtet werden.

Übergreifend für Szenarien, Prozesse und Lernräume können einige Standardrollen statisch angeboten werden. Es finden sich jedoch auch innerhalb einzelner Aktivitäten dedizierte Rollen, welche nicht übergreifend nutzbar sind. So ist beispielsweise bei einem Brainstorming die Rollenaufteilung in Moderator, Protokollführer und Diskutierende denkbar. Ebenso könnten unterschiedliche Geräte als Kamera, Display oder Eingabegerät deklariert und ihre Berechtigungen sowie ihr Datenverkehr dadurch zielorientiert angepasst werden. Rollen stellen also ein grundlegendes Element der Architektur dar, welches im Gegensatz zu den bisher genannten Bestandteilen auch aktivitätsbasierte Erweiterungen gewährleisten muss.

#### **4.1.2. B: Infrastruktur des Lernens**

Im Zentrum des dargestellten Diagramms befindet sich nun der grüne Abschnitt *B*, welcher fest in einem grundlegenden System definiert und umgesetzt werden sollte. Als Ausgangspunkt des Raumdesigns findet sich hier zunächst das *Scenario*, was die Repräsentation des Lernszenarios darstellt. So kann beispielsweise für einen Kurs oder eine Projektarbeit, aber auch für private Lernkontexte ein entsprechendes *Scenario* angelegt werden. Diesem sind dann Teilnehmer (*Participant*) zuordenbar, welche sich in einer Lehrveranstaltung vor allem aus den Lehrenden und Lernenden zusammensetzen. Ebenso können *Ressourcen* (siehe Abschnitt 4.1.4) einem *Scenario* beispielsweise als Aushängematerial der Kursinformationen oder Terminplanung hinzugefügt werden.

Ein *Scenario* führt zudem Referenzen zu beinhalteten Lernprozessen (*LearningProcess*). Lernprozesse können wiederum aus einem oder mehreren Arbeitsschritten bestehen, sowie mehrere Teilnehmer beinhalten. Einzelne Arbeitsschritte sind über eine *Session* darstellbar und eine Sequenz mehrerer Schritte über einen *ComplexProcess*. Dieser ist wiederum ein Kompositum von *LearningProcess*-Objekten, wodurch nicht nur sequentielle, sondern auch verschachtelte Lernprozesse darstellbar sind. So findet sich in einem Kurs beispielsweise ein grundlegender und zwischen allen Teilnehmern geteilter Lernprozess, welcher zunächst verschiedene Vorlesungen als einzelne *Sessions* beinhaltet. Werden anschließend Gruppen gebildet, so kann pro Gruppe ein verschachtelter Lernprozess genutzt werden, welcher wiederum abhängig von den darin beinhalteten Arbeitsschritten verschiedene Kindprozesse aufweisen kann. Diese komplexen Prozesse könnten auch als Template gespeichert werden, um diese in anderen Szenarien wiederverwenden zu können. Weiterhin können auch mit einem Prozess *Ressourcen* verbunden werden, beispielsweise grundlegende Lernmaterialien wie Skripte, Lernvideos und Präsentationsfolien, aber auch dedizierte Materialien für die verschach-

---

telten Lernprozesse einzelner Gruppen.

Für jeden Lernprozess können zudem kontextgerechte Lernräume (*LearningSpace*) erstellt und diesen Teilnehmende zugeordnet werden. So können Untergruppen in demselben Prozess verschiedene Lernräume nutzen. Lernräume sind aber ebenso über mehrere Sessions hinweg nutzbar. Auch hier sind wiederum einzelne Räume (*Space*) und ein verschachtelbares Kompositum (*ComplexSpace*) zur Unterraumbildung eingeplant. Letzteres ist wieder als Template speicherbar. Ähnlich den Szenarien und Lernprozessen können auch für Lernräume dedizierte Ressourcen festgehalten werden. Die tatsächliche Bedeutung erhält ein *LearningSpace* jedoch erst durch die Zuordnung von *Aktivitäten* (siehe Abschnitt 4.1.4).

### 4.1.3. C: Hintergrundprozesse

Die in Abschnitt 2.4 angesprochenen Algorithmen und Hintergrundprozesse bilden, zusammen mit *Events* den roten Abschnitt *C* des Diagramms. Die *Algorithm* stehen in direkter Verbindung zu den Lernprozessen, welchen sie als *Beobachter*<sup>2</sup> zugeordnet sind. So können Algorithmen Informationen über alle *Events* eines Lernprozesses und beinhalteteter Elemente erhalten, solche Events jedoch auch selbst erstellen um dadurch mit dem Prozess zu kommunizieren. Dies können allgemeine Interaktionsdaten eines Nutzers mit dem System sein, welche innerhalb eines Logs festgehalten werden. Es können aber auch raum- oder aktivitätsspezifische Events verarbeitet werden. Durch die Kombinationen von Aktivitäten und Algorithmen sind auch Bestandteile wie intelligente Assistenten denkbar, welche in Echtzeit auf Aktionen reagieren und Maßnahmen einleiten können. Die Verbindungen dieser Elemente gelten also ebenfalls als wichtige Beziehungen, wurden jedoch zugunsten der Übersichtlichkeit des Diagramms ausgelassen, zumal auch sie eine Verbindung zum Lernprozess aufweisen und dieser somit als Kommunikationsknoten nutzbar ist. Ähnlich den Rollen muss also neben allgemeinen Hintergrundprozessen auch eine kontextspezifische Entwicklung und Integration eigener Algorithmen vom System ermöglicht werden. Algorithmen weisen zudem eine Verbindung zu Ressourcen auf, da ihre Nutzung als Grundlage weiterer Analysen dienen können und Ressourcen ebenfalls durch Algorithmen erstellbar sind. Der Lernprozess kann beispielsweise als Log-Datei der wichtigsten Aktionen und Interaktionen festgehalten und aus diesen zu späteren Zeitpunkten rekonstruiert werden.

---

<sup>2</sup>Für eine Erläuterung des Beobachter-Musters kann Freeman et al. [2008, S. 44 ff.] betrachtet werden.

---

#### 4.1.4. D: Lernaktivitäten und -ressourcen

Der letzte und größte Abschnitt des Diagramms (Abschnitt *D*) beinhaltet die zwei Elemente, welche den hybriden Gedanken erst ermöglichen.

Im Sinne einer Medienökologie müssen unterschiedliche Aktivitäten als kleine Bestandteile zur Verfolgung verschiedener Zielstellungen dynamisch und flexibel erstellt, sowie kombiniert werden können. Dafür sind im System zwei Implementierungsformen einer *Activity* vorgesehen. Die *ComplexActivity* ermöglicht als Kompositum eine Kombination bestehender Funktionalitäten. Aktivitäten stellen allgemein die Bedeutung hinter einem Lernraum dar, weshalb es für unterschiedliche Situation auch wiederkehrende Strukturen gibt. Diese sind hier über eine *ComplexActivity* darstellbar und können dadurch auch als Vorlage (*Template*) festgehalten werden.

Die einzelnen Teilaktivitäten bzw. Funktionen sind als *Application* aufgelistet, welche wiederum ein Interface mit zwei Implementierungen darstellt. Die erste hiervon ist die *InternalApp*. Dazu gehören jene Applikationen, welche ausschließlich auf systemeigene Infrastrukturen zugreifen bzw. dedizierte Anwendungen des Systems verkörpern. Die zweite Variante sind die *AppAdapter*, welche auf den Schnittstellen bzw. Funktionalitäten externen Anwendungen aufbauen und diese auf die Anforderungen und Strukturen des Systems abstimmen. Dem Namen entsprechend folgen sie dem *Adapter-Pattern*<sup>3</sup>, um die Nutzung privater Medienökologien zu ermöglichen, ihre Kompatibilität mit der hybriden Anwendung herzustellen und so eine Interoperabilität verschiedener Ökosysteme zu gewährleisten. So ist beispielsweise über die Anbindung eines Forums oder einer Social Media Instanz auch die Integration externer Communities möglich.

Auch wenn sowohl *InternalApp* als auch *AppAdapter* als eigenständige Interfaces skizziert sind, muss eine Unterscheidung bei ihrer Implementierung nicht zwingend erfolgen. Sie wurden vor allem als getrennte Bestandteile aufgeführt, um die Varietät möglicher Implementierungen aufzuzeigen und erneut auf die Integrationsmöglichkeiten externer Anwendungen hinzuweisen. Innerhalb des Systems stellen sie jedoch dieselbe Schnittstelle bereit und variieren lediglich in ihren internen Implementierungen bzw. in der Kommunikation mit externen Systemen. So sind beide Arten von Applikationen innerhalb des Systems als *Strategien*<sup>4</sup> anzusehen, welche dynamisch zur Gestaltung eines Lernraums genutzt und kombiniert werden können. Besonders in Bezug auf Aktivitäten ist die flexible Erweiterbarkeit des hybriden Systems zu beachten. Werden

---

<sup>3</sup>Für eine Erläuterung des Adapter-Musters kann Freeman et al. [2008, S. 235 ff.] betrachtet werden.

<sup>4</sup>Für eine Erläuterung des Strategie-Musters kann Freeman et al. [2008, S. 558 ff.] betrachtet werden.

neue Aktivitäten benötigt, sollten diese problemlos integriert werden können, vorausgesetzt sie erfüllen die Bedingungen des *Application*-Interfaces. Denkbar ist zudem die Erweiterung der *Activity* um zugehörige Aktivitätstypen (siehe Abschnitt 3.3), welche für logische Gruppierungen in einem *User Interface (UI)* nutzbar sind und so die Suche und Auswahl geeigneter Aktivitäten vereinfachen können.

Das zweite Element ist die *Resource*, welche bereits in Zusammenhang mit Lernprozessen, Lernräumen und Algorithmen erwähnt wurde. Die *Resource* repräsentiert sämtliche Informationsquellen und Lernmaterialien, welche im System abgebildet werden müssen. Bei Ressourcen wird wiederum zwischen zwei Implementierungen unterschieden, dem *Artefact* und dem *Storage*, welche beide wiederum Interfaces sind. Das *Artefact* stellt eine einzelne Datei dar und weist demnach viele unterschiedliche Typen bzw. Implementierungen auf. So sollten unterschiedliche Formate von Text-, Audio- und Video-Dateien unterstützt werden, jedoch auch solche für Präsentationsfolien und PDF-Dateien. Ebenso können innerhalb einzelner Anwendungen eigene Formate notwendig sein, sodass auch hier eine Anreicherung des Systems mit neuen Artefakt-Typen ermöglicht werden muss. Artefakte und Ressourcen werden sowohl in der Entitätsübersicht (siehe Abschnitt 2.4), als auch in dem Design-Framework (siehe Abschnitt 3.3) als getrennte Entitäten dargestellt, auch wenn die Nutzungsmöglichkeit von Artefakten als Ressource bereits angesprochen wurde. Diese Unterscheidung ist bei der theoretischen Konzeption eines Systems sinnvoll, da durch die unterschiedlichen Einsatzgebiete ein breiteres Spektrum des Themenkomplexes erforscht wird. Aus einer technischen Perspektive ist eine Unterscheidung jedoch nicht sinnvoll, da sie auf denselben grundlegenden Datei-Typen aufbauen und sich ausschließlich ihr didaktischer Einsatz unterscheidet.

Als zweite Ausprägung der *Resource* gilt das *Storage*-Interface, welches selbst wiederum als Kompositum von Ressourcen gilt. So sind über einen *Storage* komplexe Ressourcen-Speicher wie Ordner, Clouds oder Festplatten darstellbar. Auch wenn Speicher keine eigenen Dateien repräsentieren, so bestehen sie ausschließlich aus solchen, sind für Abbildungen von Ressourcen-Hierarchien nutzbar und verfolgen dasselbe Ziel der Bereitstellung und Persistierung von Wissen. Aus einer technischen Sichtweise finden sich dieselben Einsatzgebiete wie für Artefakte, weshalb auch Speicher hier als Ressource definiert sind. Bei der Implementierung des Ansatzes kann eine gesonderte Behandlung jedoch wieder sinnvoll sein, da die Behandlung von Artefacts und Storages häufig variiert. Das *Storage*-Interface weist zwei Ausprägungen auf, welche ähnlich der aufgeführten Applikations-Typen vor allem für eine logische Unterscheidung möglicher Ansätze aufgeführt sind. Bei der Implementierung muss somit keine zwingende Unter-

scheidung dieser Subtypen erfolgen. Als erster Subtyp ist das *InternalStorage*-Interface dargestellt, welches ähnlich der *InternalApp* systemeigene Speichermöglichkeiten für Ressourcen gruppiert. Der *StorageAdapter* hingegen fokussiert wieder die Integration externer Quellen und Speichermöglichkeiten in das hybride System, weshalb auch dies als Adapter betitelt und dargestellt wird. Für spezielle Kontexte ist es so möglich, dort verwendete Speicherstrukturen auch in diesem System einzubinden, sodass eine flexible Erweiterbarkeit geboten wird.

## 4.2. Anwendungstyp und Architekturstil

Neben der Identifikation wichtiger Systemkomponenten und ihrer Beziehungen sind vor der Definition einer ausführlichen Architektur zuerst der benötigte Anwendungstyp und Architekturstil festzulegen.

Bezüglich des Anwendungstypen wurden in diesem Dokument bereits einige Anforderungen genannt. So soll eine stetige Erreichbarkeit gewährleistet werden, weshalb das System eine orts- und zeitunabhängige, sowie geräteübergreifende Nutzung ermöglichen muss. Es darf also nicht nur in vorbereiteten Räumen nutzbar sein, sondern soll von überall und auch in privaten Kontexten mit privaten Geräten verwendet werden können. Dabei gilt die ad-hoc Nutzung des Systems als wichtiger Hinweis, sodass vorangehende Installationsprozesse einer Software vermieden werden sollten. Auch wurde im Zusammenhang mit Medienökologien die Verwendung verbreiteter Strukturen, Zugangsmethoden und Anwendungen genannt, welche aktuell zu großen Teilen eine Webansicht bereitstellen. Letztlich ist das Ziel die Schaffung eines Systems, welches die Entwicklung kontextgerechter Lösungen erleichtert und durch eine gemeinsame Basis ein stetig wachsendes Ökosystem darstellt. So sollten auch Erweiterungen, unabhängig vom Ort ihrer Erstellung und den dafür genutzten Infrastrukturen, für alle Nutzer an einem zentralen Ort bereitgestellt werden. All diese Aspekte sprechen für eine Implementierung als Webanwendung. Speziell die geräteübergreifende Nutzung ist hier zu berücksichtigen. Wird beispielsweise für jedes Betriebssystem eine eigene Anwendung erstellt, so können zwar systemspezifische Möglichkeiten berücksichtigt werden, die Entwicklung würde jedoch weitaus mehr Zeit in Anspruch nehmen. Dies würde auch dazu führen, dass Erweiterungen des Systems auf die unterschiedlichen Untersysteme eingehen müssten. Geschieht dies nicht, so würden lediglich Nutzer eines Betriebssystems von der Erweiterung profitieren und nicht eine zentrale, sondern systemspezifische Medienökologien entstehen. Dennoch sollte das System auch auf die Bedürfnisse einzelner Nutzer eingehen können, weshalb eine Schaffung ergänzender



Anwendungen und neuer Front-Ends weiterhin berücksichtigt werden sollte.

Es muss also ein Backend mit möglichst offener und flexibler Schnittstelle bereitgestellt werden. Besonders unter Anbetracht der zuvor definierten Klassenstrukturen (siehe Abschnitt 4.1) ist auch die flexible Erweiterbarkeit der Webanwendung bzw. dessen Backends notwendig. Sowohl Rollen als auch Algorithmen, Aktivitäten und Ressourcen müssen dem System dynamisch hinzugefügt werden können. Da Erweiterungen nicht auf einzelne Entwicklerteams begrenzt sind, sollten diese auch im laufenden Betrieb vorgenommen werden können. Es darf also keine statische und monolithische Architektur geplant werden, welche für Neuerungen stets einen Neustart des Systems erfordert und allgemeine Probleme mit Skalierungen aufweist. Monolithische Architekturen haben zudem den Nachteil, dass diese komplex und unübersichtlich werden können, was eine ad-hoc Adaption auf eigene Kontexte erschwert. Zudem ist eine Abgrenzung der Systemabschnitte innerhalb eines Monolithen nur schwer umsetzbar, was gerade bei dynamischen Erweiterungen durch unbekannte Personen Sicherheitsrisiken bergen kann. Es muss also eine hohe Modularisierung und Isolation der dynamisch ergänzten Systembestandteile von dem Kernsystem ermöglicht werden. Aufbauend auf diesen Anforderungen wird der *Microservice*-Architekturstil als Grundlage der Architekturplanung ausgewählt und damit die Realisierung eines dynamischen „Plugin“-Systems angestrebt.

Bevor jedoch darauf aufbauende Systemplanungen präsentiert werden, widmen sich die folgenden Unterkapitel zunächst der Erläuterung von Microservices, ihren wichtigsten Bestandteilen und möglichen Technologien für ihre Umsetzung.

#### **4.2.1. Grundlagen des Microservice-Ansatzes**

Ähnlich der hybriden Lehre findet sich auch für Microservices keine einheitliche Definition. Doch ist eine grundlegende Gemeinsamkeit in Form der Separierung von Anwendungen in einzelne Prozesse ersichtlich. So beschreibt Adrian Cockcroft, auf Basis seiner Erfahrungen bei Netflix, Microservices als serviceorientierte Architektur, welche sich aus lose gekoppelten Elementen mit begrenztem Kontext zusammensetzt [vgl. Richardson, 2018, S. 8]. Wolff [vgl. 2018, S. 2] beschreibt sie als einen Ansatz zur Modularisierung von Software, bei dem Module in einzelne Programme ausgelagert werden, welche als selbstständige Prozesse laufen. So werden große Software-Systeme aufgeteilt, wodurch auch Organisationsstrukturen und Software-Entwicklungsprozesse betroffen und anzupassen sind. Auch Richardson [vgl. 2018, S. 11 f.] definiert einzelne Services als Einheit der Modularisierung dieses Architekturstils. Funktionalitäten einer

Anwendung werden in dedizierte Services ausgelagert, sodass jeder Service einen fokussierten und kohärenten Zuständigkeitsbereich aufweist. Die Modularisierung erfolgt ebenso durch eine lose Kopplung der Services. Jeder Service bietet eine öffentliche API an. Kommunikationen finden ausschließlich über diese Schnittstellen statt, wodurch interne Implementierungsdetails isoliert bleiben. Ebenso sind andere lose Netzwerkkontakte möglich, beispielsweise solche über ein asynchrones Messaging-System. Als weiteren wichtigen Punkt der losen Kopplung nennen sowohl Wolff [vgl. 2018, S. 2] als auch Richardson [vgl. 2018, S. 12] die Nutzung eigener Datenbanken oder vollständig isolierter Schemata innerhalb derselben Datenbank. Hierdurch werden Race-Conditions und ungewollte Manipulationen zwischen Services vermieden und Änderungen, wie die Verwendung neuer Datenbank-Technologien, können isoliert für einen Service vollzogen werden.

Es wurden bereits einige Vorteile dieses Architekturstils angesprochen. Als weitere Vorteile nennt Richardson [vgl. 2018, S. 14 ff.] die Wartbarkeit der einzelnen Services, da diese nur eine geringe Komplexität aufweisen und isoliert im System integriert sind. Damit verbunden werden auch das unabhängige Deployment und die unabhängige Skalierbarkeit einzelner Services genannt. Durch die Modularisierung werden zudem autonome Teams gefördert, Fehler eines Services isoliert, unabhängige Tests vereinfacht und die Technologiewahl pro Service vollständig freigestellt.

„Es gibt keine Einschränkung auf eine bestimmte Programmiersprache oder Plattform.“ [Wolff, 2018, S. 2]

So ist das Experimentieren mit und die Adaption von neuen Technologien möglich, wodurch auf den stetigen technischen Wandel eingegangen werden kann. Auch Wolff [vgl. 2018, S. 3 ff.] nennt eine leichte Ersetzbarkeit einzelner Services und dadurch nachhaltigere Software-Entwicklung als Vorteile. Ebenso bekräftigt er eine schnellere Auslieferung von Produkten und eine einfachere Integration von *Continuous Delivery*-Strategien, da „Änderungen an einem Microservice [...] unabhängig von Änderungen an anderen Microservices in Produktion gebracht werden“ können [Wolff, 2018, S. 2].

Doch finden sich auch einige Nachteile dieses Ansatzes, welche ebenfalls von Richardson [vgl. 2018, S. 17 ff.] und Wolff [vgl. 2018, S. 5 f.] angesprochen werden. Zunächst ist die Findung eines kontextgerechten Sets von Services und die Eingrenzung ihrer Zuständigkeitsbereiche eine große Herausforderung, sowie verbunden mit langen Konzeptionsprozessen. Weiterhin sind verteilte Systemarchitekturen komplex, wodurch auch ihr Betrieb mit einem hohen Verwaltungsaufwand verbunden ist und Backup-Pläne für

den Ausfall einzelner Services vorhanden sein müssen. Die Planung serviceübergreifender Funktionen, sowie ihre Entwicklung, ihr Refactoring und Veröffentlichungen sind demnach bedacht und stark koordiniert durchzuführen. Letztlich birgt auch die Modularisierung und Isolation der Services Problematiken. Beziehungen zwischen Services sind aufgrund der losen Kopplung und Kommunikationsstrukturen oft versteckt, sodass Auswirkungen einer Aktion übersehen oder zu spät wahrgenommen werden könnten. So ist zwar ein isoliertes *Unit-Testing* einfach, ein übergreifendes *Integration-* oder *System-Testing* jedoch weitaus aufwändiger und komplizierter.

Weiterhin wurden die wichtigsten Bestandteile einer Microservice-Architektur mithilfe einer ausführlichen Literaturrecherche ermittelt und beschrieben. Diese sind in Anhang 3.3 aufgeführt und können bei auftretenden Fragen eingesehen werden.

#### **4.2.2. Auswahl kritischer Technologien**

Ein großer Vorteil der Microservice-Architektur liegt in der Technologiefreiheit jedes einzelnen Services. Damit diese jedoch interagieren können, muss eine gemeinsame Basis mit einem einheitlichen Ansatz vorhanden und bekannt sein. Gerade bei kommunikationsbedingten Technologien und den zentralen Interaktionsknoten sind Vorauswahlen und genaue Spezifikationen wichtig, da Services von unterschiedlichen Instanzen erstellt und dynamisch in das Gesamtsystem integriert werden. Erst durch diese gemeinsame Basis kann sich mit der Zeit ein zusammenhängendes und kompatibles Ökosystem von unterschiedlichen Services bilden. In den folgenden Abschnitten werden deshalb für zentrale Elemente und Kommunikationswege Vorauswahlen nützlicher Technologien angesprochen.

#### **Externe Kommunikation**

Die Kommunikation zwischen Clients bzw. dem Frontend und dem System findet zu großen Teilen synchron statt, kann jedoch auch asynchrone Kommunikationen, beispielsweise über *Web Sockets* beinhalten. Es ist wichtig, dass diese Kommunikation nach außen ein bekanntes Protokoll, sowie eine etablierte und für viele Programmiersprachen angebotene Schnittstellenspezifikation unterstützt. Eine *REST-Schnittstelle* würde diese Anforderungen beispielsweise erfüllen. In dem vorliegenden Kontext sind jedoch auch weitere Aspekte wichtig, darunter vor allem die flexible Erweiterung und Adaption auf eigene Anwendungsfälle. Neue Services werden dynamisch ergänzt, wodurch auch neue Datenstrukturen und Verknüpfungen zwischen diesen Strukturen entstehen können. Spezielle Frontends können dann für einzelne Clients und Services notwendig sein. Ebenso können neue Services für die Darstellung nur Teile der Daten an-

---

derer Services benötigen. Da diese Anforderungen zu Beginn noch nicht feststehen, ist eine Vorbereitung von API Gateways, welche nach dem Prinzip *Backend For Frontend* auf einzelne Services und Clients abgestimmt sind, nicht vollends möglich. Um Schnittstellen dennoch dynamisch zu halten und den Overhead einer Kommunikation durch unnötige Daten zu reduzieren, sollten Clients bei einer Anfrage definieren können, welche Attribute einer Datenstruktur sie von dem Service benötigen. Auch dies ist über etablierte Schnittstellenspezifikationen wie REST möglich, beispielsweise über *Query-Parameter* oder *Reflections*, welche einfach über bestehende Bibliotheken eingebunden werden können. Dennoch stellen beide Alternativen eine erhöhte Komplexität dar, sowohl für die Nutzer durch eine unübersichtliche URL, als auch für Entwickler durch die zusätzliche Entwicklungszeit und Verwaltung des größeren Technologie-Stacks. In solchen Situationen bietet sich nun die Nutzung der *GraphQL*-Schnittstellenspezifikation an, welche bereits von vielen großen Unternehmen wie *Airbnb*<sup>5</sup>, *GitHub*<sup>6</sup>, *Twitter*<sup>7</sup>, *Paypal*<sup>8</sup> und *Netflix*<sup>9</sup>, adaptiert wurde.

*GraphQL* wurde 2012 von Facebook entwickelt, ist eine stark typisierte Abfragesprache für APIs und zeitgleich eine serverseitige Laufzeitumgebung. Serverseitig wird die Schnittstelle innerhalb eines *Schemas* definiert, welches die Datenstrukturen und Operationen für Clients bereitstellt. Die synchrone Kommunikation findet über *HTTP-Requests* statt, es werden jedoch auch asynchrone *Subscriptions* angeboten, welche meist auf *WebSockets* basieren. Drei der wichtigsten Grundprinzipien der *GraphQL*-Spezifikation von Facebook [2018] sind:

- Die Schnittstelle ist hierarchisch als Graph im JSON-Format aufgebaut. Datenstrukturen sind demnach verschachtelt, wodurch Zusammenhänge und logische Verknüpfungen ersichtlich werden. Auch Abfragen von Clients nutzen diese Hierarchie, wodurch eine natürliche Überführung erreicht wird.
- Clients spezifizieren, anhand der vom Server gegebenen Möglichkeiten, bei Anfragen selbst, welche Daten bzw. Attribute sie von einer Entität / Ressource benötigen. Durch die Verknüpfung und Verschachtelung der Daten kann so eine flexible Datenaggregation mit einer minimalen Anzahl von Anfragen erreicht werden, wodurch auch die Datenmenge auf ein minimal benötigtes Maß reduziert wird.
- *GraphQL* ist introspektiv und stark typisiert. Das Typensystem eines Servers ist dadurch durchsuchbar, wodurch für Clients eine genaue Definition aller Möglich-

---

<sup>5</sup>Airbnb: Siehe <https://www.airbnb.com/>, zuletzt gesichtet 23.08.2021

<sup>6</sup>GitHub: Siehe <https://github.com/>, zuletzt gesichtet 23.08.2021

<sup>7</sup>Twitter: Siehe <https://twitter.com/>, zuletzt gesichtet 23.08.2021

<sup>8</sup>Paypal: Siehe <https://www.paypal.com/>, zuletzt gesichtet 23.08.2021

<sup>9</sup>Netflix: Siehe <https://www.netflix.com/>, zuletzt gesichtet 23.08.2021

keiten vorliegt. Dies ermöglicht zudem eine einfache Erstellung unterstützender Werkzeuge, welche mit allen GraphQL-Schnittstellen interagieren können.

Wichtig ist, dass GraphQL lediglich eine Spezifikation darstellt, also flexibel auf alle Programmiersprachen überführt werden kann. So finden sich für alle gängigen Programmiersprachen eigenständige Frameworks und Bibliotheken, welche jedoch aufgrund der gemeinsamen Grundlage vollständig kompatibel bleiben. Zudem findet sich für GraphQL bereits eine Vielzahl verschiedener Hilfs-Bibliotheken<sup>10</sup>. Einige dieser Bibliotheken fokussieren die Anpassung bestehender Anwendungen mit anderen Schnittstellentypen, andere unterstützen bei der Erkundung des Datengraphen eines Servers. Für die hier geplante Systemarchitektur sind jedoch speziell auf Microservices abgestimmte Bibliotheken hilfreich. Hier ist *Apollo Federation*<sup>11</sup> nutzbar, welches eine verteilte Erstellung des Datengraphen unterstützt. Implementiert ein Service diese Bibliothek, so kann er in seinem eigenen Daten-Schema auf die Typen anderer Services zugreifen, vorausgesetzt auch sie integrieren die Bibliothek. Hierdurch können Typen mit einzelnen Attributen angereichert, Enumerationen erweitert und weitere Sub-Typen für Interfaces oder Klassen erstellt werden. So ist es in dem geplanten System beispielsweise möglich, in einem zentralen Service grundlegende Typen für Ressourcen, Aktivitäten, Algorithmen oder Rollen zu definieren und diese dann über dynamisch ergänzte Services zu spezifizieren. Dies funktioniert jedoch nur in Zusammenhang mit der *Apollo Gateway* Bibliothek<sup>12</sup>, welche eine einfache Zusammenführung verschiedener *Federated-Services* in einem Gateway ermöglicht (siehe Anhang D.0.5). Die Erweiterung von Typen wird nur zwischen allen, mit demselben Gateway verbundenen Services ermöglicht. Trotz der Entkopplung von Services sind ihre Schnittstellen bzw. die dadurch bereitgestellten Datenstrukturen nun ähnlich einem Monolithen dynamisch verwaltbar.

## API Gateway

Bezüglich des API Gateways wurde bereits *Apollo Gateway* als Möglichkeit genannt. Es finden sich jedoch auch andere etablierte Alternativen, wie beispielsweise das *Tyk Open Source API Gateway*<sup>13</sup>, das *Amazon API Gateway*<sup>14</sup> oder das von Netflix entwickelte *Zuul*<sup>15</sup>, welche ebenfalls mit GraphQL arbeiten können. Dennoch bietet keines dieser

---

<sup>10</sup>Eine ausführliche Auflistung von Hilfs-Bibliotheken ist auf der GraphQL-Webseite gegeben: <https://graphql.org/code/>, zuletzt gesichtet 23.08.2021

<sup>11</sup>Apollo Federation: Siehe <https://www.apollographql.com/docs/federation/>, zuletzt gesichtet 23.08.2021

<sup>12</sup>Apollo Gateway: Siehe <https://www.apollographql.com/docs/federation/gateway/>, zuletzt gesichtet 23.08.2021

<sup>13</sup>Tyk API Gateway: Siehe <https://tyk.io/open-source-api-gateway/>, zuletzt gesichtet 23.08.2021

<sup>14</sup>Amazon API Gateway: Siehe <https://aws.amazon.com/api-gateway/>, zuletzt gesichtet 23.08.2021

<sup>15</sup>Zuul: Siehe <https://github.com/Netflix/zuul>, zuletzt gesichtet 23.08.2021

---

Gateways solch eine Abstimmung auf *Federated-Services* wie das *Apollo Gateway*, weshalb dieses hier zu präferieren ist. Ein Problem dieses Gateways ist jedoch, dass es zwar synchrone Operationen automatisiert kombiniert, asynchrone *Subscriptions* aber noch nicht kombinierbar sind. Dies kann mithilfe eines zweiten Services zur Ergänzung des Gateways umgangen werden. Die Integration der angereicherten Datenstrukturen ist dann aufwändig und es findet sich ein zweiter Einstiegspunkt, welcher einerseits verwaltet werden muss und andererseits die Logik der Clients komplexer gestaltet.

Ein weiteres Gateway, welches *Federated Services* interpretieren kann und auch *Subscriptions* kombiniert, ist *Wundergraph*<sup>16</sup>. Dieses interpretiert GraphQL-Queries nicht zur Laufzeit, sondern agiert als generischer Compiler, der für alle Services einheitlich virtuelle GraphQL-APIs erstellt und diese zu einem virtuellen Schema kombiniert. Dabei können auch REST-Services genutzt werden, deren Schnittstelle automatisch überführt wird, solange diese die *OpenAPI Specification (OAS)* befolgen. Wundergraph selbst kann von Clients über ein *JSON-RPC Protocol* angesprochen werden, wodurch eine sicherere Kommunikation mit einem geringeren Overhead, sowie eine höhere Performance erreicht wird. Auch wenn Clients selbstständig mit diesem Gateway kommunizieren können, so vereinfacht Wundergraph eine Nutzung durch die Bereitstellung eines Kommunikations-Clients, welcher auf Basis des virtuellen Schemas automatisiert erstellt wird und kontextabhängige Anfragen bereitstellt.

Auch *GraphQL Mesh*<sup>17</sup> ist als Gateway nutzbar. Dabei verfolgt es einen alternativen Weg zur Nutzung von GraphQL Federation in Form von *Schema Stitching*. Hierbei können mehrere GraphQL-Schnittstellen verlinkt, kombiniert, aggregiert, modifiziert und auch transformiert werden, wobei die Kombinerungslogik selbst bestimmbar ist. So sind beim Stitching auch zusätzliche Queries definierbar, in welchen bestehende Datentypen unterschiedlicher Schnittstellen genutzt und verbunden werden können, um so eine komplexere Logik einfacher zu ergänzen. Änderungen oder Ergänzungen eines Services erfordern jedoch ähnlich Wundergraph eine Anpassung, sowie einen Neustart des Gateways. Schnittstellen können, neben der Verbindung als Gateway, auch direkt in ein vollständig typisiertes SDK umgewandelt werden, wodurch einfachere Service-Zugriffe direkt im Code stattfinden können und eine Code-Autovervollständigung gegeben ist. Ähnlich Wundergraph ermöglicht auch GraphQL Mesh die Integration anderer Schnittstellen-Typen, bietet hier jedoch eine größere Vielfalt. So unterstützt es neben gängigen Formaten wie REST oder gRPC auch GraphQL Federation und unterschiedlichste Datenbanktypen. Diese Übersetzung ist auch unabhängig vom Gateway

---

<sup>16</sup>Wundergraph: Siehe <https://wundergraph.com/docs/introduction>, zuletzt gesichtet 23.08.2021

<sup>17</sup>GraphQL Mesh: Siehe <https://www.graphql-mesh.com/>, zuletzt gesichtet 23.08.2021

---

nützlich, da bestehende Schnittstellen automatisiert mit GraphQL-Adaptoren versehen werden können, um so eine einheitliche Verwendung der unterschiedlichen APIs als Ressourcen zu ermöglichen. GraphQL Mesh wird von *The Guild*<sup>18</sup> als *Open Source* Produkt gepflegt und weiterentwickelt. The Guild stellt auch viele weitere GraphQL-Bibliotheken bereit, sodass sich GraphQL Mesh in ein größeres Ökosystem eingliedert, welches die Entwicklung eigener Systeme vereinfacht und stark beschleunigt.

Sowohl *Apollo Gateway* als auch *Wundergraph* und *GraphQL Mesh* sind als Gateway der geplanten Anwendung geeignet. Sie weisen jedoch unterschiedliche Qualifikationen auf, aufgrund derer eine Entscheidung zu tätigen ist. Während *Wundergraph* mehr Funktionalitäten bereitstellt, performanter arbeitet, unterschiedliche Schnittstellentypen vereinen kann und somit eine höhere Technologiefreiheit der Services darstellt, ist es als Compiler anzusehen, sodass die Ergänzung weiterer Services nicht zur Laufzeit erfolgen kann. Ähnliche Vorteile bietet auch *GraphQL Mesh*, welches jedoch mehr Kontrolle über die Integration und Kombination von Services bietet. Zudem ist es flexibler einsetzbar (als SDK-System oder Gateway) und kann mehr Schnittstellentypen automatisiert in GraphQL überführen. Letztlich steht hinter *GraphQL Mesh* ein bekanntes Entwicklerteam, welches auch viele weitere Tools für GraphQL bereitstellt und jedes davon als Open Source Produkt anbietet. *Apollo Gateway* interpretiert Requests, im Gegensatz zu den zwei genannten Alternativen, zur Laufzeit, sodass neue Services keinen Neustart des Gateways benötigen. Zudem ist es von denselben Entwicklern wie *Apollo Federation*, sodass eine höhere Kompatibilität und schnellere Abstimmung auf Updates zu erwarten sind. Dabei ist jedoch eine höhere Latenz von Anfragen in Kauf zu nehmen, genauso wie eine geringere Kontrolle über die Kombination von Services und mehr Aufwand bezüglich asynchroner, externer Kommunikation.

Die Wahl ist somit von dem Einsatzgebiet der entstehenden Architektur abhängig. Wird sie für ein zentrales System eingesetzt, welches eine dynamische Ergänzung von Services auch externen Parteien ermöglichen soll, so ist die Nutzung des *Apollo Gateways* sinnvoll. Findet die Entwicklung jedoch in einem kleineren, statischen und fest definierten Rahmen statt, so ist *Wundergraph* oder *GraphQL Mesh* zu präferieren. Wird die Überführung bestehender Services in eine GraphQL-Schnittstelle benötigt und Wert auf ein einheitliches, jedoch weitreichendes Ökosystem von Softwareprodukten gelegt, so ist allgemein *GraphQL Mesh* das mächtigste Tool.

---

<sup>18</sup>The Guild: Siehe <https://the-guild.dev/>, zuletzt gesichtet 23.08.2021

## Service Discovery

Auch für die *Service Discovery* (siehe Anhang D.0.4) bietet Apollo eine Lösung, welche vollständig mit dem Gateway kombinierbar ist. Über *Apollo Managed Federation*<sup>19</sup> erhalten Entwickler Zugriff auf eine *Apollo Schema Registry*, bei welcher *Federated Services* registriert werden können. Diese Registry nimmt dann das Schema des Services entgegen, verifiziert dieses und fügt es in das komponierte *Federated Schema* der bereits registrierten Services ein. Dies hat den Vorteil, dass automatisch die Kompatibilität des Services überprüft wird, sodass auf der Ebene des Schemas keine Laufzeitfehler entstehen. Das Schema wird anschließend über einen *Apollo Uplink* Endpunkt bereitgestellt, mit welchem sich *Apollo Gateways* verbinden können, sodass ihre Schemata bei Änderungen stets aktualisiert werden. Ein weiterer Vorteil dieser Lösung ist, dass ein Schema in *Sub-Graphs* aufgeteilt und diese von unterschiedlichen Gateways aufgegriffen werden können. Diese können auch zur Laufzeit modifiziert werden. Dabei ist es auch möglich, Typen innerhalb eines Sub-Graphen zu definieren und sie dann innerhalb anderer Sub-Graphen zu nutzen, wodurch die Implementierung des Typen in einem isolierten Service vollzogen werden kann. Als Nachteil gilt jedoch, dass zur Nutzung dieser Software ein *Apollo Studio* Account benötigt wird. Das Hosting findet vollständig in der *Apollo Cloud* statt und kann nicht lokal erfolgen. Grundlegende Funktionen werden dabei bereits kostenlos angeboten, Erweiterungen wie die Verifizierung allerdings erst innerhalb eines Bezahlmodells. Darin befinden sich dann jedoch viele weitere Features, wie beispielsweise ein Monitoring der einzelnen Graphen, welche den Umgang mit einem Microservice-System erleichtern.

Wird hingegen eine Lösung benötigt, welche auf lokaler Infrastruktur aufgesetzt werden kann, so finden sich auch diesbezüglich etablierte Angebote. Neben der eigenen Implementierung solcher Funktionalitäten kann, abhängig von der gewählten Deployment- / Hosting-Infrastruktur, auf erprobte Software wie *Netflix Eureka*<sup>20</sup>, *Apache ZooKeeper*<sup>21</sup>, *HashiCorp Consul*<sup>22</sup> oder *Docker Registry*<sup>23</sup> zurückgegriffen werden. Im Vergleich zu *Apollo Studio* müssen hier jedoch sämtliche Integrationen und Verifikationen der GraphQL-Schemata mit viel Mehraufwand selbst implementiert werden, weshalb ersteres zu präferieren ist, vorausgesetzt die monatlichen Kosten sind in dem jeweiligen Kontext tragbar.

---

<sup>19</sup>Apollo Managed Federation: Siehe <https://www.apollographql.com/docs/federation/managed-federation/overview/>, zuletzt gesichtet 23.08.2021

<sup>20</sup>Netflix Eureka: Siehe <https://github.com/Netflix/eureka>, zuletzt gesichtet 23.08.2021

<sup>21</sup>Apache Zookeeper: Siehe <https://zookeeper.apache.org/>, zuletzt gesichtet 23.08.2021

<sup>22</sup>HashiCorp Consul: Siehe <https://www.consul.io/>, zuletzt gesichtet 23.08.2021

<sup>23</sup>Docker Registry: Siehe [https://hub.docker.com/\\_/registry](https://hub.docker.com/_/registry), zuletzt gesichtet 23.08.2021



## Interne Kommunikation

Letztlich ist die Technologie für interne Kommunikationen zu wählen. Wie in Anhang D.0.3 dargestellt, sind synchrone Kommunikationen zwischen Services nicht ausgeschlossen, welche ebenfalls über GraphQL-Schnittstellen stattfinden können. Dennoch sind vor allem lose gekoppelte Kommunikationswege zu präferieren. Auch die Anforderungen aus Abschnitt 4.1 unterstreichen die Nutzung solcher Kommunikationswege, da für die Implementierung der *Algorithmen* ein eventbasiertes Messaging notwendig ist. Als Message Broker hierfür können wiederum einige etablierte Produkte genutzt werden, welche unterschiedliche Anforderungen erfüllen. Ist die Persistenz von Nachrichten nicht notwendig, sind verlorene Nachrichten zu verkraften und ist die Performanz am wichtigsten, so ist der in-memory Message Broker *Redis*<sup>24</sup> ein optimales Produkt. Wird hingegen ein Message Broker benötigt, der auch ein komplexes Routing mit geeigneten Funktionen unterstützt, so kann *RabbitMQ*<sup>25</sup> gewählt werden. Wird letztlich ein Alleskönner benötigt, welcher sich besonders für die lange Speicherung vieler Nachrichten eignet, die Verwendung von *One-To-Many Channels* vereinfacht und allgemein einen hohen Durchsatz aufweist, so ist die Open-Source Software *Apache Kafka*<sup>26</sup> eine geeignete Wahl. Da in dem geplanten System eine hohe Flexibilität angestrebt wird, dynamisch neue Services hinzugefügt werden sollen, Services von unterschiedlichen Hosts mit ebenso unterschiedlichen Wartungszeiten und Ausfällen eingebunden werden könnten, sowie eine Vielzahl verschiedener Services Nachrichten veröffentlichen kann, erscheint *Apache Kafka* in diesem Kontext als sinnvollster Message Broker.

### 4.3. Architektur des Systems

Abschließend verbleibt nun die Darstellung der geplanten Systemarchitektur, welche das Domänen-Modell (siehe Abschnitt 4.1), den gewünschten Anwendungstypen und den angestrebten Architekturstil (siehe Abschnitt 4.2) vereint. Hierfür wurde zunächst auf Basis der Vor- und Nachteile unterschiedlicher Technologien folgender Technologiestack ausgewählt:

- Für die externe Kommunikation wurde GraphQL als geeignete Schnittstellen-Spezifikation identifiziert.
- Die Verwendung des gesamten Apollo-Ökosystems ist aufgrund der Flexibilität, bereitgestellten Funktionalität und erprobten Nutzung durch große Unternehmen vorgesehen. So werden die Schnittstellen der einzelnen Services mithilfe von

<sup>24</sup>Redis: Siehe <https://redis.io/topics/pubsub>, zuletzt gesichtet 23.08.2021

<sup>25</sup>RabbitMQ: Siehe <https://www.rabbitmq.com/>, zuletzt gesichtet 23.08.2021

<sup>26</sup>Apache Kafka: Siehe <https://kafka.apache.org/>, zuletzt gesichtet 23.08.2021

Apollo Federation definiert, durch Apollo Studio bzw. der davon bereitgestellten Service Registry auf Kompatibilität geprüft und dort zu einem Schema zusammengesetzt. Besonders die Nutzung des Apollo Gateways gilt als sinnvoll, da Änderungen eines Services dynamisch erfolgen können, ohne dass dieses neugestartet werden muss.

- Als interner Kommunikationsweg wurde sich für ein asynchrones Messaging mithilfe von Apache Kafka entschieden. Über One-To-One-Channel kommunizieren Services dabei gezielt miteinander, während One-To-Many-Channel für die Übermittlung sämtlicher Events genutzt werden. Dies ist besonders bei der Synchronisierung von Datenbeständen mithilfe *orchestrierter Sagas* sinnvoll (siehe Anhang D.0.2), welche die lose Kopplung aufrecht erhalten.
- Bietet ein Service ein Micro Frontend an, so wird dieses über eine globale Create-Funktion bereitgestellt, welche dann als Script-Element eingebunden und ausgeführt werden kann (siehe Anhang D.0.6). Diese sind dann in einem zentralen Frontend renderbar, wodurch ihnen eine lokale Interaktion sowie ein Zugriff auf denselben Kontext bzw. dieselben Daten ermöglicht wird.

Ist die Nutzung dieses Software-Stacks in einem Kontext nicht möglich, so kann stets auf Alternativen zurückgegriffen werden. Hierdurch sind dann jedoch möglicherweise Anpassungen der Architektur notwendig. Die folgenden Unterabschnitte befassen sich nun mit der Erläuterung des geplanten Systems, welches etappenweise vorgestellt wird. Aufbauend auf Architekturdiagrammen liegt dabei der Fokus besonders auf den Aufgabenbereichen und Kommunikationswegen jedes Bestandteils. Die Grundaufgaben bekannter Microservice-Strukturen werden dabei nicht gesondert aufgelistet, da diese in Anhang D dargestellt sind.

#### 4.3.1. Zentrales Grundsystem

Abbildung 4.2 präsentiert das *Base System*, in welchem alle zentralen Bestandteile des hybriden Systems aufzufinden sind. Hierzu zählen zunächst die *Basic Services*. Diese stellen die benötigte Infrastruktur der grundlegenden Komponenten bereit (siehe Abschnitt 4.1) und hierdurch die Kernfunktionalität des Systems. So findet sich für die Erstellung und Verwaltung von Nutzern, Geräten, Communities, Szenarien, Sessions und Räumen ein dedizierter Service. Der aufgeführte *Template-Service* ist für die Erstellung und Verwaltung der meisten komplexen Datenstrukturen zuständig, welche von Nutzern des Systems dynamisch konstruierbar sind und deren Zusammensetzung für spätere Wiederverwendung gespeichert werden sollte. Dazu gehören sowohl Vorlagen komplexer und verschachtelter Prozesse, als auch solche von komplexen Räumen

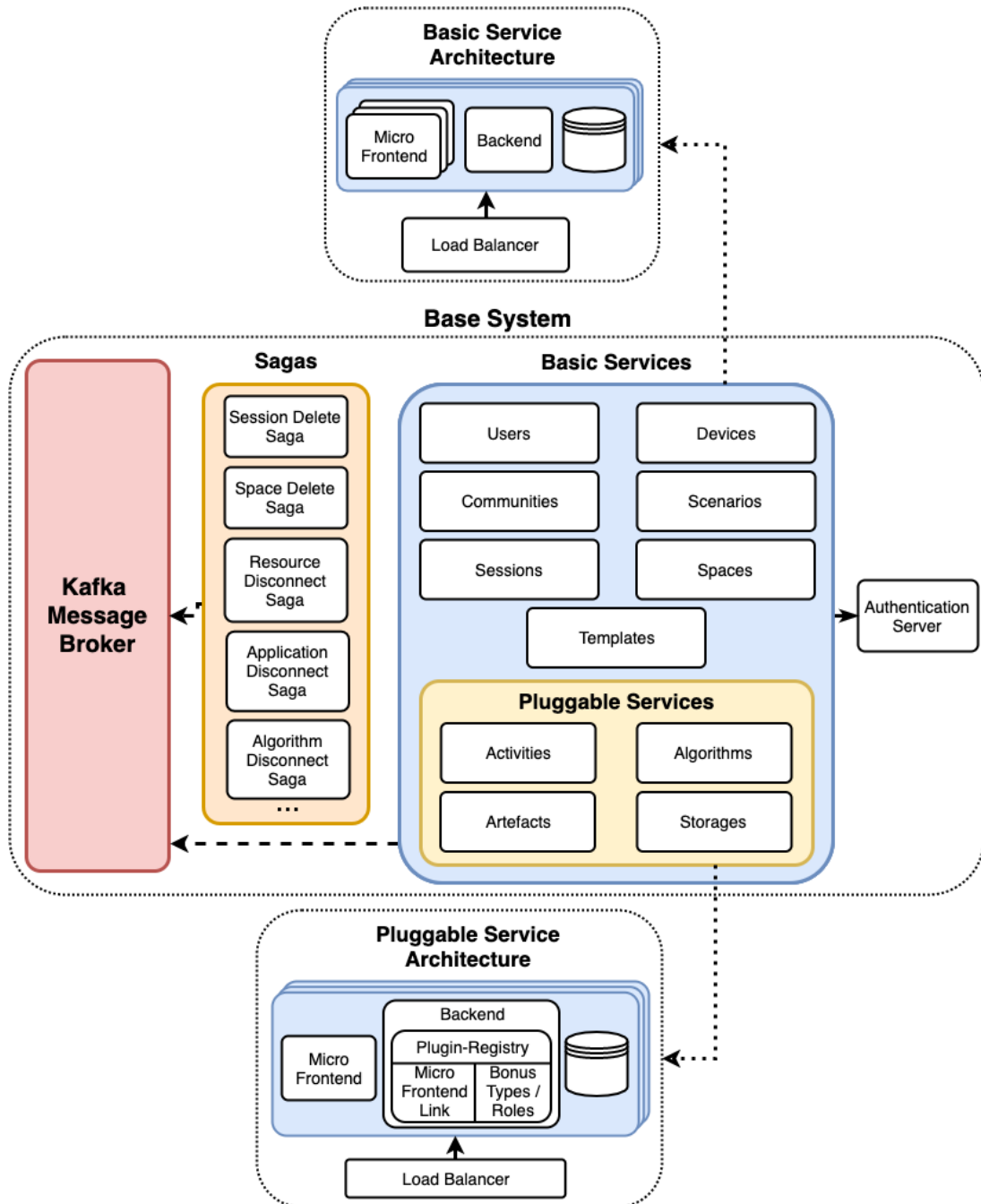


Abbildung 4.2.: Architektur des Basis-Systems

und Aktivitäten (siehe Abschnitte 4.1.2 und 4.1.4). So kann es beispielsweise vorkommen, dass ein Lehrender drei Sessions generiert. In der ersten Session bereitet er einen Vorlesungsraum vor, in welchem beispielsweise Aktivitäten für einen Video-Stream, ein Feedback der Lernenden und die Präsentation der Vorlesungsmaterialien platziert wer-

den. Die zweite Session besteht aus mehreren identischen Gruppenarbeitsräumen und in der letzten Session befindet sich ein gemeinsamer Raum für die Zusammenführung der Gruppenergebnisse. Die Aktivitätskombination des ersten Vorlesungsraums könnte auch innerhalb der Gruppenräume notwendig sein, sodass die Erstellung eines Templates sinnig wäre. Ebenso ist bei der Erstellung mehrerer identischer Gruppenräume die Erstellung einer Vorlage und ihre Wiederverwendung nützlich. Letztlich stellt auch die gesamte Konstellation der Sessions ein gängiges Schema dar, sodass auch dieses als Template hinterlegt und für weitere Veranstaltungen genutzt werden kann.

Jeder der genannten *Basic Services* stellt zur Erhaltung der losen Kopplung eine einheitliche Infrastruktur bereit. Um größere Lasten durch eine horizontale Skalierung kompensieren zu können, ist der Einstiegspunkt stets ein eigener Load Balancer, welcher ebenso als *Circuit Breaker* agiert (siehe Anhang D.0.3). Dieser verteilt die Anfragen an die zugehörigen Service-Instanzen, sodass eine Unterscheidung nicht außerhalb des Services geschehen muss. Jede Service-Instanz besteht weiterhin aus drei Komponenten. Sie besitzt eine eigene Datenbank, ein Backend für die jeweilige Logik und mindestens ein Micro Frontend. Die Anzahl der Micro Frontends ist dabei abhängig von der Anzahl darzustellender Zuständigkeiten. So muss der User-Service unter anderem ein Micro Frontend für die Erstellung, eines für die Verwaltung und eines für die Auflistung von Nutzern bieten. Diese können dann angefragt und in einem gemeinsamen Frontend unabhängig eingebunden werden. Finden sich für einen Service zu viele Zuständigkeitsbereiche, so kann dieser wiederum in mehrere Services aufgeteilt werden. So kann beispielsweise die Logik des Template-Services auf drei Services verteilt werden, einen für Prozesse, einen für Räume und einen für Aktivitäten. Zudem kann für diese Services evaluiert werden, ob ihre Micro Frontends vollständig in einem zentralen Frontend zu implementieren sind, da die Services eine solide Grundstruktur bereitstellen und somit wenigen Änderungen unterliegen sollten. Sind jedoch genügend Kapazitäten und Arbeitskräfte vorhanden, so sollte stets die Auslagerung der Frontends bevorzugt werden, da sie eine höhere Flexibilität darstellt.

Zu den Basiskomponenten gehört zudem eine Unterkategorie von Services, welche eine variierende und stärker vorgegebene Infrastruktur des Backends aufweisen. Die Kategorie der *Pluggable Services* umspannt alle Komponenten des Systems, welche dynamisch erweitert werden müssen. Dazu zählen Aktivitäten, Algorithmen und Ressourcen. Für die Erweiterbarkeit des Systems wurde ein *Plugin-Verfahren* ausgewählt, welches in Abschnitt 4.3.3 genauer erläutert wird. Für dieses Plugin-Verfahren stellen die Pluggable-Services Verwaltungsfunktionen bereit, über die eine Registrierung neuer Plugins des jeweiligen Typen möglich ist. Ebenso findet sich jeweils eine Route, über welche

Plugins einen *Heartbeat* vollziehen können (siehe Anhang D.0.4), um so ihre Lauffähigkeit in regelmäßigen Abständen mitzuteilen. Neben der Verwaltung von Plugins sind Pluggable Services bei Anfragen eines speziellen Typen oder Micro Frontends für die Weiterleitung an den zugehörigen Plugin-Service zuständig. Wird beispielsweise eine Aktivität als Plugin registriert und in einem Raum referenziert, so muss bei dessen Render-Prozess zunächst der *Activities*-Service angefragt werden, welcher als Router nun die Anfrage dem passenden Service übermittelt bzw. die passende Micro Frontend URL dem Anfragenden mitteilt. Bei der Registrierung werden jedoch nicht nur URLs übermittelt, sondern auch Metainformationen über den Service. Dazu gehören unter anderem dessen Name bzw. Typ, zusätzliche Typen- oder Service-Beschreibungen, ein Preview-Bild, sowie von ihm interpretierbare Rollen. Letzteres ist besonders wichtig, da eine Autorisierung der Zugriffe jedem Service selbst überlassen wird und Rechte vor allem über Rollen definiert sind. So erhält jedes Plugin die Möglichkeit, eigene Rollen zu definieren und diese mit dem System zu synchronisieren. Neben dieser Plugin-Logik, welche in einem Backend des jeweiligen Services vollzogen wird, müssen Pluggable Services wiederum eine eigene Datenbank und ein Micro Frontend aufweisen. Da die meisten Anfragen weitergeleitet werden, ist das Micro Frontend vor allem für die Auflistung bzw. Präsentation vorhandener Plugins des jeweiligen Typen zuständig. Damit mehrere Instanzen parallel genutzt werden können, ist auch hier wieder ein eigenes Load Balancing zu vollziehen.

Als weiteres Kernelement zählt die Infrastruktur für asynchrone, interne Kommunikationen, welche hier in Form des *Kafka Message Brokers* dargestellt ist. Über diesen werden sämtlichen Kanäle angelegt, welche für eine lose gekoppelte Kommunikation notwendig sind. Die Übermittlung von Events kann beispielsweise in Kanälen für gesamte Prozesse, Raumkonstellationen oder einzelne Aktivitäten stattfinden. Ebenso kann für grundlegende Services ein jeweils dedizierter Kanal erstellt werden, welcher dann dem Empfang von Command-Messages dient. Solche Kanäle sind besonders in Zusammenhang mit *Sagas* wichtig (siehe Anhang D.0.2), welche auch hier der Synchronisation von Datenbeständen dienen. So finden sich in diesem System beispielsweise *Sagas* für Erstellungen, Löschungen und Modifikationen von Nutzern, Geräten, Prozessen oder Räumen. Ebenso müssen *Sagas* für die Verbindung eines Plugins oder dessen Verbindungsabbruch vorhanden sein. Letztlich ist als Bestandteil des Base Systems der *Authentication Server* zu nennen. Auch in dieser Microservice-Architektur ist die Verwendung von *Access Tokens* vorgesehen (siehe Anhang D.0.7), welche zentral von diesem Server verwaltet werden. Services benötigen Zugriff auf diesen, damit sie die erhaltenen Tokens validieren können. Vor allem der *Users*-Service kommuniziert häufig mit dem Authentication Server, da er ihm registrierte oder gelöschte Nutzer mitteilen

muss. Hier wurde eine synchrone Kommunikation gewählt, da der Authentication Server eine zentrale Instanz darstellt, ohne die das System nicht genutzt werden könnte. Ist dieser offline, wird keine Anfrage gelingen, sodass eine synchrone Kommunikation hier keine negativen Auswirkungen hat. Zudem ist die Software für diesen Server nicht festgelegt. Wird dabei das Produkt eines bestehenden Anbieters verwendet, so kann dieser von einem anderen Host bereitgestellt werden. Kritische Infrastrukturen wie der zentrale Message Broker sollten jedoch nicht öffentlich sichtbar sein, weshalb Kommunikationen über eine API zu präferieren sind.

### 4.3.2. Bereitstellung der Services

Nicht nur gegenüber einem externen Authentication Server sondern auch gegenüber den Clients sollten interne Implementierungsdetails verborgen bleiben. So zeigt Abbildung 4.3, dass hier keine *Client-Side Discovery* der Services angestrebt wird, sondern eine *Server-Side Discovery* innerhalb von API Gateways (siehe Anhänge D.0.4 und D.0.5).

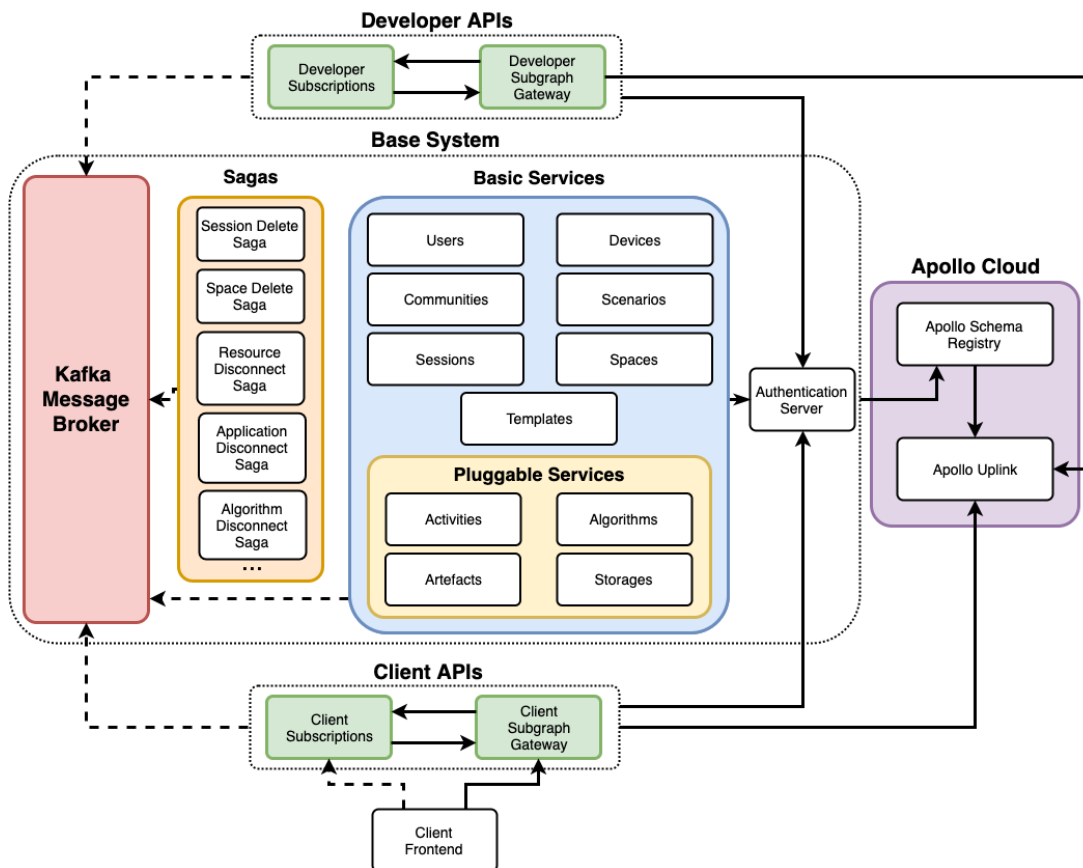


Abbildung 4.3.: Bereitstellung der Services über Gateways

Die dafür benötigte *Service Registry* wird durch *Apollo Studio* bzw. die *Apollo Cloud* bereitgestellt. Wird ein Service gestartet, so registriert sich dieser selbstständig bei der Service Registry. Da pro Service ein Load Balancer eingeplant ist und dessen Ausfall ebenso den Verlust damit verbundener Service-Instanzen darstellt, ist es die Aufgabe des Load Balancers, das Schema des Services anzufragen und sich mit diesem an der Service Registry anzumelden. Hierdurch werden auch Schema-Dopplungen innerhalb der Registry vermieden. Das resultierende Schema wird von der Registry an den *Apollo Uplink* übergeben und dient von dort aus der Aktualisierung zweier Gateways.

Das erste Gateway ist für die Verwendung durch Clients bzw. ein einheitliches Frontend gedacht. Es greift auf einen Subgraph des Schemas zu, welcher sämtliche Typen und Operationen für eine Darstellung beinhaltet. So ist bereits bei der Registrierung eines Services darauf zu achten, welche Teile für den Client notwendig sind. Damit dieser auch auf asynchrone Events reagieren kann, erhält er zusätzlich den Zugriff auf einen *Subscription Server*, welcher notwendige Events des Message Brokers abfängt und über GraphQL-Subscriptions bereitstellt. Da externe Instanzen so keinen direkten Zugriff auf kritische Systembestandteile erhalten, werden Implementierungsdetails versteckt, eine stärkere Entkopplung erreicht und eine höhere Sicherheit geboten. Bei dem Subscription Server ist darauf zu achten, dass dieser dasselbe Typensystem anbietet bzw. nutzt wie das zugehörige Gateway. Da sich das Gateway stets aktualisieren kann, muss es nach einer Aktualisierung den Subscription Server über Neuerungen informieren.

Ähnlich handelt auch das zweite Gateway, welches speziell zur Implementierung und Integration von Plugins genutzt wird. So stellt das *Developer Gateway* alle Operationen des Developer Subgraphs bereit, in welchem beispielsweise Funktionen zur Registrierung einer Aktivität, einer Ressource oder eines Algorithmus aufzufinden sind. Auch können hierüber weitere Operationen bereitgestellt werden, welche einen detaillierteren Zugriff auf grundlegende Daten ermöglichen. In solch einem Falle sollte über die Verwendung eines *API Keys* nachgedacht werden, welcher nur an vertrauenswürdige Entwickler ausgehändigt wird. Ähnlich dem Gateway ist hier ein Subscription Server speziell auf die Plugin-Entwicklung ausgelegt. Dieses nimmt besonders bei der Entwicklung von Algorithmen eine entscheidende Rolle ein. Im Gegensatz zu dem Subscription Server des Clients bietet dieses Pendant für Entwickler weitere Funktionen, über welche Events nicht nur konsumiert, sondern auch erstellt werden können. Hierüber ist es ihnen möglich, Grundprozesse anzustoßen und miteinander zu kommunizieren, was besonders bei der Implementierung von Algorithmen entscheidend sein kann.

Abbildung 4.3 zeigt weiterhin, dass beide Gateways auch einen Zugriff auf den Authentication Server erhalten, damit Authentifizierungen bereits in diesen zentralen Knotenpunkten vollzogen werden können. Zudem ist neben der Authentifizierung und Weiterleitung von Anfragen auch das *Rate Limiting* ein Aufgabenfeld der Gateways.

### 4.3.3. Einbindung von Plugins

Aufbauend auf den *Developer APIs* und den *Pluggable Services* kann nun ein *Extended System* geschaffen werden, welches dynamisch neue Funktionalitäten bereitstellt. Sämtliche Algorithmen, Ressourcen und Aktivitäten werden über diesen Mechanismus eingebunden, egal ob diese von externen Instanzen oder dem zentralen Entwicklungsteam erstellt wurden. So sind bereits grundlegende Artefakt-Typen und Speicher, sowie Algorithmen und Aktivitäten als Plugin zu integrieren, auch wenn diese dieselbe Hosting-Infrastruktur nutzen. Dies hat den Vorteil, dass die Integration einheitlich erfolgt und bereits vor der ersten Veröffentlichung weitreichend erprobt ist. Die Eingliederung des Extended Systems wird in Abbildung 4.4 veranschaulicht.

Die Infrastruktur jedes Plugins gleicht dabei den Basic Services, sodass es einen Load Balancer bereitstellen sollte und jede Instanz innerhalb eines Backends neben den Zugriffen auf das Kernsystem auch zusätzliche Logik implementieren kann. Eigene Datenstrukturen sollten innerhalb eigener Datenbanken abgelegt werden und jedes Plugin bietet ein dediziertes Micro Frontend an, welches auf die Nutzung und Darstellung der zusätzlichen Logik bzw. des jeweiligen Plugin-Typen abgestimmt ist. Da dieses Micro Frontend über das gemeinsame Client Frontend aufgerufen wird, erhält es Zugriff auf einen gemeinsamen Kontext und auch auf die Client APIs, sodass Interaktionen zwischen Services und Plugins möglich sind. Da sich Plugins beim Kernsystem registrieren und dabei sowohl die Verbindung zum Micro Frontend, als auch zusätzliche Typen und Rollen übermitteln können, sind sie auch innerhalb allgemeiner Verwaltungen und Darstellungen nutzbar. So sind Activity-Plugins beispielsweise auch in einem Template speicherbar und Artefakte in allen Speicherorten ablegbar. Bei der Erstellung neuer Services können Plugins jedoch auch aufeinander abgestimmt werden. Wird eine Aktivität für Brainstorming-Prozesse erstellt, so kann das Endprodukt bereits über vorhandene Artefakttypen wie JPEG-Bilder gespeichert werden. Ist jedoch die Speicherung eines Zwischenstands notwendig, welcher zu einem späteren Zeitpunkt fortgeführt werden soll, so sind mehr Metainformationen notwendig als eine Bild-Datei darstellen kann. Hierfür ist dann gesondert ein neuer Artefakttyp zu implementieren, welcher auf die Aktivität optimiert ist, sowie von dieser erstellt, interpretiert und bearbeitet werden kann. Da Artefakte jedoch auch in anderen Kontexten geöffnet werden könnten,



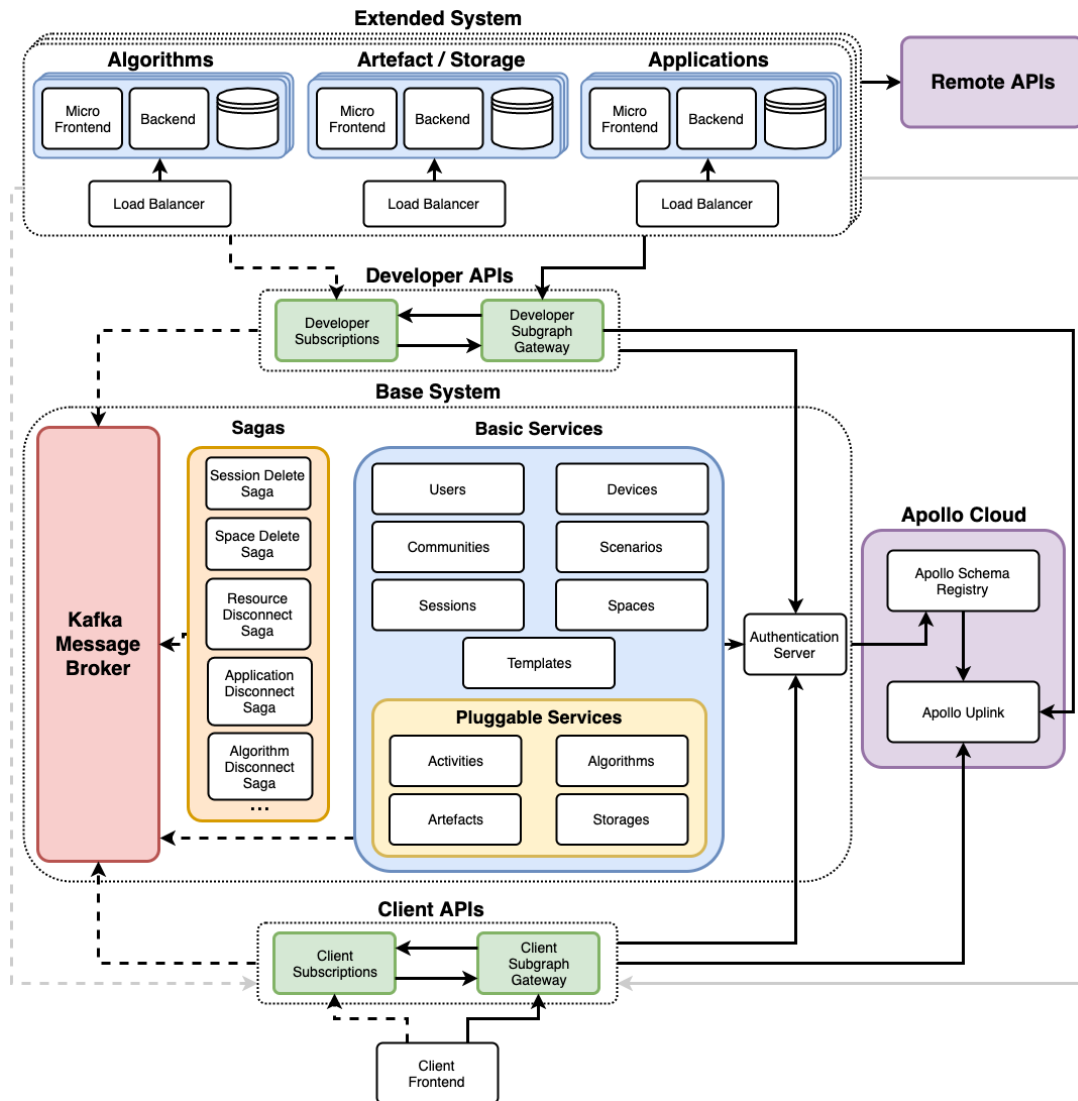


Abbildung 4.4.: Dynamische Microservice-Architektur für hybride Lernraumsysteme

beispielsweise als Ressource innerhalb einer Veranstaltung, müssen auch diese unabhängig von der Aktivität ein eigenes Micro Frontend anbieten, welches die beinhalteten Daten sinngemäß darstellt. So sind sie auch an diversen Speicherorten ablegbar, welche lediglich die Darstellung der Artefaktübersicht bzw. deren Auflistung implementieren müssen und für eine Anzeige einzelner Artefakte auf das vorhandene Micro Frontend des jeweiligen Artefakttypen zugreifen können.

Zur Verdeutlichung der Plugin-Integration sind in Abbildung 4.5 beispielhaft alle notwendigen Kommunikationssequenzen für die Registrierung eines Artefakts aufgeführt.

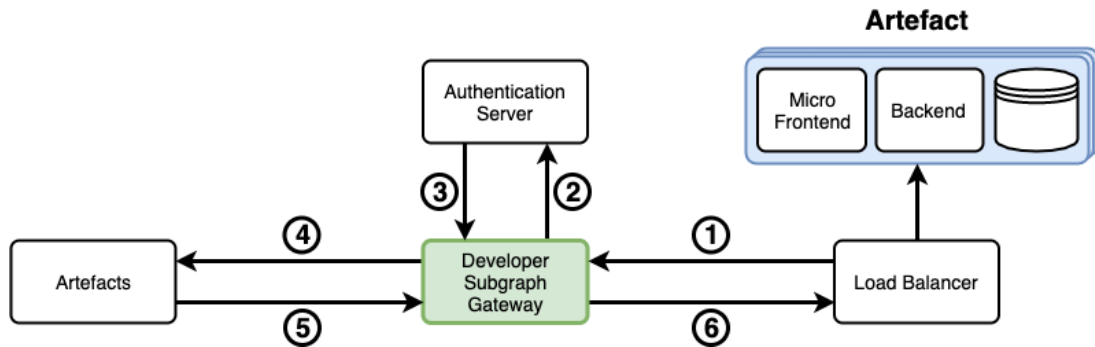


Abbildung 4.5.: Registrierung eines Artefakt-Plugins

Die Kommunikation findet also in folgender Reihenfolge statt:

1. Der Load Balancer des Artefakt-Plugins nutzt die vom Gateway bereitgestellte Operation zur Selbstregistrierung. Dabei übermittelt er einerseits den API Key zur Verifizierung des Entwicklers und andererseits die URL des Micro Frontends, sowie den Namen des Artefakts bzw. dessen Typen.
2. Das Developer Gateway kontaktiert den Authentication Server, um den API Key zu verifizieren.
3. Der Authentication Server antwortet entweder mit einem JWT zur Identifikation des Entwicklers, oder mit einer Fehlermeldung, falls der API Key ungültig ist.
4. Ist ein Fehler bei der Authentifizierung aufgetreten, wird vom Gateway ebenso eine Fehlermeldung an den Service bzw. Load Balancer gesendet und der Vorgang bricht ab. Andernfalls leitet das Gateway die Anfrage mitsamt JWT an den Artefacts Service weiter, welcher nun die Registrierung vollziehen soll.
5. Der Artefacts Service autorisiert das JWT. Ist keine Berechtigung vorhanden, so wird der Aufruf abgebrochen und ein Fehler gesendet. Ist die Autorisierung jedoch erfolgreich, so speichert der Service die Daten des neuen Plugins und sendet eine Erfolgsmeldung.
6. Das Gateway übersendet dem anfragenden Service entweder die Erfolgsmeldung oder den Fehler.

Auch die Nutzung eines Plugins durch Clients wird folgend in Abbildung 4.6 dargestellt. In diesem Falle möchte das Client Frontend einen Lernraum rendern, welcher drei Aktivitäten und ein Artefakt beinhaltet. Dies könnte beispielsweise ein Vorlesungsraum mit einem Livestream, einem Text-Chat, einem Dokumentations-Tool für Notizen und der Anzeige des Skripts sein.

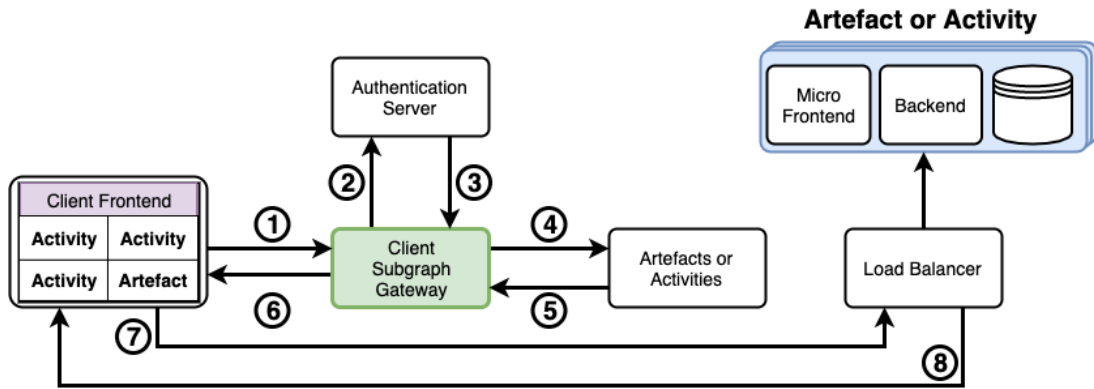


Abbildung 4.6.: Rendering eines Artefakts

Für das Rendering der Aktivitäten und des Artefakts wird dann die folgende Kommunikations-Sequenz benötigt:

1. Um die Aktivitäten oder das Artefakt zu rendern startet das Client Frontend zunächst einen Aufruf an das Gateway. Dabei wird entweder ein bereits vorhandenes Session Token oder die Login Daten des Nutzers übersendet. Ebenso gibt das Frontend an, welche Aktivität oder welches Artefakt angezeigt werden soll.
2. Das Developer Gateway kontaktiert den Authentication Server, um die Login Daten zu verifizieren. Ist bereits ein valides Session Token vorhanden, dessen Ablaufdatum noch nicht überschritten wurde, wird dieser und der nächste Schritt übersprungen. Ist das Ablaufdatum des Tokens erreicht, wird in diesem Schritt ein neues Token generiert.
3. Der Authentication Server antwortet entweder mit einem JWT zur Identifikation des Nutzers, oder mit einer Fehlermeldung, falls die Login Daten ungültig sind.
4. Ist ein Fehler bei der Authentifizierung aufgetreten, wird vom Gateway ebenso eine Fehlermeldung an den Nutzer gesendet und der Vorgang bricht ab. Andernfalls leitet das Gateway die Anfrage mitsamt JWT an den Activities oder Artefacts Service weiter.
5. Der angefragte Service autorisiert das JWT. Ist keine Berechtigung vorhanden, so wird der Aufruf abgebrochen und ein Fehler gesendet. Ist die Autorisierung jedoch erfolgreich, so ermittelt der Service den Typen des Artefakts oder der Aktivität und antwortet mit der zugehörigen Micro Frontend URL.
6. Das Gateway übersendet dem anfragenden Service entweder die URL oder den Fehler.

7. Im Erfolgsfall erstellt der Client ein Script-Element mithilfe der URL, wodurch der Service bzw. dessen Micro Frontend importiert wird.
8. Nach dem Import des Frontends als Bibliothek kann dessen globale Funktion aufgerufen werden, wodurch das Frontend in einem definierten Bereich gerendert wird. Die Daten des Artefakts können dabei entweder als Parameter übergeben oder von der Render-Funktion über den gemeinsamen Kontext verwendet werden.

Als letzter Punkt der Plugin-Einbindung sind externe APIs zu nennen. Da jedes Plugin eine eigene Logik implementieren kann, ist es in diesen auch möglich, auf externe APIs, Anwendungen und Ressourcen zuzugreifen. Dadurch ist die Realisierung eines Adapters denkbar, welcher externe Daten auf interne Strukturen angleicht und somit eine Interkompatibilität schafft (siehe Abschnitt 4.1.4). So kann beispielsweise die Integration des campuseigenen LMS erfolgen, dessen Daten und Funktionen mithilfe dedizierter Speicher und Aktivitäten überführt werden. Auf eine ähnliche Weise kann eine Youtube-Anbindung stattfinden, welche neben dem Speicher auch einen Artefakttypen zur Darstellung solcher Videolinks bereitstellt. In einer zusätzlichen Aktivität können dann Notiz-Funktionen bereitgestellt und diese mit einem Video bzw. dessen *Time Stamps* synchronisiert werden.

Solch ein Plugin-Mechanismus komplementiert also die Architektur, indem es eine hohe Flexibilität der Anbindung neuer Funktionalitäten, sowie eine hohe Kompatibilität unterschiedlicher Konzepte und Komponenten bietet, wodurch ein erweiterbares Ökosystem mit mannigfaltigen Anwendungsgebieten und Interaktionsmöglichkeiten entstehen kann.

## 5. Fazit & Ausblick

In diesem abschließenden Kapitel findet nun eine rückwirkende Betrachtung und kritische Bewertung der Forschungsergebnisse statt, sowie ihre Einordnung in den aktuellen Forschungskontext. Weiterhin werden Anschlusspunkte für folgende Projekte genannt, welche einer Verwertung, Verbesserung und Erweiterung der vorgestellten Konzepte sowie erarbeiteten Modelle dienen können.

### 5.1. Fazit

Die vorliegende Arbeit befasste sich mit dem hybriden Lehrparadigma und der damit verbundenen Konzeption hybrider Lernräume. Der besondere Fokus lag dabei auf einer technischen Perspektive, sodass als zentrales Ergebnis die Erstellung einer dynamischen Systemarchitektur für hybride Ökosysteme angestrebt wurde.

Da trotz der Relevanz des Themenfeldes noch kein einheitliches Verständnis in aktuellen Forschungsergebnissen ersichtlich war, musste zunächst eine ausführliche Literaturrecherche vollzogen werden. Darauf aufbauend konnten detaillierte Begriffsdefinitionen hybrider Lehre und Lernräume entstehen, in welchen auch verwandte Begriffe wie das Blended Learning oder Seamless Learning Beachtung fanden. So wurde ersichtlich, dass bedingt durch den rapiden technischen Fortschritt noch keine begründete und eindeutige Trennung der Begriffe bzw. ihrer Bedeutungen möglich ist. Die Recherche führte zudem zu einer Identifikation der wichtigsten Bestandteile und Charakteristiken hybrider Lehre. Identifizierte Kernbestandteile wurden hierbei in Übersichtsmodellen zusammengefasst und mitsamt ihrer wichtigsten Beziehungen beschrieben. Ebenso fand eine Erläuterung der beschreibenden Charakteristiken statt, welche als Dimensionen festgehalten und mit jeweiligen Unterdimensionen sowie Ausprägungen aufgelistet wurden. Diese Konzepte unterstützen dabei, ein tieferes Verständnis des Themenfeldes aufzubauen und können demnach bei der Planung einer hybriden Veranstaltung wichtige Hilfestellungen bieten.

Aufgrund der Vielschichtigkeit hybrider Lehre war die Bereitstellung dieser Artefakte für einen strukturierten Designprozess jedoch nicht ausreichend. So stellten sich

hybride Dimensionen aufgrund der großen Varietät beinhalteter Unterdimensionen, Ausprägungen und Kombinationsmöglichkeiten als eine zu komplexe Basis heraus. Als alternativer Ausgangspunkt eines Designprozesses wurden deshalb die zur Aufgabenbewältigung notwendigen Lernaktivitäten verwendet, welche eine einheitlichere und strukturierte Erkundung der Anforderungen eines Lehr- / Lernkontextes ermöglichen. Auf Basis der Forschungsergebnisse von Goodyear and Ellis [2008], einer Erweiterung der Aktivitätstypen von Conole [2007] und zusätzlicher Recherchen des didaktischen Instruction Designs, wurde anschließend ein Framework zur Planung hybrider Systeme geschaffen und schrittweise beschrieben. Dieses schlägt vor, aufbauend auf dem vorliegenden Lehr- / Lernszenario zunächst die zur Aufgabenbewältigung notwendigen Lernaktivitäten zu ermitteln. So festgehaltene Aktivitäten können dann der Identifikation des benötigten digitalen Ökosystems bzw. des in digitalen Räumen bereitzustellenden Social- und Set-Designs dienen, sowie weitere technische Komponenten ersichtlich machen. Wichtige Faktoren bei der Auswahl stellen zudem die physischen Räume mitsamt ihrer Infrastruktur, dem sie umschließenden Setting und den beinhalteten Communities dar, sowie die privaten Medienökologien potentieller Nutzer.

Zur Beantwortung der zentralen Forschungsfrage wurde abschließend eine Systemarchitektur konzipiert, welche in gemeinsamer Nutzung mit dem Design-Framework eine zielgerichtete, flexible und schnelle Entwicklung hybrider Systeme bzw. hybrider Lernräume ermöglicht. Dabei wurde zunächst aufbauend auf den Erkenntnissen vorangegangener Kapitel ein übergreifendes Domänenmodell erstellt, welches die Zuständigkeitsbereiche und Zusammenhänge der wichtigsten Systemkomponenten aus einer technischen Perspektive darstellt. Mithilfe von Best-Practices des Microservice-Ansatzes und einer auf GraphQL ausgerichteten Technologiewahl fand dann die Konzeption der modularen und erweiterbaren Webanwendung statt. In dieser wurde zusätzlich zu einem zentralen Basis-System auch besonders die dynamische Integration neuer Plugin-Services bedacht, durch welche das Gesamtsystem flexibel auf neue oder sich ändernde Bedürfnisse der Nutzer reagieren und mit der Zeit zu einem umfassenden Ökosystem heranwachsen kann.

Die genannten Ergebnisse stellen nun sowohl für die Forschungsgemeinde, als auch für die Praxis eine Bereicherung dar. So sind für weitere Erforschungen hybrider Lehre und Lernräume besonders die Definitionen, Entitäten und Dimensionen interessant. Bestehende Ansätze und Definitionen des Paradigmas weisen eine große Varietät auf, sodass darauf aufbauende Forschungsergebnisse isoliert in dem jeweiligen Kontext zu betrachten sind. Die hier ermittelten Begriffsdefinitionen und Übersichtsmodelle der Entitäten sowie Dimensionen basieren auf einer umfassenden Literaturrecherche und

vereinen dadurch übergreifend die wichtigsten Konzepte unterschiedlicher Disziplinen. Somit ist eine Grundlage für folgende und interdisziplinäre Forschungsprozesse geschaffen worden, welche eine systematische Erforschung des großen Themenfeldes in kleinen Schritten und aus unterschiedlichen Perspektiven ermöglicht. Die unterschiedlichen Konzepte wurden jedoch nicht nur aufgelistet, sondern auch deren Bedeutungen analysiert, Überschneidungen darin ermittelt und interdisziplinäre Beziehungen erarbeitet. Dadurch erhofft sich der Autor, die Verwirrung durch konkurrierende Ansichten zu beseitigen, neue Denkanstöße zu schaffen und einen Austausch der Forschenden unterschiedlicher Disziplinen zu unterstützen.

In der Darstellung angestrebter Ziele dieser Arbeit wurden jedoch besonders praktische Überführungen der Theorie angesprochen, sodass vor allem die Praxis von den Ergebnissen profitiert. In einem stringenten Forschungsprozess wurden abstrakte und weitreichende Begriffe analysiert, in Beziehung gesetzt und mit Beispielen angereichert, wodurch diese theoretischen Konzepte für eine praktische Überführung greifbar gemacht werden. Diese Kombination von Theorie und Praxis wurde genutzt, um Leitfäden für die Erstellung hybrider Lehre und Lernräume zu erstellen, welche ein tieferes Verständnis der umfassenden Thematik bereitstellen und so auch eine schnellere Adaptierung auf eigene Kontexte ermöglichen. So wurde für geschaffene Modellen auch stets die praktische Verwendung beschrieben und für diese wenn nötig zusätzliches Hilfsmaterial wie beispielsweise Fragebögen angeboten. Die Erstellung dieser Modelle und Materialien basierte auf übergreifenden Konzepten, sodass diese entgegen vieler bestehender Modelle nicht auf einen speziellen Kontext optimiert sind und sich für dynamische Erkundungen diverser Kontexte eignen. Die erstellte Architektur bietet beispielsweise einen Abstraktionsgrad, welcher für ein Verständnis und eine gezielte Umsetzung ausreicht, jedoch kontextspezifische Ergänzungen und Anpassungen ermöglicht. Auch bei der Erstellung des Design-Frameworks wurde eine hohe Flexibilität angestrebt, sodass es innerhalb einzelner Fachbereiche mit fachspezifischen Komponenten angereichert werden kann. Es ist demnach nicht nur für die Erforschung und Konzeption hybrider Systeme nutzbar, sondern auch für die Planung kontextgerechter hybrider Veranstaltungen. Ebenso sind mit der Entitäts- und Dimensionsübersicht sowie dem Design-Framework bestehende Veranstaltungen, Räume und Systeme strukturiert analysierbar. Wiederkehrende Strukturen können dann als Good Practices festgehalten werden und in folgenden Designprozessen als praktische Vorlagen gelten. Die Ergebnisse dieser Arbeit können also diverse Vorlagen für die praktische Umsetzung hybrider Lehre bieten, welche vor allem in Anbetracht des in Abschnitt 1.2 genannten zeitlichen Drucks dringend benötigt werden.

Die Abstraktion der Ergebnisse ist jedoch auch ein Aspekt, welcher kritisch zu betrachten ist. Diese war zwar für eine diverse und praktische Verwertbarkeit sowie die begrenzte Bearbeitungszeit dieser Arbeit notwendig, jedoch können dadurch auch wichtige Elemente übersehen worden sein. Trotz einer ausführlichen Recherche könnten beispielsweise grundlegende Entitäten und Eigenschaften fehlen. Da die Ergebnisse dieser Arbeit vollständig aufeinander aufbauen, sind dadurch auch das Design-Framework und die Systemarchitektur betroffen, welche dann ebenfalls einer Anpassung unterzogen werden müssen. Die Ergänzung domänenspezifischer Bestandteile kann dabei in allen Modellen dynamisch geschehen, jedoch ist die Integration neuer Kernelemente mit einem höheren Aufwand verbunden. Dies ist besonders in Zusammenhang mit dem rapiden technischen Wandel zu bedenken. Bereits bei der Definitionsbildung wurde diese Abhängigkeit hervorgehoben, sodass auch die bereitgestellten Definitionen keine endgültigen Ergebnisse darstellen. In Bezug auf diese Abstraktion und Eingrenzung untersuchter Elemente ist ebenfalls das Design-Framework erneut zu betrachten. Hier wurden zugunsten technischer und raumbedingter Komponenten die didaktischen Bestandteile nur oberflächlich analysiert. Diese haben jedoch einen erheblichen Einfluss auf das Raumdesign, sodass auch die Erkenntnisse einer tieferen Ergründung des Interaction Designs und dessen Wechselwirkung mit dem gegebenen Kontext in das Modell einfließen sollten. Ebenso ist das Framework in der Praxis zu testen, vor allem da es auf modifizierten Aktivitätstypen aufbaut, deren Nutzen noch nicht evaluiert wurde. Auch durch diese Erprobung können neue Details identifiziert werden und zu weiteren Iterationen des Modells führen. Ähnlich wurde auch die Architektur bisher auf einer rein theoretischen Basis erstellt, sodass noch keine genaue Aussage über ihre praktische Verwertbarkeit und den allgemeinen Nutzen getätigt werden kann. Letztlich wurden in der Architektur vor allem logische Verknüpfungen und Zuständigkeitsbereiche dargestellt, Implementierungs- und Deployment-Details jedoch noch nicht betrachtet, deren Planung weiterhin einen hohen Aufwand darstellen und eine große Auswirkung auf den endgültigen Erfolg des Systems haben können.

## 5.2. Ausblick

Abschließend werden nun mögliche Erweiterungen des Forschungsprozesses genannt. Jedes Teilergebnis dieser Arbeit wurde im vorherigen Abschnitt kritisch bewertet. Dabei zeichneten sich gesonderte Überarbeitungs- und Erweiterungsmöglichkeiten ab, welche folgend in einzelnen Abschnitten kurz aufgeführt sind.

Sowohl die Definitionen, als auch Kernentitäten und Dimensionen sollten der Forschungsgemeinde bereitgestellt und aus der Perspektive unterschiedlicher Fachdiszipli-



nen evaluiert, sowie diskutiert werden. Sie versuchen eine einheitliche Grundlage zu gewährleisten, welche erst durch interdisziplinäre Zustimmung und Anreicherung erreicht wird.

Dies gilt auch für das Design-Framework. Es kann zudem für unterschiedliche Fachdisziplinen mit konkreten Auswahlmöglichkeiten erweitert werden, sodass es eine kontextspezifisch detailliertere Basis des Designs hybrider Lehrveranstaltungen und Systeme bietet. Hierüber ist dann eine Sammlung erprobter Konstellationen mitsamt ihrer Vor- und Nachteile möglich, welche beispielsweise als Pattern-Language veröffentlicht werden kann. Diese würde dann eine schnelle Auswahl kontextgerechter Lösungen auch durch unerfahrene Individuen begünstigen. Hierbei sollte auch die Einbindung des physischen Raums und der Medienökologien genauer betrachtet werden. Durch die Anwendung des Frameworks werden diesbezüglich effektive Herangehensweisen bekannt, welche ebenso in Patterns festgehalten werden können.

Letztlich erfordert die präsentierte Systemarchitektur für eine praktische Nutzung noch einige Arbeitsschritte. Sie sollte zunächst prototypisch umgesetzt und so ihre Machbarkeit geprüft werden. Dabei sind besonders die kritischen Bestandteile einem Test zu unterziehen, allem voran das Plugin-System mitsamt der dafür benötigten Basis-Services. Stellen sich diese als valide heraus, so sollten zunächst Deployment-Strukturen geklärt werden, bevor die Implementierung des restlichen Basis-Systems beginnt. Dabei ist zu entscheiden, ob das System als globaler Service auch externen Instanzen bereitstehen soll, oder ob lediglich die Bereitstellung als Bibliothek bzw. Image für lokale Nutzungen angestrebt wird. Ebenso sollten die Möglichkeiten von Plugins genauer spezifiziert werden. So ist zu entscheiden, welche globalen Informationen im Frontend bereitgestellt bzw. welche Information einer Render-Funktion allgemein übergeben werden. Zusätzlich sind die Channels für eine pluginübergreifende Kommunikation festzulegen, sowie das zugehörige Messaging-Format. Damit verbunden ist auch die Identifikation des Frontend-Stils sowie die Definition eines allgemeinen Style-Guides, sodass sich Plugins bestmöglich in die Gesamtanwendung eingliedern. Weiterhin sind mithilfe des Design-Frameworks grundlegende Algorithmen, Ressourcen und Aktivitäten zu identifizieren, welche einen übergreifenden Nutzen aufweisen. Hierzu zählen beispielsweise Chat- und Streaming-Funktionen, die Bearbeitung unterschiedlicher Textformate und die Integration gängiger Speicher wie DropBox oder Sciebo. Dieses grundlegende System ist dann in Nutzertests zu evaluieren, wodurch weitere, notwendige Bestandteile ersichtlich werden und eine Einschätzung der *Usability* stattfinden kann. Erst wenn es in den Tests als solide Basis anerkannt wird, sollte eine Veröffentlichung erfolgen.

## Abbildungsverzeichnis

1.2. Benyons mentale Räume, inhaltlich entnommen aus Benyon and Mival [2012] . . . . .	27
2.1. Pedagogy–Space–Technology Framework, inhaltlich entnommen aus Radcliffe [2009] . . . . .	36
2.2. Komponenten hybrider Lernräume und deren Beziehungen, inhaltlich entnommen aus Trentin [2015, S. 3] . . . . .	37
2.3. Abstrahiertes Modell von Lernräumen, inhaltlich entnommen aus Ellis and Goodyear [2016, S. 175] . . . . .	37
2.4. Dimensionen hybrider Lernräume und Aktivitätstypen, inhaltlich entnommen aus Trentin [2015, S. 9] . . . . .	41
2.5. Dimensionen des Blended Learnings, entnommen aus Suo and Shi [2008, S. 196] . . . . .	42
2.6. Dimensionspyramide: Hierarchie von Lernraumattributen, inhaltlich entnommen aus Cunningham and Tabur [2012, S. 1] . . . . .	44
2.7. Place for Learning Spectrum, entnommen aus Radcliffe et al. [2008, S. 13] . . . . .	46
2.8. Online Hybrid Learning Model, inhaltlich entnommen aus Mossavar-Rahmani and Larson-Daugherty [2007, S. 3] . . . . .	48
2.9. Ablauf eines Lernprozesses . . . . .	50
2.10. Nutzung von Lernräumen innerhalb eines Lernprozesses . . . . .	51
2.11. Gesamtübersicht identifizierter Entitäten hybrider Lehre und ihre Zusammenhänge . . . . .	52
2.12. Identifizierte Entitäten hybrider Lehre und ihre Zusammenhänge, mit Beispielen . . . . .	55
3.1. Activity-centred model, entnommen aus Goodyear and Ellis [2008, S. 24] . . . . .	62
3.2. Aktivitätszentriertes Framework als Grundlage des Designs hybrider Systeme . . . . .	72
3.3. Erweitertes Übersichtsmodell für das Design hybrider Systeme . . . . .	78
4.1. Abstraktes Klassendiagramm beinhalteteter Entitäten . . . . .	82
4.2. Architektur des Basis-Systems . . . . .	99
4.3. Bereitstellung der Services über Gateways . . . . .	102
4.4. Dynamische Microservice-Architektur für hybride Lernraumsysteme . . . . .	105
4.5. Registrierung eines Artefakt-Plugins . . . . .	106
4.6. Rendering eines Artefakts . . . . .	107

---

## Tabellenverzeichnis

1.2. Vergleich hybrider Lehransätze. Inhalt entnommen aus Irvine [2020, S. 3]	19
1.3. Vergleich der Bedeutungen von Objekten, Personen und Rollen in traditioneller und HLS-Lehre nach Trentin [vgl. 2015, S. 7]	30
2.1. Eigenschaften hybrider Lehre und deren identifizierte Ausprägungen	58
3.1. ACAD-Cards (Activity-Centred Analysis and Design), inhaltlich entnommen aus Goodyear et al. [2021, S. 456]	64
3.2. ACAD-Wireframe (Activity-Centred Analysis and Design), inhaltlich entnommen aus Goodyear et al. [2021, S. 457]	65
3.3. Taxonomie mit sechs Aktivitätstypen nach Conole [vgl. 2007, S. 84 ff.]	68
3.4. Liste identifizierter Aktivitätstypen	70
B.1. Beispielhafte Fragen zur Ermittlung notwendiger Aktivitäten	125
C.1. Fragen für das Design hybrider Systeme	127

## Literaturverzeichnis

- Hans Aebli. 2006. *Zwölf Grundformen des Lehrens: eine allgemeine Didaktik auf psychologischer Grundlage* (13 ed.). Klett-Cotta.
- Noor M Alqudah, Hisham M Jammal, Omar Saleh, Yousef Khader, Nail Obeidat, and Jumana Alqudah. 2020. Perception and experience of academic Jordanian ophthalmologists with E-Learning for undergraduate course during the COVID-19 pandemic. *Annals of Medicine and Surgery* 59 (2020), 44--47.
- Brian J Beatty. 2007. Hybrid classes with flexible participation options--If you build it, how will they come. *2007 Annual Proceedings-Anaheim: Volume 15* (2007).
- David Benyon. 2014. Spaces of interaction, places for experience. *Synthesis Lectures on Human-Centered Information* 7, 2 (2014), 1--129.
- David Benyon and Oli Mival. 2012. Blended spaces for collaborative creativity. In *Proceedings of Workshop on Designing Collaborative Interactive Spaces AVI2012*. Citeseer.
- David Benyon and Oli Mival. 2015. Blended spaces for collaboration. *Computer Supported Cooperative Work (CSCW)* 24, 2 (2015), 223--249.
- Benjamin S Bloom, David R Krathwohl, and Bertram B Masia. 1984. Bloom taxonomy of educational objectives. In *Allyn and Bacon*. Pearson Education.
- Ruth Boelens, Bram De Wever, and Michiel Voet. 2017. Four key challenges to the design of blended learning: A systematic literature review. *Educational Research Review* 22 (2017), 1--18.
- Curtis J Bonk and Charles R Graham. 2012. *The handbook of blended learning: Global perspectives, local designs*. John Wiley & Sons.
- Matt Bower, Barney Dalgarno, Gregor E Kennedy, Mark JW Lee, and Jacqueline Kenney. 2015. Design and implementation factors in blended synchronous learning environments: Outcomes from a cross-case analysis. *Computers & Education* 86 (2015), 1--17.
- Theresa Capra. 2011. Online education: Promise and problems. *Journal of Online Learning and Teaching* 7, 2 (2011), 288--293.
- Tak-Wai Chan, Fei-Ching Chen, and Chih-Yueh Chou. 2006. Profile enhanced classroom learning. In *2006 Fourth IEEE International Workshop on Wireless, Mobile and Ubiquitous Technology in Education (WMTE'06)*. IEEE, 3--6.

- 
- Anat Cohen, Rikke Toft Nørgård, and Yishay Mor. 2020. Hybrid learning spaces----- Design, data, didactics. (2020).
- Gráinne Conole. 2007. Describing learning activities. *Rethinking pedagogy for a digital age: Designing and delivering e-learning* (2007), 81--91.
- Melvin E. Conway. 1968. How Do Committees Invent? (1968). Retrieved 2021-08-23 from <http://www.melconway.com/research/committees.html>
- Petra HM Cremers, Arjen EJ Wals, Renate Wesselink, and Martin Mulder. 2016. Design principles for hybrid learning configurations at the interface between school and workplace. *Learning Environments Research* 19, 3 (2016), 309--334.
- Heather V Cunningham and Susanne Tabur. 2012. Learning Space Attributes: Reflections on Academic Library Design and Its Use. *Journal of learning spaces* 1, 2 (2012), n2.
- Unna N Danner, Henk Aarts, and Nanne K de Vries. 2008. Habit vs. intention in the prediction of future behaviour: The role of frequency, context stability and mental accessibility of past behaviour. *British Journal of Social Psychology* 47, 2 (2008), 245--265.
- Elly De Bruijn and Yvonne Leeman. 2011. Authentic and self-directed learning in vocational education: Challenges to vocational educators. *Teaching and Teacher Education* 27, 4 (2011), 694--702.
- LGA De Putter-Smits, R Taconis, and WMG Jochems. 2013. Mapping context-based learning environments: The construction of an instrument. *Learning environments research* 16, 3 (2013), 437--462.
- Margaret Driscoll. 2002. Blended learning: Let's get beyond the hype. *E-learning* 1, 4 (2002), 1--4.
- Charles Duhigg. 2012. *Die Macht der Gewohnheit: warum wir tun, was wir tun*. ebook Berlin Verlag.
- Universitätsklinikum Düsseldorf. 2021. Arbeitsformen und Methoden. (2021). Retrieved 2021-08-23 from <https://www.uniklinik-duesseldorf.de/ausbildung-karriere/ausbildungen/fachbereich-physiotherapie/paedagogisches-verstaendnis/arbeitsformen-und-methoden>
- e-teaching.org. [n. d.]. Definition für Blended Learning. ([n. d.]). Retrieved 2021-08-23 from <https://www.e-teaching.org/materialien/glossar/blended-learning>
- e-teaching.org. 2017. Lehrmethoden. (2017). Retrieved 2021-08-23 from <https://www.e-teaching.org/didaktik/konzeption/methoden>
- e-teaching.org. 2021a. Lehrszenarien. (2021). Retrieved 2021-08-23 from <https://www.e-teaching.org/lehrszenarien>

- 
- e-teaching.org. 2021b. Lernmanagement-Systeme (LMS). (2021). Retrieved 2021-08-23 from <https://www.e-teaching.org/technik/distribution/lernmanagementsysteme/>
- Wolfgang Einsiedler. 1981. *Lehrmethoden. Probleme und Erkenntnisse der Lehrmethodenforschung*. Urban & Schwarzenberg.
- Robert A Ellis and Peter Goodyear. 2016. Models of learning space: integrating research on space, place and learning in higher education. *Review of Education* 4, 2 (2016), 149–191.
- Dieter Euler, Sabine Seufert, and Karl Wilbers. 2006. eLearning in der Berufsbildung. In *Handbuch der Berufsbildung*. Springer, 432–450.
- Eric Evans and Eric J Evans. 2004. *Domain-driven design: tackling complexity in the heart of software*. Addison-Wesley Professional.
- Facebook. 2018. GraphQL Spezifikation 2018. (2018). Retrieved 2021-08-23 from <http://spec.graphql.org/June2018/#sec-Overview>
- Gilles Fauconnier and Mark Turner. 2008. *The way we think: Conceptual blending and the mind's hidden complexities*. Basic Books.
- Eric Freeman, Elisabeth Robson, Bert Bates, and Kathy Sierra. 2008. *Head first design patterns*. O'Reilly Media, Inc.
- Gabi Reinmann. 2021. Vortrag-Praesent-Hybrid-2021. (20 3 2021). Retrieved 2021-08-23 from <https://lecture2go.uni-hamburg.de/12go/-/get/v/49152>
- Carla Garnham and Robert Kaleta. 2002. Introduction to hybrid courses. *Teaching with technology today* 8, 6 (2002), 5.
- Peter Goodyear, Lucila Carvalho, and Pippa Yeoman. 2021. Activity-Centred Analysis and Design (ACAD): core purposes, distinctive qualities and current developments. *Educational Technology Research and Development* (2021), 1–20.
- Peter Goodyear and Robert A Ellis. 2008. University students' approaches to learning: rethinking the place of technology. *Distance Education* 29, 2 (2008), 141–152.
- Charles R Graham. 2006. Blended learning systems. *The handbook of blended learning: Global perspectives, local designs* 1 (2006), 3–21.
- Jan Herrington and Ron Oliver. 2000. An instructional design framework for authentic learning environments. *Educational technology research and development* 48, 3 (2000), 23–48.
- Charlotta Hilli, Rikke Toft Nørgård, and Janus Holst Aaen. 2019. Designing hybrid learning spaces in higher education. *Dansk Universitetspædagogisk Tidsskrift* 15, 27 (2019), 66–82. <https://tidsskrift.dk/dut/article/view/112644>

- 
- Hans Hinterberger, Lukas Fässler, and Bettina Bauer-Messmer. 2004. From hybrid courses to blended learning: A case study. *ICNEE, 27-30 September 2004* (2004), 1--7.
- James Hollan, Edwin Hutchins, and David Kirsh. 2000. Distributed cognition: toward a new foundation for human-computer interaction research. *ACM Transactions on Computer-Human Interaction (TOCHI)* 7, 2 (2000), 174--196.
- Valerie Irvine. 2020. The landscape of merging modalities. (2020).
- Valerie Irvine, Jillianne Code, and Luke Richards. 2013. Realigning higher education for the 21st century learner through multi-access learning. *Journal of Online Learning and Teaching* 9, 2 (2013), 172.
- Cam Jackson. 2019. Micro Frontends. (2019). Retrieved 2021-08-23 from <https://martinfowler.com/articles/micro-frontends.html#IntegrationApproaches>
- Fred Kent and Phil Myrick. 2003. How to become a great public space. *American Libraries* 34, 4 (2003), 72--76.
- Christian Kohls. 2017. Hybrid learning spaces. In *Proceedings of the VikingPLoP 2017 Conference on Pattern Languages of Program*. 1--12.
- Christian Kohls. 2019. Hybrid Learning Spaces for Design Thinking. *Open Education Studies* 1, 1 (2019), 228--244.
- Christian Köppe, Rikke Toft Nørgård, and Alex Young Pedersen. 2017. Towards a pattern language for hybrid education. In *Proceedings of the VikingPLoP 2017 Conference on Pattern Languages of Program*. 1--17.
- David R Krathwohl. 2002. A revision of Bloom's taxonomy: An overview. *Theory into practice* 41, 4 (2002), 212--218.
- George D Kuh. 1996. Guiding principles for creating seamless learning environments for undergraduates. *Journal of college student development* 37, 2 (1996), 135--48.
- Chee-Kit Looi, Peter Seow, BaoHui Zhang, Hyo-Jeong So, Wenli Chen, and Lung-Hsiang Wong. 2010. Leveraging mobile technology for sustainable seamless learning: a research agenda. *British journal of educational technology* 41, 2 (2010), 154--169.
- Patrick Lowenthal, Jered Borup, Richard West, and Leanna Archambault. 2020. Thinking beyond Zoom: Using asynchronous video to maintain connection and engagement during the COVID-19 pandemic. *Journal of Technology and Teacher Education* 28, 2 (2020), 383--391.
- Abraham Harold Maslow. 1943. A theory of human motivation. *Psychological review* 50, 4 (1943), 370.
- methopedia.eu. 2017. Methopedia: Sammlung von Unterrichtsmethoden. (2017). Retrieved 2021-08-23 from <https://methopedia.eu/de/categories/methode/>

- 
- H Meyer. 2009. *Unterrichts-Methoden* (13. Auflage). Berlin: Cornelsen Verlag Scriptor (2009).
- Iwona Miliszewska. 2008. Transnational education programs: Student reflections on a fully-online versus a hybrid model. In *International Conference On Hybrid Learning And Education*. Springer, 79--90.
- Farhang Mossavar-Rahmani and Cynthia Larson-Daugherty. 2007. Supporting the hybrid learning model: A new proposition. *MERLOT Journal of Online Learning and Teaching* 3, 1 (2007), 67--78.
- Anders Norberg, Charles D Dziuban, and Patsy D Moskal. 2011. A time-based blended learning model. *On the Horizon* (2011).
- QUA-LiS NRW. 2018. QUA-LiS NRW: Methodensammlung. (2018). Retrieved 2021-08-23 from <https://www.schulentwicklung.nrw.de/methodensammlung/liste.php>
- W Ian O'Byrne and Kristine E Pytash. 2015. Hybrid and blended learning: Modifying pedagogy across path, pace, time, and place. *Journal of Adolescent & Adult Literacy* 59, 2 (2015), 137--140.
- Falko Peschel. 2003. *Offener Unterricht*. Schneider-Verlag Hohengehren.
- David Radcliffe. 2009. A pedagogy-space-technology (PST) framework for designing and evaluating learning places. In *Learning spaces in higher education: Positive outcomes by design. Proceedings of the Next Generation Learning Spaces 2008 Colloquium, University of Queensland, Brisbane*. 11--16.
- David Radcliffe, Hamilton Wilson, Derek Powell, and Belinda Tibbetts. 2008. Designing next generation places of learning: Collaboration at the pedagogy-space-technology nexus. *The University of Queensland* (2008), 1--20.
- Kersten Reich. 2017. Universität zu Köln: Methodenpool. (2017). Retrieved 2021-08-23 from [http://methodenpool.uni-koeln.de/frameset\\_uebersicht.htm](http://methodenpool.uni-koeln.de/frameset_uebersicht.htm)
- Gabi Reinmann. 2021. Hybride Lehre – Ein Begriff und seine Zukunft für Forschung und Praxis. *Impact Free - Journal für freie Bildungswissenschaftler* 35 (2 2021), 1--10.
- Chris Richardson. 2018. *Microservices patterns: with examples in Java*. Simon and Schuster.
- Dietmar Rösler. 2012. *Lehr- und Lernformen*. J.B. Metzler, Stuttgart. [https://doi.org/10.1007/978-3-476-05284-1\\_6](https://doi.org/10.1007/978-3-476-05284-1_6)
- Niall Sclater, Alice Peasgood, and Joel Mullan. 2016. Learning analytics in higher education. *London: Jisc. Accessed February 8, 2017* (2016), 176.
- Patricia L Smith and Tillman J Ragan. 2004. *Instructional design*. John Wiley & Sons.



- 
- Michael Smythe. 2012. Toward a framework for evaluating blended learning. *Future challenges, sustainable futures. Proceedings ascilite Wellington* (2012), 854–858.
- Jesse Stommel. 2012. Hybridity, pt. 2: What is Hybrid Pedagogy? (2012). Retrieved 2021-08-23 from <https://hybridpedagogy.org/hybridity-pt-2-what-is-hybrid-pedagogy/>
- Jesse Stommel and Pete Rorabaugh. 2012. Hybridity, pt. 3: What Does Hybrid Pedagogy Do? (2012). Retrieved 2021-08-23 from <https://hybridpedagogy.org/hybridity-pt-3-what-does-hybrid-pedagogy-do/>
- Yue Suo and Yuanchun Shi. 2008. Towards blended learning environment based on pervasive computing technologies. In *International Conference on Hybrid Learning and Education*. Springer, 190–201.
- Guglielmo Trentin. 2015. Orientating pedagogy towards hybrid learning spaces. *Progress in education* 35 (2015), 105–124.
- Mark Van Harmelen. 2006. Personal learning environments. In *Sixth International Conference on Advanced Learning Technologies*. IEEE Computer Society, 815–816.
- Arjen EJ Wals, Thomas Lans, and Hendrik Kupper. 2012. Blurring the boundaries between vocational education, business and research in the agri-food domain. *Journal of Vocational Education & Training* 64, 1 (2012), 3–23.
- Eberhard Wolff. 2018. *Microservices: Grundlagen flexibler Softwarearchitekturen*. dpunkt. verlag.
- Lung-Hsiang Wong and Chee-Kit Looi. 2011. What seams do we remove in mobile-assisted seamless learning? A critical review of the literature. *Computers & Education* 57, 4 (2011), 2364–2381.
- Lung-Hsiang WONG and Mike SHARPLES. 2013. Seamless learning: An international perspective on next generation technology enhanced learning. (2013).
- Ji-Ping Zhang. 2008. Hybrid learning and ubiquitous learning. In *International Conference on Hybrid Learning and Education*. Springer, 250–258.

# Anhang

## A. Dichotomien für das Design hybrider Lehrveranstaltungen

Zwei der meistreferenzierten Quellen bezüglich hybrider Dimensionen benennen diese als Auflistungen von Dichotomien, welche bei der didaktischen Gestaltung von Lehrveranstaltungen zu beachten sind. Diese Auflistungen werden folgend präsentiert.

### A.1. Dichotomien des Blended Learnings

Bonk and Graham [vgl. 2012, S. 20] verdeutlichen in ihrem zusammenfassenden Werk bezüglich des Blended Learnings eine Vielzahl unterschiedlicher Dichotomien. Diese Varietät begründen sie unter anderem mit der Diversität der in dem Sammelband beinhalteten Ansichten und Forschungsrichtungen / -ergebnissen verschiedener Autoren, welche sich mit dem Feld des Blended Learnings auseinandersetzen. Sie heben dabei hervor, dass die Gestaltung von Lehrveranstaltungen optimal als *blend of blends* zu sehen ist, bei welchem demnach kontextgerechte Mischformen mehrerer Dichotomien einzusetzen sind. Die 13 von ihnen genannten Dichotomien lauten dabei wie folgt (Bonk and Graham [vgl. 2012, S. 20]):

- Fleeting know-how / Lasting knowledge
- Individual / Community
- Generic / Proprietary
- Training / Knowledge sharing
- Text / Visual
- Self-directed / Guided navigation
- Content focus / Experience focus
- Exploring / Participating
- Push / Pull
- Personalized / One-size-fits-all
- Skills / Values
- Information / Transformation
- Formal / Informal

---

## A.2. Dichotomien der Hybrid Pedagogy

Stommel [2012] verdeutlicht die Vielschichtigkeit und Interdisziplinarität des hybriden Ansatzes, indem er beispielhaft 17 Gegensätze nennt. Diese seien bei der Planung einer Lehrveranstaltung zu berücksichtigen und ihre Unterschiede kontextgerecht zu überbrücken. Die neu geschaffenen Mischformen sollen dabei jedoch keine isolierte Betrachtung hervorrufen, sondern weiterhin die Ausgangspunkte und Herkünfte jedes Bestandteils ersichtlich machen. Es geht demnach nicht um eine unbemerkte Grenzüberschreitung, sondern um eine Definition klarer Grenzen und Gemeinsamkeiten, entlang derer interdisziplinäre Lehre entstehen kann. Die 17 Dichotomien lauten:

- Physical Learning Space / Virtual Learning Space
- Academic Space / Extra-academic Space
- On-ground Classrooms / Online Classrooms
- Permanent Faculty / Contingent Faculty
- Institutional Education / Informal Education
- Garden-walled Academia / Open Education
- Scholars / Teachers
- Academic Product / Learning Process
- Disciplinarity / Interdisciplinarity
- Performed (School-y) Selves / Real (Vulnerable) Selves
- Individual Teachers, Students, and Scholars / Collaborative Communities
- Learning in Schools / Learning in the World
- Analog Pedagogy / Digital Pedagogy
- Use of Tools / Critical Engagement with Tools
- Machine and Machine-like Interaction / Human Interaction
- Passive Learning / Experiential Learning
- Teaching and Learning / Critical Pedagogy

## B. Beispielhafte Fragen zur Ermittlung notwendiger Aktivitäten

Ergänzend zur Darstellung der unterschiedlichen Aktivitätstypen in Abschnitt 3.2 werden in Tabelle B.1 Fragen zur Ermittlung notwendiger Aktivitäten des jeweiligen Typen vorgestellt. Die Auflistung ist dabei nicht erschöpfend. Sie soll vor allem eine mögliche Herangehensweise präsentieren, durch welche die Bedeutung der Typen praktisch verdeutlicht wird.

Tabelle B.1.: Beispielhafte Fragen zur Ermittlung notwendiger Aktivitäten

Typ	Mögliche Fragen
Darbietend	Müssen Ergebnisse vorgestellt werden? Wem und von wem werden Informationen präsentiert? Welche Präsentationsform ist angestrebt?
Assimilativ	Welche Ressourcen werden zum Lernen benötigt? Wie, von wem und für wen werden sie bereitgestellt?
Erarbeitend	Ist neben der Bereitstellung von Ressourcen eine eigenständige Erarbeitung von Wissen notwendig? Welche Quellen werden bei Recherchen genutzt? Wie werden Ressourcen gesammelt, sortiert und festgehalten? Wie werden Ressourcen analysiert, Informationen extrahiert und klassifiziert?
Adaptiv	Wie kann theoretisches Wissen praktisch erprobt oder praktisches Wissen festgehalten werden? Wie können physische Aktivitäten und Objekte digitalisiert werden? Wie lassen sich digitale Aktivitäten und Artefakte in physische Aufgaben integrieren?
Kommunikativ / Sozial	Wie und mit wem wird Kontakt aufgebaut und gehalten? Welche Art der Kommunikation ist für die Aufgabenstellung notwendig? Werden Soziale Medien oder andere externe Instanzen eingebunden? Wie transparent sind Lernprozesse und Ergebnisse?

Typ	Mögliche Fragen
Produktiv	<p>Ist ein Artefakt als Ergebnis notwendig?            Welche Form der Wissensrepräsentation eignet sich?            Werden Artefakte kollaborativ oder individuell erstellt?            Wie werden Artefakte übermittelt und für wen sind diese sichtbar?</p>
Explorativ	<p>Ist ein Projekt oder Selbststudium sinnvoll?            Wie können Denkanstöße gegeben werden?            Wie werden Ideen gesammelt?            Wie kann der Lernprozess verfolgt werden?</p>
Organisatorisch / Administrativ	<p>Wie werden Gruppen erstellt?            Welche organisatorischen Schritte sind vor der Durchführung des Lernens / Lehrens notwendig?            Wie können Arbeitsschritte geplant, festgehalten, zugeordnet und kontrolliert werden?            Wer kann sich bei der Strukturierung des Lernraums wie beteiligen?</p>
Begleitend / Support	<p>Wer kann und sollte eine Support-Funktion übernehmen?            Wie können Lehrende oder Lernende andere bei ihrer Tätigkeit unterstützen?            Woran kann der Bedarf nach einer Unterstützung erkannt werden bzw. wie wird dieser mitgeteilt und ermöglicht?</p>
Reflektiv / Evaluierend	<p>Müssen Lernende ihre Ergebnisse hinterfragen oder gegenseitig bewerten?            Ist direktes Feedback erwünscht und wie ist es möglich?            Was wird reflektiv bewertet und wie wird diese Bewertung festgehalten bzw. mitgeteilt?</p>
Prüfend	<p>Muss eine Überprüfung erlernten Wissens stattfinden?            Was ist die Prüfungsgrundlage? (Artefakte, Prozesse, Individuum, Gruppe...)            Wie wird die Benotung übermittelt?            Bauen Benotungen aufeinander auf?</p>

## C. Zu beantwortende Fragen beim Design hybrider Systeme

Beim Design hybrider Systeme sind systematisch einige Fragen zu beantworten, welche sowohl das Lehrscenario als auch den Funktionsumfang des Systems, physische Gegebenheiten und private Medienökologien betreffen. Als Leitfaden für solche Designprozesse wird in Tabelle C.1 eine Auflistung der wichtigsten Fragen präsentiert, welche im besten Falle in der dargestellten Reihenfolge zu beantworten sind.

Tabelle C.1.: Fragen für das Design hybrider Systeme

Designschritt	Entitäten	Fragen
<b>Lehr- / Lernscenario definieren</b>	Lehr- / Lernkontext	Was sind die Anforderungen und Restriktionen des Curriculums?
		Welche örtlichen und zeitlichen Gegebenheiten sind vorbestimmt?
		Welche Geräte, Garnituren, Ressourcen und Personen sind eingeplant bzw. im Campus vorhanden?
	Unterrichtsdesign	Welche Lernziele sollen erreicht werden?
		Welche Sozialformen sind dabei hilfreich / notwendig?
		Welche Arbeitsformen sind wünschenswert?
		Welche Methoden passen zu dem definierten Ziel, sowie den Sozial- und Arbeitsformen?
		Wie muss der Kontext angepasst werden um das Unterrichtsdesign zu ermöglichen?

Designschritt	Entitäten	Fragen
<b>Lernprozess &amp; Arbeitsschritte planen</b>	Settings	Was für Settings sind bei der Aufgabenbewältigung nutzbar / sinnvoll?
	Aktivitäten	Welche Aktivitätstypen sind in den einzelnen Arbeitsschritten notwendig? (siehe B.1)
		Welche Lernziele können wie und in welchen Artefakttypen festgehalten werden?
	Social Design	Welche Communities nehmen an einzelnen Aktivitäten teil?
		Gibt es individuelle und kollaborative Teil-Aktivitäten?
		Wie finden Gruppeneinteilungen statt?
	Set Design	Welche Rollen haben die Teilnehmer innerhalb einer Aktivität / Was müssen sie in dieser Aktivität machen?
		Welche Anwendungen ermöglichen eine oder mehrere Aktivitäten?
		Welche Ressourcen werden vor oder während einer Aktivität benötigt / gegeben?
		Gibt es individuelle und kollaborative Teil-Aktivitäten?
		Wie bzw. über welche Geräte erhalten potentielle Nutzer Zugriff auf den digitalen Raum?
		Für wen ist das System sichtbar?
	Weitere Systemkomponenten	Welche Zugangsmethoden gibt es? (Login, QR-Code, etc.)
		Welche Speichermöglichkeiten muss das System für Ressourcen und Artefakte bieten?
Welche Algorithmen sollten parallel zum Lernprozess automatisch agieren?		



Designschritt	Entitäten	Fragen
<b>Kontext- gegebenheiten berücksichtigen</b>	Physischer Raum	Wie können die Gegebenheiten des physischen Raums genutzt werden / unterstützen?
		Wie kann das digitale System den physischen Raum ergänzen?
		Welche Infrastruktur steht im Setting zur Verfügung?
		Welche Communities sind in dem Setting anzutreffen?
	Private Medienökologien	Welche Aspekte der privaten Medienökologien sollten einfließen?
		Welche Quellen oder Anbieter für Ressourcen und die Artefaktspeicherung können eingebunden werden?
		Welche vielgenutzten Anwendungen ermöglichen bereits eine Aktivität und müssen nicht neuentwickelt werden? Welche Integrationsmöglichkeiten bieten sie?
		Welche privaten Geräte und Objekte können und sollten eingeplant / unterstützt werden, um einen Zugriff auf das System und gewisse Aktivitäten zu ermöglichen?
		Welche externen Communities sollten eingebunden werden und wie können diese am Lernprozess partizipieren bzw. dabei unterstützen?

## D. Wichtige Bestandteile einer Microservice-Architektur

Durch die steigende Nutzung von Microservices innerhalb der letzten Jahre ist dieser Ansatz zu einem etablierten Architekturstil herangewachsen. Durch damit gesammelte Erfahrungen wurden mehr und mehr *Good Practices* bekannt, sodass sich einzelne Komponenten als wichtige Kernbestandteile abzeichnen. Ebenso finden sich Pattern und Tutorials bezüglich vorteilhafter Herangehensweisen für das Design solch einer Architektur. In diesem Abschnitt werden nun die wichtigsten Komponenten kurz erläutert und zugehörige Pattern des Designs und der Nutzung angesprochen.

### D.0.1. Service

Services bilden den Kern einer Microservice-Architektur. Sie sind unabhängige Grundbausteine, welche kleine Zuständigkeitsbereiche abdecken und zusammen das gesamte System abbilden. Auch wenn der Name „Microservice“ eine Aussage über ihre Größe vermuten lässt, ist diese bei dem Design eines Services weniger entscheidend [Richardson, 2018, S. 43]. Die Planung eines Services folgt laut Wolff [vgl. 2018, S. 2] vielmehr der UNIX-Philosophie, welcher nach ein Programm nur eine Aufgabe erledigen soll, diese jedoch bestmöglich umsetzen muss. Weiterhin sollen, dieser Philosophie folgend, Programme zusammenarbeiten können und eine universelle Schnittstelle nutzen. Dabei ist eine Isolation der Services wichtig, welche durch eine fachliche Architektur sichergestellt wird. Service-interne Prozesse sollten im besten Falle ohne externe Einflüsse agieren können und dringend notwendige Kommunikationen müssen lose gekoppelt über Schnittstellen geschehen. Durch diese lose Kopplung wird eine technologische Freiheit der einzelnen Services erreicht, wodurch bei einer Systemarchitektur lediglich die Zuständigkeitsbereiche, aber noch keine Implementierungsdetails zu betrachten sind. Dabei verweist Wolff [vgl. 2018, S. 39] auf das Gesetz von Conway:

„Any organization that designs a system (defined more broadly here than just information systems) will inevitably produce a design whose structure is a copy of the organization's communication structure.“ [Conway, 1968]

Demzufolge sind Services ebenso abhängig von der Struktur einer Organisation bzw. ihrer Kommunikationsstruktur. Die Planung von Services ist dann auch fest verbunden mit der Planung von Projektteams und Kommunikationshierarchien. Laut Richardson [2018, S. 43] sollten Services dabei so konzipiert werden, dass diese von einem kleinen Team innerhalb einer minimalen Durchlaufzeit und mit einem minimalen Kollaborationsaufwand zu anderen Teams implementiert werden können.

---

Bei der Planung von Services schlägt Richardson [2018] nun zwei mögliche Ansätze vor. Laut seinem ersten Ansatz sollten Services abhängig von den *Businessfunktionen* aufgeteilt werden [vgl. Richardson, 2018, S. 51 ff.]. Dafür sind zunächst die Zielstellungen, Strukturen und Prozesse einer Organisation zu analysieren, auf deren Basis die *top-level Funktionen* identifiziert werden können. Zu diesen werden dann hierarchisch die jeweiligen *Unterfunktionen* extrahiert, sodass beispielsweise bei der Buchhaltung eines Online-Lebensmittel-Lieferdienstes unterschiedliche Zahlungsmanagements für Konsumenten, Restaurants und Kuriere vorhanden sein sollten. Anhand dieser Hierarchie können nun dedizierte Services geschaffen werden, wobei ihre Einteilung subjektive und kontextspezifische Entscheidungen beinhaltet. So könnte eine ganze top-level Funktion als Service umgesetzt werden, wenn alle beinhalteten Unterfunktionen eine hohe Kohärenz aufweisen. Auch könnten für unterschiedliche Unterfunktionen ebenso unterschiedliche Services geschaffen werden, vorausgesetzt, sie weisen eine hohe Divergenz auf. Ist zwischen den Unterfunktionen verschiedener top-level Funktionen eine starke Gemeinsamkeit ersichtlich, so kann auch dies als gruppierender Faktor für einen gemeinsamen Service genutzt werden. So ist eine Einteilung nach den Businessfunktionen nicht immer eindeutig und benötigt entsprechende Vorlaufzeit. Der Vorteil einer vorausschauenden Strukturierung ist hier jedoch, dass identifizierte Funktionen beständig sind und mit geringerer Wahrscheinlichkeit Änderungen benötigen, sodass dies auch auf die resultierende Architektur zutrifft.

Der zweite Ansatz basiert auf einem *Domain-Driven-Design (DDD)*<sup>1</sup> und beschäftigt sich mit der Identifikation von *Sub-Domänen* [vgl. Richardson, 2018, S. 54 f.]. Innerhalb des DDD wird die Entwicklung mehrerer, objektorientierter Sub-Domänen-Modelle aus unterschiedlichen Perspektiven angestrebt, welche für weitere Entwicklungen ein umfangreiches Vokabular bereitstellen. Ähnlich der Identifikation von Businessfunktionen werden auch Sub-Domänen über die Analyse des Unternehmens gewonnen, wobei hier allgemeine Bestandteile der Domäne im Vordergrund stehen, während Businessfunktionen sich auf den direkten Kontext des Unternehmens fokussieren. Die Modelle dieser Sub-Domänen befinden sich dann in einem jeweiligen *Bounded Context*, zu welchen anschließend jeweils ein oder mehrere Services zugeordnet werden. Dieser Ansatz hat den Vorteil, dass aufgrund der detaillierten Aufteilung in Sub-Domänen und Bounded Contexts automatisch eine Isolation stattfindet, welche die Bildung von *Gott-Klassen* verhindert und eine Aufteilung in Services vereinfacht.

Zur Unterstützung dieser Konzepte verweist Richardson [vgl. 2018, S. 56 ff.] zudem auf zwei der Prinzipien des Objekt-orientierten Designs von Bob Martin<sup>2</sup>. Ähnlich dem *Single Responsibility Principle* sollten auch Services nur eine Zuständigkeit und nur einen Grund haben, sich zu ändern. Dem *Common Closure Principle* folgend sollten zudem zwei oder mehr Services, welche sich aus denselben Gründen ändern müssen, in einem Service zusammengefasst werden.

---

<sup>1</sup>Domain-Driven-Design: Siehe Evans and Evans [2004] für noch detailliertere Informationen des Ansatzes

<sup>2</sup>Bob Martin Patterns: Siehe <http://butunclebob.com/ArticleS.UncleBob.PrinciplesOfOod>, zuletzt gesichtet 23.08.2021

---

Abschließend muss in Zusammenhang mit Services auch ihr *Deployment* bzw. die damit verbundene Infrastruktur betrachtet werden. Wolff [vgl. 2018, S. 263 ff.] empfiehlt, für jeden Service eine eigene virtuelle Maschine (VM) zu nutzen, da hierdurch ein unabhängiges Deployment erfolgen kann. Durch die Isolation einer VM wird zudem die Stabilität und Resilienz gesteigert, da die Last und der Ausfall eines Services keine Auswirkungen auf andere Services hat. Fehler lassen sich durch eine Isolation zudem einfacher analysieren und es wird ein Deployment als vollständiges Festplatten-Image ermöglicht, in welchem die komplette Umgebung der VM auf den jeweiligen Service, sowie dessen Technologiestack bis hin zum Betriebssystem abgestimmt werden kann. So sind Services auch besser skalierbar, indem lediglich eine weitere VM gestartet und dem *Load Balancing* (siehe Abschnitt D.0.5) beigelegt wird. Im Vergleich zur Erweiterung physischer Hardware kann die Ergänzung weiterer VMs viel dynamischer und schneller erfolgen. Dies ist beispielsweise zu Testzwecken innerhalb einer *Continuous-Delivery-Pipeline* notwendig, welche häufig eine vollautomatische Inbetriebnahme neuer virtueller Maschinen benötigt. Hierfür ist der ergänzende Einsatz von Clouds sinnvoll, da diese die Erstellung neuer Instanzen mithilfe einfacher API-Aufrufe ermöglichen. Sie verteilen zudem die Last der Services, welche die Leistung eines einzigen Servers schnell übersteigen kann. Vorteilhaft ist hier auch die Nutzung eines Container-Systems wie *Docker*<sup>3</sup>, welches aufgrund der Linux-Container eine leichtgewichtige Alternative zu virtuellen Maschinen darstellt. Doch hat dies auch einige Nachteile. So basieren Docker-Container auf Linux und greifen alle auf denselben Kernel des Host-Betriebssystems zu, weshalb beispielsweise Windows-spezifische Anwendungen nicht direkt unterstützt werden. Durch die Nutzung desselben Kernels ist die Trennung der Container zudem nicht so stark wie bei virtuellen Maschinen, sodass ein Kernel-Fehler alle Container betrifft. Letztlich läuft Docker nicht nativ auf Windows oder MacOSX, sodass Docker hier selbst von einer virtuellen Umgebung umhüllt werden muss, was durch die Verschachtelung zu Zugriffsproblemen und Overhead führen kann. Als letzte Alternative ist die Nutzung des *Platform as a Service (PaaS)* Ansatzes möglich. Hierbei werden von dem PaaS standardisierte Server / Umgebungen installiert und Anwendungen über eine Versionskontrolle von dem PaaS auf den Servern ausgerollt. Die eigentliche Infrastruktur der Server / virtuellen Rechner ist vor den Anwendungen versteckt. Die Umgebung kann aufgrund ihrer Berechtigung automatische Skalierungen und ebenso automatische Deployment-Schritte vollziehen. Durch die standardisierte Umgebung wird jedoch die technische Freiheit eingeschränkt, sodass Microservices auf die davon bereitgestellten Features begrenzt sind.

## D.0.2. Datenbank

Wie bereits in Abschnitt 4.2.1 angesprochen sollte jeder Service eine eigene Datenbasis besitzen, sodass ihnen jeweils eine dedizierte Datenbank zuzuordnen ist. Da eine Transaktion mit dem Gesamtsystem jedoch auch mehrere Services umspannen kann und meist zusätzliche Message Broker zum Einsatz kommen, ist gerade die Aufrechterhaltung der Datenkonsistenz unterschiedlicher Services mit einem großen Mehraufwand

---

<sup>3</sup>Docker: Siehe <https://www.docker.com/>, zuletzt gesichtet 23.08.2021

verbunden. Diese Konsistenz ist jedoch aufgrund der Verteiltheit notwendiger Daten ausschlaggebend, weshalb spezielle Mechanismen der Datenhaltung zu verwenden sind. Richardson [vgl. 2018, S. 114 ff.] rät hier zur Nutzung des *Saga*-Patterns anstelle mehrerer manueller Transaktionen. Für Operationen, welche zur Datenmanipulation in mehreren Services führen, werden mithilfe eines asynchronen Nachrichtenaustauschs die notwendigen Schritte in den jeweiligen Services sequentiell angestoßen. Der direkt durch einen Request angesprochene Service beginnt dabei, nachdem er seine lokale Operation durchgeführt hat, asynchron ein Event über einen *Message Broker* zu versenden. Auf dieses Event hört nun der nächste Service, welcher ebenso nach seiner lokalen Operation ein Event versendet, bis alle Manipulationen dieser Kettenreaktion stattgefunden haben. Die asynchrone Kommunikation ist hierbei ein entscheidender Faktor, da sie neben der losen Kopplung auch einen Garant für die Vollendung aller notwendigen Operationen darstellt. Ist ein Service kurzzeitig nicht erreichbar, so sammelt der Message Broker alle verpassten Events und übermittelt diese beim nächsten Kontakt. Kann einer dieser Schritte fehlschlagen, so müssen alle davor durchgeführten Transaktionen eine *Roll Back* Operation anbieten, welche dann ebenso über ein Event-Messaging in umgekehrter Reihenfolge abgelaufen werden.

Die Implementierung des Saga-Pattern kann auf zwei Weisen geschehen. Als erste Variante nennt Richardson [vgl. 2018, S. 118 ff.] *koordinierte Sagas*. Dabei werden Entscheidungen und Reihenfolgen von Operationen vollständig den Services überlassen, welche über ein *Publish/Subscribe* System lediglich auf Events hören und nach der Operation selbst eines versenden. Die Vorteile dieses Ansatzes sind dessen Einfachheit und die lose Kopplung. Nachteilig sind jedoch die komplexen und unübersichtlichen Kommunikationswege bzw. verdeckten Zusammenhänge zwischen den Services. Ebenso können bei diesem Ansatz unbewusste zyklische Abhängigkeiten auftreten. Letztlich bleiben zwar Implementierungsdetails voreinander verborgen, jedoch können benötigte Absprachen zwischen Services und Teams wiederum zu stärkeren Kopplungen führen. Ein Service muss wissen, wie er auf die Events anderer Services zu reagieren hat, was eine strukturierte und stete Kommunikation der Service-Teams erfordert. Zudem könnten Änderungen eines Services auch Änderungen anderer Services verlangen.

Der zweite Ansatz zur Umsetzung des Saga-Pattern sind *orchestrierte Sagas* [vgl. Richardson, 2018, S. 121 ff.]. Dabei wird die Sequenz notwendiger Transaktionen vollständig von einer dritten Saga-Instanz angestoßen und überwacht. Jeder Service besitzt hierbei einen eigenen Channel im Message Broker (siehe Abschnitt D.0.3), über welchen er von allen Saga-Orchestratoren angesprochen werden kann. Dieser sendet nun ein Event an den ersten Service, welche nach seiner Operation über einen spezifischen, zum Saga-Orchestrator gehörenden Channel antwortet. Nachdem ein Service den Erfolg der jeweiligen Operation mitgeteilt hat, kann der Orchestrator den nächsten Service kontaktieren. Im Falle eines Misserfolgs muss auch dieser die Roll Backs anstoßen. Vorteilhaft an diesem Ansatz ist, dass lediglich einseitige Abhängigkeiten genutzt werden und Services nicht wissen müssen, welches Saga gerade eine Operation anstößt. Für die vollständige Entkopplung sollte der Orchestrator jedoch bei einer Kontaktierung auch den Namen des Orchestrator-Channels übersenden (*Return Address*), damit Services dynamisch über diesen antworten können, ohne stets eine Zuordnung vollziehen

---

zu müssen. So werden zyklische Abhängigkeiten vermieden und Services benötigen keine Kenntnisse über die Operationen anderer Services. Da für neue Sagas einfach neue Orchestratoren eingebunden werden können, ist die Kopplung weitaus geringer, die Erweiterbarkeit höher und die Business-Logik einfacher und unabhängiger implementierbar. Als einziger negativer Aspekt ist das Risiko zu sehen, dass innerhalb eines Orchestrators zu viel Logik implementiert wird. Hier ist besonders darauf zu achten, dass Orchestratoren lediglich die Sequenzierung vornehmen, die Logik aber weiterhin von dem jeweiligen Service bereitgestellt wird. Allgemein rät Richardson [vgl. 2018, S. 125] in den einfachsten Fällen zu koordinierten Sagas, nennt orchestrierte Sagas jedoch als Hauptvariante.

### D.0.3. Kommunikation

In Bezug zur Kommunikation sind neben den Client-Server Interaktionen auch jene zwischen den Services selbst zu beachten. Für diese finden sich unterschiedliche *Inter Process Communication (IPC)* Technologien, deren Betrachtung sich dieser Abschnitt widmet. Kommunikationsarten unterscheiden sich dabei zunächst auf Basis des Interaktionstypen, welcher *One-To-One* oder *One-To-Many* sein kann [vgl. Richardson, 2018, S. 67 f.]. Zudem ist zu entscheiden, ob eine *synchrone* oder *asynchrone* Kommunikation notwendig ist. Systeme bieten speziell für die Client-Server Kommunikation häufig synchrone Möglichkeiten, beispielsweise in Form von REST-Schnittstellen. Erfordert eine synchrone Anfrage von Clients nicht nur die Daten eines einzigen Services, so wird häufig auch zwischen Services eine synchrone Anfrage vollzogen. Dabei können Prozesse so lange warten, bis eine Antwort erhalten wird. Ist ein angefragter Service jedoch nicht erreichbar, so kann dieser Aufruf das System blockieren, wodurch synchrone Kommunikationen die Erreichbarkeit der gesamten Anwendung reduzieren. Richardson [vgl. 2018, S. 72 ff.] nennt dieses Pattern *Remote Procedure Invocation (RPI)*, bei welchem stets ein *Circuit Breaker* als Gegenmaßnahme eingeplant werden sollte. Dies wird mithilfe von *RPI Proxies* erreicht, welche die Anfragen an einen Service entgegennehmen, diese weiterleiten und schließlich das Ergebnis an den Anfragenden zurücksenden. Überschreitet jedoch die Anfrage an den Service ein vordefiniertes *Timeout* bzw. haben in einer Zeitspanne zu viele Anfrageversuche stattgefunden, so wird die Anfrage automatisch abgebrochen. Hierbei kann entweder ein Fehler zurückgegeben werden, ein Standardwert oder aber ein zuvor erfolgreiches Anfrageergebnis, welches in einem *Cache* zwischengespeichert wurde. Solche *Circuit Breaker* sind in einer Architektur für sämtliche synchronen Kommunikationen einzuplanen.

Die Kommunikation zwischen Services sollte jedoch vor allem eine lose Kopplung unterstützen, weshalb hier asynchrone Kommunikationen über ein *Messaging* zu präferieren sind [vgl. Richardson, 2018, S. 85]. *One-To-One Messaging* erfordert dabei einzelne *Channels*, welche speziell für die Kommunikation mit einem Service bereitgestellt sind, während das *One-To-Many Messaging* vorwiegend über ein *Publish/-Subscribe*-Pattern vollzogen wird. Ersteres ist zu nutzen, wenn mithilfe von *Command-Messages* gezielte Operationen eines Services ausgelöst werden sollen, wie es beispielsweise bei dem orchestrierten Saga-Pattern der Fall ist (siehe Abschnitt D.0.2). Für *Event-Messages* hingegen, bei welchen ein Service ohne direkte Implikation seine

---

Statusveränderung mitteilt, ist die Nutzung des Publish/Subscribe-Patterns sinnvoll, sodass eine unbestimmte Anzahl von weiteren Services diese Informationen für eigene Zwecke verarbeiten kann. Als Infrastruktur für beide Varianten ist in den meisten Architekturen ein *Message Broker* eingeplant, welcher eine zentrale Kommunikationsinstanz darstellt. So müssen Services lediglich diese Instanz kennen und sind lose gekoppelt. Zudem erhöhen Message Broker die Stabilität und Resilience des Systems, da sie für kurzzeitig abwesende Services die verpassten Nachrichten sammeln und beim nächsten Kontakt übermitteln. Wird bei solch einer Kommunikation eine Antwort der empfangenden Services (*Consumer*) erwartet, so kann der sendende Service (*Provider / Sender*) im *Header* der Nachricht den Namen des Antwort-Channels mitsenden, was *Publish/Async Response* genannt wird. Um über den Message Broker gezielte Service-Channels nutzen zu können, ohne dass sich Teams dafür stark absprechen müssen, wird als Grundlage des Channel-Namens häufig der Name der Domänen-Klasse genutzt.

Die asynchrone Kommunikation über einen Message Broker weist, neben den bereits genannten Vorteilen der Nachrichten-Speicher und Abstraktionen von Service-Adressen, auch wichtige Nachteile auf [vgl. Richardson, 2018, S. 93 f.]. So können sie ein *Bottleneck* der Anwendung darstellen, wenn sämtliche Kommunikation darüber stattfindet, wodurch sie ebenfalls als *Single-Point-Of-Failure* die Zuverlässigkeit des Systems beeinflussen. Ihr Einsatz sollte auf logische Abschnitte oder Anwendungsbereiche des Systems begrenzt werden, um die Last so zu verteilen. Zudem sollten Message Broker mit einer hohen Skalierbarkeit und Erreichbarkeit ausgewählt werden. Der zweite Nachteil befasst sich mit der zusätzlichen Komplexität durch solch einen Message Broker. Er stellt einen zusätzlichen Systembestandteil dar, welcher installiert, konfiguriert, verwaltet und in allen Services integriert werden muss. So sind bei dessen Wahl zusätzliche Aspekte zu beachten [vgl. Richardson, 2018, S. 92]. Aufgrund der gewünschten Technologiefreiheit von Microservices ist eine Unterstützung vieler Programmiersprachen anzustreben. Dabei sind auch unterstützte Messaging-Standards zu beachten, sodass eine schnelle Adaption erfolgen kann. Die Nachrichten-Reihenfolge ist eine häufige Fehlerquelle, sodass beachtet werden muss, ob ein Broker die Reihenfolge beibehält oder variiert. Letztlich ist auch speziell die Latenz entscheidend. In einer Microservice-Architektur befinden sich viele einzelne Instanzen, welche miteinander kommunizieren müssen. Dabei können zeitkritische Operationen auftreten, welche durch eine erhöhte Latenz gefährdet werden.

#### **D.0.4. Service Discovery**

Während ein Message Broker die Netzwerk-Adressen der teilnehmenden Services abstrahiert, müssen diese bei synchronen Kommunikationen bekannt sein. Dies ist in einer Microservice-Architektur problematisch, in welcher dynamisch neue und unabhängige Services hinzugefügt werden können. Es ist jedoch auch ein allgemeines Skalierungsproblem, da zur Kompensierung hoher Lasten, Ausfälle oder Updates dynamische Replikationen bestehender Server zum Einsatz kommen. Besonders in modernen cloud-basierten Anwendungen erhalten solche Service-Instanzen oft dynamische und sich ändernde Netzwerk-Adressen, weshalb Richardson [vgl. 2018, S. 80 f.] die Nutzung eines

---

*Service Discovery* Mechanismus empfiehlt. Die zentrale Einheit solcher Mechanismen stellt eine *Service Registry* dar, also ein eigener Service, der die Adressen aller Instanzen innerhalb einer Datenbank oder einfachen Tabelle sammelt und aktualisiert. Vollziehen Clients nun Anfragen, so werden von der Service Discovery aus der Service Registry die zugehörigen Services identifiziert und Anfragen weitergeleitet. Solche eine Service Discovery kann entweder auf der Client- / Service-Seite implementiert werden (*Application-Level Service Discovery*), sodass diese direkt mit der Registry kommunizieren, oder aber von einer zentralen Deployment-Infrastruktur übernommen werden (*Platform-Provided Service Discovery*).

Bei der *Application-Level Service Discovery* findet eine *Self Registration* von Services statt, in welcher sie der Service Registry ihre Netzwerkadresse und zusätzliche Metainformationen des Services übermitteln [vgl. Richardson, 2018, S. 81 ff.]. Einige etablierte Service Registries erfordern als zusätzliche Information die URL zu einem *Health Check* des jeweiligen Services, über welchen in regelmäßigen Abständen die Verfügbarkeit kontrolliert wird. Als alternativen Mechanismus zur Schaffung von Stabilität und Resilience fordern einige Service Registries einen regelmäßigen *Heartbeat* von registrierten Services, über welchen sie wiederum ihre Verfügbarkeit mitteilen und so in der Registry vermerkt bleiben. Clients führen bei Anfragen eigenständig eine *Client-Side Discovery* durch, erhalten eine Liste erreichbarer Services und entscheiden selber, welchen davon sie kontaktieren müssen. Somit ist auch das *Load Balancing* dem Client überlassen, weshalb dieses innerhalb jedes Clients und jedes aktiv kommunizierenden Services zu integrieren ist. Verwendete Service Discovery und Load Balancing Bibliotheken sollten deshalb in vielen Sprachen vorliegen. Diese Alternative stellt jedoch einen hohen Mehraufwand für die Clients und Services dar, weshalb meist eine *Platform-Provided Service Discovery* eingesetzt wird [vgl. Richardson, 2018, S. 83 ff.]. Viele etablierte Deployment-Plattformen wie Docker oder Kubernetes<sup>4</sup> stellen solche Discovery-Mechanismen bereit (*Server-Side Discovery*), welche jedem registrierten Service einen DNS-Namen und / oder eine virtuelle IP (VIO) Adresse zuordnet. Über diese können Clients nun Anfragen stellen und werden von der Plattform mit einer der zugehörigen Service-Instanzen verbunden. Hierbei findet häufig eine *3rd Party Registration* statt, in welcher eine dritte Instanz anstelle der Services selbst die Registrierung vornimmt. Dieser Ansatz hat den Vorteil, dass sowohl Service-Registrierung als auch Service-Discovery und Load Balancing von einem zentralen System übernommen werden, sodass diese unabhängig von der Programmiersprache eines Clients oder Services nutzbar sind. Als Nachteil gilt jedoch die Einschränkung der Discovery- und Balancing-Möglichkeiten auf die von der genutzten Plattform angebotenen Alternativen.

Wolff [vgl. 2018, S. 147] nennt eine weitere Einschränkung des letzten Ansatzes. Zentrale Service Discoveries mit integriertem Load Balancing stellen wiederum ein *Bottleneck* dar, da jegliche Kommunikation über diese geleitet wird. Ausfälle des zentralen Services führen dann zum Gesamtausfall des Systems. Auch ist die Konfiguration eines zentralen Load Balancers aufwändiger, da dieser auf jegliche Ausnahme- und Sonderfälle der Services optimiert werden muss, was wiederum eine starke zentrale Abhängigkeit dar-

---

<sup>4</sup>Kubernetes: Siehe <https://kubernetes.io/>, zuletzt gesichtet 23.08.2021



stellt. Deshalb sollte für jeden Service ein eigener Load Balancer zum Einsatz kommen, beispielsweise innerhalb des *RPI Proxies* (siehe Abschnitt D.0.3), welcher auf die dynamischen Anforderungen der jeweiligen Instanzen optimiert ist. Fällt dieser aus, so ist lediglich die zugehörige Service-Familie davon betroffen. Da jedoch die Last auf viele Balancer aufgeteilt ist, erfolgen Ausfälle zudem mit geringerer Wahrscheinlichkeit.

### D.0.5. API Gateway

Durch eine Service Discovery erhalten auch externe Clients einen einfachen Zugang zu den bereitgestellten Services. So können diese alle Informationen erhalten, welche sie für eigene Operationen oder die Anzeige von visuellen Elementen benötigen. Die Daten hierfür sind jedoch weiterhin über mehrere Services verteilt, weshalb Clients eine *API Composition* vollziehen müssen. Diese beschreibt Richardson [vgl. 2018, S. 221 ff.] als Praktik, bei der ein Client nacheinander Anfragen an die APIs verschiedener Services stellt, um so alle benötigten Daten zu erhalten und diese lokal zu kombinieren. Da diese Anfragen jedoch fest im Client integriert sind und so eine Kapselung der Komposition fehlt, sind folgende Änderungen von APIs oder allgemeinen Zuständigkeitsbereichen der Services mit viel Mehraufwand verbunden. Zudem können einzelne Services unterschiedliche Kommunikationsprotokolle nutzen, welche nicht für die Kommunikation an externe Instanzen geeignet sind oder die Komposition von Daten sehr komplex gestalten. Solche Probleme können mithilfe des *API Gateway* Pattern beseitigt werden [vgl. Richardson, 2018, S. 259 ff.]. Dabei wird ein Service als Einstiegspunkt in das System angeboten. Dieses kapselt die interne Architektur und bietet eine eigene, umfassende und kombinierte API für externe Clients. Das Gateway kann, ähnlich einem *Reverse Proxy*, einzelne Anfragen an den jeweiligen Service weiterleiten, jedoch auch im Sinne der *API Composition* eigene Routen anbieten, in denen es die Daten mehrerer Services anfragt und aggregiert. So muss ein externer Client lediglich eine Anfrage stellen, um alle notwendigen Daten zu erhalten. So ersetzt ein Gateway aus Client-Sicht die Service Discovery, da diese intern von dem Gateway verwendet wird. Da das Gateway ein Bindeglied zwischen Clients und Services darstellt, sollte hier auch die Übersetzung der Kommunikationsprotokolle stattfinden, sodass Clients ein standardisierter Kommunikationsweg ermöglicht wird, beispielsweise in Form einer REST-Schnittstelle. Ein API Gateway kann zudem weitere Funktionen übernehmen. So ist es üblich, hier eine *Authentifizierung* und *Autorisierung* zu vollziehen. Ebenso ist ein *Rate Limiting* und ein *Caching* häufig angefragter Daten möglich. Für statistische Auswertungen können im Zuge eines *Monitorings* zudem Nutzungs- und Anfragedaten aufgezeichnet werden.

Da unterschiedliche Client-Typen auch unterschiedliche Anforderungen besitzen und ebenso variierende Daten von dem Gateway benötigen können, ist eine *One-Size-Fits-All (OSFA)* Lösung nicht immer sinnvoll. Sind unterschiedliche Anforderungen präsent, sollte ein Gateway auch unterschiedliche APIs anbieten, beispielsweise für Android Geräte, IOS Geräte und Desktop-PCs. Dies kann entweder innerhalb desselben Gateways geschehen, oder aber über dedizierte Gateways erfolgen, was als *Backend For Frontend* bekannt ist [vgl. Richardson, 2018, S. 262]. Letzteres ermöglicht es, das Gateway-Design an dedizierte Frontend Teams auszulagern, wodurch auf Team-Ebene eine bessere Separation der Zuständigkeitsbereiche (*Separation Of Concerns*) gegeben

ist. Als Nachteil gilt jedoch der größere Aufwand, da weitere hochgradig verfügbare Komponenten entwickelt, veröffentlicht und aktualisiert werden müssen. Aktualisierungen sind zudem von allen Services abhängig, deren Operationen auch externen Instanzen bereitgestellt werden müssen. Somit finden diese häufig statt und das dynamische Einfügen neuer Services wird erschwert. Letztlich sind Gateways mit denselben Problematiken verbunden wie Service-Discoveries, da sie als zentrale Komponente ein *Bottleneck* darstellen.

#### D.0.6. Micro Frontend

Services in einer Microservice-Architektur sollen lose gekoppelt, individuell verwaltet und unabhängig veröffentlicht werden. Die Daten dieser Services müssen jedoch auch für einen Nutzer visualisiert werden, was wiederum Probleme aufwirft. Wird lediglich ein Frontend erzeugt, welches manuell auf alle Services zugreift und dafür visuelle Komponenten definiert, so stellt dieses wiederum eine stärkere Abhängigkeit dar und muss unter Umständen bei jedem Update eines Services angepasst werden. Um dies zu umgehen und die Zuständigkeitsbereiche beizubehalten, sollte jeder Service eine eigene UI anbieten, was als *Micro Frontends* bekannt ist. Hierdurch sind dieselben Teams, welche für die Services zuständig sind, auch mit der Aktualisierung dedizierter Visualisierungen beauftragt. Diese Teil-UIs müssen jedoch wieder in einer Gesamtanwendung gesammelt werden, wozu es unterschiedliche Ansätze gibt [Wolff, 2018, S. 167 ff.].

Als Erstes könnte für jeden Service eine vollständige *Single Page Application (SPA)* erstellt werden, welche sich dann für eine gegenseitige Integration verlinken. Dies ist vorteilhaft, wenn die Services komplexe Interaktionen oder offline Fähigkeiten besitzen müssen bzw. immer eine vollständige Seitendarstellung übernehmen sollen. Dadurch sind diese vollständig isoliert, was jedoch eine übergreifende Interaktion erschwert. Der stetige Wechsel zwischen Webseiten führt zudem zu langen Ladezeiten. Es ist bei diesem Ansatz zudem darauf zu achten, dass die SPAs eine einheitliche Authentifizierung und Autorisierung integrieren, damit wiederholte Logins vermieden werden.

In vielen Fällen besteht eine Webpage jedoch aus mehreren Abschnitten, welche von unterschiedlichen Services bereitgestellt werden. Hier sind vollständig isolierte SPAs nicht nutzbar. Stattdessen sollten *SPA-Module* zum Einsatz kommen, wobei jeder Service eines dieser Module bereitstellt. Die Module selbst werden dann zentral auf einem Server gesammelt und sind unabhängig von dem jeweiligen Service-Team verwaltbar. Da hier jedoch wieder eine zentrale Instanz als Anlaufpunkt mehrerer Teams vorhanden ist, führt dies wiederum zu einer stärkeren Kopplung und Abhängigkeit. Die SPA muss bei dem Update eines Moduls stets neu gebaut werden, da ein Deployment meist nur als vollständige Anwendung möglich ist [vgl. Wolff, 2018, S. 171]. Somit wird eine Koordination unterschiedlicher Teams benötigt. Ebenso können Änderungen eines Moduls zu Änderungen anderer Module führen. Bei beiden SPA-Ansätze werden zudem für ein einheitliches Design entweder ebenso einheitliche Design-Richtlinien, oder aber ein zentraler *Asset-Server* benötigt, welcher wiederverwendbare Code-Abschnitte und Ressourcen bereitstellt. Für externe Nutzer sollten die Microservices zudem

wie eine einzige Webanwendung erscheinen und somit auch unter einer gemeinsamen URL erreichbar sein, wodurch Probleme wie die *Same Origin Policy* umgangen werden.

Eine beliebtere Alternative stellt die Zusammenstellung einer Webseite aus HTML-Komponenten bzw. Fragmenten dar. Auch hierfür finden sich unterschiedliche Ansätze wie *Edge Side Includes*, *Server Side Includes* oder *Java Portlets* [vgl. Wolff, 2018, S. 174 ff.]. Jackson [2019] beschreibt ebenfalls verschiedene, etablierte Praktiken. Bei *Server-Side Template Composition* wird ein zentraler *Frontend-Server* geschaffen, welcher nach der Anfrage eines Clients die notwendigen HTML-Fragmente von den Services anfragt, dann die Gesamtseite rendert und abschließend ausliefert. Eine *Build-Time Integration* hingegen bedingt, dass Micro Frontends als Package ausgeliefert und von einer Container-Anwendung als Bibliothek integriert werden. Das Problem solcher *Build-Time* Ansätze ist jedoch, dass bei der Änderung eines Packages auch jedes andere Micro Frontend neu kompiliert und freigegeben werden muss. Attraktiver sind Lösungen, welche eine Integration zur Laufzeit ermöglichen. Ein Ansatz hierfür ist die Verwendung von *IFrames*, mithilfe derer die UI eines Services über dessen URL angezeigt werden kann. So kann eine Seite einfach in Unterabschnitte eingeteilt werden und der zentrale Frontend-Server ist lediglich für das Skelett der Anwendung, sowie für das Routing zuständig. Dadurch können Inhalte auch dynamisch ergänzt oder ausgetauscht werden. Die starke Isolation erschwert jedoch die übergreifende Kommunikation, welche dann meist über ein Event-Messaging auf Service-Ebene vollzogen wird. Viel flexibler ist die Nutzung von Javascript zur Laufzeit-Integration. Dabei werden Micro Services im HTML als *script*-Element hinterlegt, was einer Bibliotheksintegration ähnelt und auch dynamisch durch Javascript ermöglicht wird. Die Micro Frontends selbst stellen als Einstiegspunkt eine globale Funktion bereit, welche von dem Frontend-Server genutzt werden kann, um diese Komponente dynamisch in einem Bereich zu rendern. Im Gegensatz zu IFrames haben diese Micro Frontends durch den gemeinsamen Kontext des Aufrufs flexible Interaktionsmöglichkeiten. Zudem können den Funktionen kontextbedingte Parameter übergeben werden, um dynamische Konfigurationen vorzunehmen. Jackson [2019] nennt dies als meistgenutzte Alternative. Hier finden sich auch Variationen der Umsetzung. So können anstelle globaler Funktionen auch *Web Components*<sup>5</sup> genutzt werden, welche dieselbe Funktionalität über benutzerdefinierte HTML-Elemente ermöglichen.

### D.0.7. Sicherheit

Sicherheit ist auch bei Microservices ein entscheidendes Thema, nicht zuletzt, da dynamische Erweiterungen von Services ermöglicht werden sollen. Im Gegensatz zu monolithischen Architekturen können jedoch keine zentralen Sessions oder In-Memory Sicherheitskontexte genutzt werden, da sich diese lokal auf einen Service beschränken. Nun könnte jeder Service eine Authentifizierung des anfragenden Nutzers vollziehen, jedoch würden sie dafür einen Zugriff auf das interne Netzwerk benötigen. Die Beachtung von Sicherheitsaspekten würde dann zusätzlichen Mehraufwand für jedes Entwicklerteam darstellen. Wie in Abschnitt D.0.5 beschrieben, sollte deshalb ein

---

<sup>5</sup>Web Components: Siehe [https://developer.mozilla.org/en-US/docs/Web/Web\\_Components](https://developer.mozilla.org/en-US/docs/Web/Web_Components), zuletzt gesichtet 23.08.2021

---

zentrales Gateway die Authentifizierung übernehmen. Da es den Einstiegspunkt aller externen Anfragen darstellt, muss die Funktionalität nur einmal bereitgestellt werden, wodurch Sicherheitslücken reduziert werden. Clients authentifizieren sich nun entweder bei jeder Anfrage erneut beim Gateway oder erhalten von diesem nach der ersten Authentifizierung ein *Session-Token*. Nun müssen jedoch auch die Services wissen, dass eine Anfrage authentifiziert wurde und wer diese anstieß. Für authentifizierte Nutzer ergänzt das Gateway deshalb die Anfragen an Services um ein *Access Token*, welches beispielsweise die Identität und Rolle des Nutzers beinhaltet [vgl. Richardson, 2018, S. 354 f.]. Nun kann eine Autorisierung durch die einzelnen Services erfolgen. Diese könnte auch von dem Gateway durchgeführt werden, würde dann jedoch wieder eine starke Kopplung darstellen.

Für die Umsetzung eines *Access Tokens* können unterschiedliche Token-Typen verwendet werden [vgl. Richardson, 2018, S. 356 ff.]. *Opaque Tokens* verschleiern ihren Inhalt beispielsweise als *UUID* und erfordern von Services, dass diese mithilfe des Tokens einen zentralen *Security Server* für weitere Nutzerdetails synchron anfragen. Sie führen deshalb zu einer reduzierten Performanz und Erreichbarkeit, sowie zu einer höheren Latenz. *Transparent Tokens* stellen die präferierte Alternative dar, da sie die Nutzerdaten direkt bereitstellen. Hierfür ist die Nutzung von *JSON Web Tokens (JWTs)* ein verbreiteter Standard, welche im *Body* neben der Identität und Rolle eines Nutzers auch ein Verfallsdatum (*Expiration Date*) beinhalten. Es ist mithilfe von *Secrets* signiert und verschlüsselt, welche lediglich dem Gateway und empfangenden Services bekannt sind. Um einem Missbrauch abgefangener JWTs vorzuzugreifen, ist ein kurzes Verfallsdatum zu wählen, welches jedoch erhöhten Aufwand für Aktualisierungen bzw. Erneuerungen der Tokens hervorruft. Deshalb gibt es den Sicherheitsstandard *OAuth 2.0*, welcher folgende Bestandteile beinhaltet:

- **Authentication Server:** Ein zentraler Server, welcher die Authentifizierung übernimmt und ein *Access Token*, sowie *Refresh Token* generiert.
- **Access Token:** Ein Token, welches für den Zugang zu einem *Resource Server* benötigt wird.
- **Refresh Token:** Ein Token, welches für die Erzeugung neuer *Access Tokens* eines Nutzers notwendig ist.
- **Resource Server:** Sämtliche Services einer Microservice-Architektur, welche ein *Access Token* für die Autorisierung eines Nutzers verwenden.
- **Client:** Ein Client, welcher auf einen *Resource Server* zugreifen möchte. Innerhalb einer Microservice-Architektur ist dies immer das *API Gateway*.

Die Verwendung des *OAuth 2.0*-Standards sieht nun wie folgt aus. Ein Nutzer löst einen Request an das Gateway aus und stellt dabei Login-Daten bereit. Das Gateway kontaktiert den *Authentication Server* mithilfe eines *OAuth 2.0 Password Grant Request*. Der Server validiert die Anfrage und antwortet im Erfolgsfall mit dem *Access* und *Refresh Token*, welches das Gateway nun der Anfrage an den tatsächlichen Service (*Resource Server*) beifügt. Dieser validiert das Token und nutzt es für weitere

Autorisierungs-Prozesse. Das Gateway kann alternativ auch eine *Login-Route* anbieten und die erhaltenen Tokens dem Client übergeben. Von diesem können sie dann beispielsweise als Cookie gespeichert und folgenden Anfragen in Form eines *Session Tokens* angehängen werden. Bei der Planung einer Microservice-Architektur ist also auch solch ein Authentication Server, sowie dessen Kommunikation mit dem Gateway und ggf. den Services einzuplanen.