

Torben Spieker

Eine Untersuchung des
impliziten Menschenbilds in
Kollaborationssoftware und
dessen Entwicklung



Hamburger Papiere zur Designtheorie
und -forschung an der HFBK Hamburg

DENKEN ÜBER DESIGN

Torben Spieker

Eine Untersuchung des
impliziten Menschenbilds in
Kollaborationssoftware und
dessen Entwicklung

Bachelor-Arbeit im Studienschwerpunkt Design

Betreuende Professoren:
Jesko Fezer, Dr. Friedrich von Borries

Reihenherausgeber:
Jesko Fezer, Dr. Friedrich von Borries
Gestaltungskonzept: Friederike Wolf
Umsetzung: Torben Spieker
Korrekturen: Marie-Theres Böhmker
Druck: Scharlau

material 383-14

Materialverlag
der Hochschule für Bildende Künste Hamburg
2019

Hamburger Papiere zur Designtheorie
und -forschung an der HFBK Hamburg

Inhalt

6	Einleitung	16	Örtlichkeit
8	Plattformen	18	Zeitlichkeit
10	Softwareentwicklung	21	Körperlichkeit
		24	Hierarchie
		27	Fazit

Einleitung

Im Kontext eines zunehmenden Wandels der Arbeitswelt hin zu fluideren Arbeitszusammenhängen und „remote“-Kooperation, also einer Form der Kooperation, in der die Beteiligten nicht oder nicht überwiegend von dem selben physischen Ort aus tätig sind, wächst der Bedarf nach Hilfsmitteln, welche die damit wegfallenden Funktionen realweltlichen Zusammentreffens ersetzen können. Eine besondere Bedeutung kommt dabei einer angemessenen Abbildung von Kommunikation als essenziellen Aspekt menschlicher Interaktion zu.

Zwar wurde der Bedarf nach ortsungebundener Kommunikation schon verschiedentlich beantwortet, doch werden bisherige Lösungen den höheren Anforderungen einer zielorientierten Verständigung in komplexeren Gruppen nicht gerecht. So sind etwa Telefon und Instant Messenger wie ICQ oder WhatsApp auf eine eins-zu-eins- oder Kleingruppenkommunikation zugeschnitten, E-Mail sowie Internetforen aufgrund ihrer Asynchronität nicht hinreichend für einen flüssigen Austausch geeignet und Chat-Systeme wie IRC und XMPP angesichts ihrer technischen Hürden einer breiten Masse an Nutzern unzugänglich, da sie Kenntnisse erfordern, die über die Installation von Applikationen und eine minimale Konfiguration hinausgehen. Diese bisherigen Lösungen könnten zwar in Kombination eine ausreichende Kommunikation ermöglichen, doch leidet unter dem parallelen Einsatz mehrerer Kommunikationssysteme die Übersichtlichkeit und Auffindbarkeit von Informationen.

So ist in den letzten Jahren unter dem Begriff der „Kollaborationssoftware“ eine neue Klasse digitaler Kommunikationsplattformen entstanden, deren Vertreter angeben, auf die Bedürfnisse kollektiver Arbeitsprozesse und die Abbildung von Merkmalen diverser Gruppen zugeschnitten zu sein, indem sie verschiedene Austauschprinzipien in einer Oberfläche vereinen und diese mit einer Vielzahl kooperationspezifischer Funktionen erweitern. So wecken diese Anwendungen die Erwartung, dass sie in ihrer Konzeption auf Probleme der digitalen Kommunikation hinsichtlich der Abwesenheit physischer Treffen angemessen reagieren und bisherigen digitalen Kommunikationsformen in diesem Aspekt deutlich überlegen sind.

Diese Erwartung hat sich aber in meiner Erfahrung in kollektiven Arbeitsprozessen bei Einsatz solcher Kollaborationsplattformen bisher nur Ansatzweise erfüllt. Oft wird funktionale Zusammenarbeit erst durch gelegentliche Realwelttreffen möglich oder bleibt bei Unmöglichkeit solcher Treffen dysfunktional. Dabei scheint Erfolg und Misserfolg weniger von der jeweiligen Natur des Projekts, sondern mehr von den persönlichen Charakteristiken einzelner Teilnehmer:innen abzuhängen, da regelmäßig die selben Personen, ungeachtet derer Aufgeschlossenheit gegenüber der Kommunikationsmittel, ein unzureichendes Kommunikationsverhalten an den Tag legen.

Dies führt mich zu der Annahme, dass existierende Kollaborationsanwendungen Gestaltungsmerkmale aufweisen, die deren Versprechen einer egalitären Verwendbarkeit, ungeachtet der jeweiligen Kommunikationsteilnehmer:innen, entgegen stehen und vielmehr durch ihre Funktionsmechanismen bestimmte Menschen bevorzugen. Darum möchte ich in dieser Arbeit anhand dreier Beispiele solcher Kollaborationsanwendungen untersuchen, ob – und wenn ja inwiefern – eine Bevorzugung bestimmter Charaktermerkmale in die Anwendungen eingeschrieben ist, und damit ob der Entwicklung der Anwendungen ein bestimmtes Menschenbild zugrunde liegt.

Dabei konzentriere ich mich in der Analyse auf den Kernbereich der Kommunikationsmechaniken und weniger auf die erweiterten Produktivitätsfunktionen. Eine erste Betrachtung der Anwendungen hat gezeigt, dass sie sich in diesem Kernbereich stark ähneln. Darum werde ich nach einer kurzen Vorstellung der Plattformen zunächst auf Umstände in der Softwareentwicklung eingehen, die diese Ähnlichkeit verursachen und die grundlegenden Paradigmen herausstellen. Danach werde ich diese Paradigmen anhand der Aspekte von Örtlichkeit, Zeitlichkeit, Körperlichkeit und Hierarchie betrachten und eventuelles Abweichen von den Paradigmen in den einzelnen Kollaborationsplattformen herausstellen.

Plattformen

Slack

Slack kann als Begründer der hier zu behandelnden Klasse von Kollaborationsplattformen betrachtet werden. Zwar ist nicht auszuschließen, dass einige Mitbewerber intern schon vor Slacks Veröffentlichung¹ (2013) ähnliche Entwicklungen betrieben haben, doch hat Slack als Erstes einen Massenmarkt erreicht, damit eine Art Standard gesetzt und sich bis heute als Marktführer etabliert². Slacks Ikonizität geht soweit, dass Konkurrenzprodukte üblicherweise abgrenzend als „so etwas wie Slack, aber...“ definiert werden. Durch seine Marktmacht sowie das Vertriebsmodell, welches im Gegensatz zu den beiden anderen betrachteten Plattformen auch eine kostenpflichtige Version beinhaltet, haben die Anbieter von Slack die größten Entwicklungskapazitäten und bieten so den größten Funktionsumfang.

Discord

Discord ist eine Kommunikationsplattform, deren Ursprung im Gaming-Bereich liegt. Zu Beginn hat sie ihren Hauptfokus auf Sprachchatfunktionalität gelegt und sich so als Konkurrent von Teamspeak oder Skype positioniert³. Die eingebauten Funktionen, mit denen sich etwa sogenannte „Raids“, also Kämpfe in Onlinerollenspielen mit quasi-militärischen Befehlsstrukturen, effizient organisieren lassen, haben sich aber auch für andere Organisationen als nützlich erwiesen. Dadurch hat sich Discords Funktionsumfang mehr und mehr in Richtung der von Slack etablierten Muster der Onlinekollaborationsplattformen verschoben. Zweifelhafte Bekanntheit erlangte Discord im deutschsprachigen Raum als Organisationswerkzeug für das rechtsextreme Onlinekampagnennetzwerk „Reconquista Germanica“⁴.

Riot

Riot ist eine Kollaborationssoftware, die auf dem „Matrix“-Protokoll basiert und mit diesem eine funktionale Einheit bildet. Das Matrix-Protokoll definiert auf technischer Ebene den Informationsaustausch. Riot ist eine Referenzimplementation dieses Protokolls, hergestellt durch den selben Anbieter, also eine Oberfläche, die möglichst alle Funktionen des Protokolls nutzbar macht. Sowohl das Matrix-Protokoll als auch Riot werden quelloffen entwickelt und unter Open-Source-Lizenz vertrieben. So steht es auch anderen Entwickler:innen frei, in der Entwicklung beizutragen oder andere Oberflächen für das Matrix-Protokoll anzubieten. Da im Gegensatz zu Slack und Discord bei Riot/Matrix der Quellcode eingesehen werden kann, eigene Server zur Verteilung von Nachrichten verwendet werden können und die Option besteht ende-zu-ende-verschlüsselt zu kommunizieren, ist Riot vor allem in datenschutzsensiblen Anwender:innenkreisen beliebt.

Genese

Ein Kennzeichen von Software, im Vergleich zu physischen Produkten, ist deren ephemerer Charakter. Vereinfacht gesagt besteht Software aus einer Reihe von Anweisungen an elektronische Komponenten, also einer systematisch festgeschriebenen Aneinanderreihung von Entscheidungen und Gedanken ohne stoffliche Komponente. Das ausgelieferte Produkt ist jeweils ein bloßes Abbild dieser Aneinanderreihungen zu einem bestimmten Zeitpunkt und sieht man jedes Update als neues Produkt – was berechtigt ist, da jedes Update einen geänderten Funktionsumfang mit sich bringt – so ist die Lebensdauer eines Softwareprodukts nach unten hin nur durch die Geschwindigkeit der jeweiligen Update-Mechanismen begrenzt und die technischen so wie zeitlichen Kosten der Bereitstellung einer neuen Produktversion gehen gegen null, so lange keine Komponenten außerhalb der Software angepasst werden müssen.

Diese Vereinfachung der Bereitstellung im Vergleich zu physischen Objekten resultiert in einer veränderten Produktgenese. Zwar gibt es auch im Softwarebereich Projekte, die dem klassischen Modell der Trennung von Konzeptions- und Fertigungsphase⁵, mit einer gewissen Trennungsunschärfe im Bereich der Prototypenentwicklung, folgen. In vielen Softwareprojekten weitet sich die Unschärfe aber zu einer kompletten Parallelität von Konzeption und Ausführung aus. Das weit verbreitete⁶ Konzept des „Agile Development“ propagiert etwa diese Parallelität unter der Methode des „Incremental Development“: „[...] this means that each successive version of the product is usable, and each builds upon the previous version [...]“⁷. Sobald nur minimale, überprüfbare Annahmen zur Entwicklung des grob skizzierten Endprodukts getroffen wurden, beginnt die Umsetzung und die konkrete Feinkonzeption reicht oft nur zum nächsten „minimum viable product“⁸, der nächstgrößeren „lebensfähigen“ Programmversion, die neue Erkenntnisse zur Weiterentwicklung des Programms bieten kann. So ist jede Programmversion sowohl ein Produkt, als auch der Prototyp der jeweils nächsten Version.

Im Extremfall dieses inkrementellen Ansatzes läuft die Entwicklung

eines Softwareprodukts über längere Zeit gar ohne größeren konzeptionellen Rahmen ab. So ist es insbesondere im Umgang mit Unix-Betriebssystemen (beispielsweise Linux) üblich, mittels Kleinstprogrammen, sogenannten „Scripts“, die oft nur wenige Zeilen Code enthalten, einfache Funktionen einzusetzen oder Abläufe zu automatisieren. Diese Scripts werden dann gegebenenfalls im persönlichen Umfeld verteilt, in mächtigere Programmiersprachen übersetzt und langsam aber stetig mit Funktionen erweitert, bis sie für eine größere Zielgruppe interessant und nun überhaupt erst als Produkt wahrgenommen werden. In meiner Erfahrung findet oft erst zu diesem Zeitpunkt eine Bewusstseinsmachung der Autor:innen als Produktentwickler:innen statt und damit auch erst zu diesem Zeitpunkt eine Kontextualisierung der Produktentwicklung im Bezug auf eine diversere Benutzer:innengruppe.

Je mehr sich eine jeweilige Produktgenese diesem Extrem annähert, also je weniger der oder die jeweilige Softwareentwickler:in sich überhaupt als Planer:in begreift, desto wahrscheinlicher wird es, dass in unserer Kultur angesammeltes Wissen über gute Planung keine Anwendung findet und desto wahrscheinlicher wird es, dass eine Prozesskritik entlang planungstheoretischer Ansätze ins Leere läuft. So geht etwa Walter Schönwandt in „Denkfallen beim Planen“ auf Probleme der selektiven Wahrnehmung beim Zusammentragen von Informationen ein⁹, doch setzt eine Kritik entlang dieser Probleme voraus, dass in der Konzeption überhaupt ein nennenswertes Zusammentragen von Informationen im Hinblick auf die spätere Nutzer:innengruppe stattfindet.

Ein:e Entwickler:in, die zunächst nur sich oder beispielsweise einen kleinen Kreis von Kolleg:innen als potenzielle Zielgruppe ansieht, wird berechtigterweise eine Software entwickeln, die für diese Nutzer:innen gut funktioniert und dabei bewusste und unbewusste Entscheidungen in den Quellcode einschreiben, die diese Nutzer:innen bevorzugt und dadurch wiederum andere Nutzer:innen diskriminiert. Eine Software, die aus einem inkrementellen Prozess heraus in eine größere Öffentlichkeit getragen wird, wird immer Teile dieser diskriminierenden Aspekte beibehalten.

Diese inkrementelle Vorgehensweise ist besonders in der Entwicklung von Slack zu beobachten. Slack entstand als Seitenprojekt einer Spielefirma zur besseren internen Kommunikation, als Ersatz für den Austausch über Mailinglisten und sonstigem Wildwuchs verschiedener Kommunikationskanäle¹⁰. Die Entwickler:innen haben sich dabei der Funktionen des „Internet Relay Chat“ (IRC) bedient und diese nach und nach mit eigenen Funktionen erweitert.

IRC ist ein 1988 veröffentlichtes Protokoll zur Nachrichtenübermittlung, das noch heute, vor allem in der Techszene, verbreitet ist und als ein

Urvater der digitalen Kommunikation bezeichnet werden kann¹¹. Grundlegende Paradigmen von IRC sind der überwiegend synchrone textbasierte Austausch mit zeitlich aufsteigender Visualisierung vergangener Nachrichten und die Organisation in sogenannte „Channels“, also inhaltlich getrennte Abfolgen von Nachrichten, gegebenenfalls mit jeweils verschiedenen Rezipient:innen pro Channel. Somit ist IRC ein Prototyp der Kommunikationsform welche heute unter dem Begriff des „chattens“ gefasst wird. Diese Paradigmen sind bis heute in Slack eingeschrieben und machen das Gros der dortigen Kommunikation aus. Sie wurden also bei der Wandlung Slacks von internem Werkzeug zu öffentlich vertriebenem Produkt nicht verändert.

Auch Discord ist laut Aussagen der Entwickler:innen als ein „skunkworks project“ einer Spieleentwicklungsfirma, also als ein experimentelles internes Projekt gestartet worden, jedoch war es von Beginn an als Reaktion auf einen empfundenen Mangel im Ökosystem der verfügbaren Software für die Sprachkommunikation von Computerspielenden ausgerichtet¹². So ist anzunehmen, dass die Funktionen, die diesen Anwendungsfall betreffen, bewusst für die Zielgruppe der Computerspielenden konzipiert wurden. Discord bietet neben diesen spezialisierten Funktionen aber auch, ähnlich zu Slack, IRC-artige Möglichkeiten des Textaustausches, wodurch Discord auch außerhalb der Computerspielszene und abseits der Nutzung der Sprachkommunikation angenommen wird¹³. Mittlerweile hinzugefügte Funktionen belegen, dass die Discord-Entwickler:innen ihre Software auch in Richtung dieser neuen Nutzer:innengruppe vorantreiben¹⁴. Diese hinzukommenden Funktionen, etwa der automatisierte Abruf von Daten aus Projektmanagementsystemen, verändern allerdings keine der Kernmechanismen zum Informationsaustausch und so bleibt auch hier eine tiefer gehende Anpassung an zukommende Nutzer:innentypen aus.

Design patterns

Im Gegensatz zu Slack und mehr noch als bei Discord ist bei Riot davon auszugehen, dass, aufgrund der oben angesprochenen Client-Protokoll-Dualität und Open-Source Natur, eine ausgiebige Konzeption der Software vor der Übersetzung in ausführbaren Code stattgefunden hat. Die Entwicklung eines Protokolls, auf dessen Basis andere Entwickler:innen Applikationen entwickeln können sollen, bedingt, dass jenes bei Veröffentlichung bereits möglichst weit ausdefiniert ist, da nachträgliche Änderungen am Protokoll die Funktionsweisen aufsetzender Programme beeinträchtigen können.

Die konzeptionellen Kernpunkte des Matrix-Protokolls sind Offenheit, Dezentralität und Sicherheit¹⁵. Außerdem ist es erklärtes Ziel, dass Matrix mit möglichst vielen anderen Kommunikationsplattformen „sprechen“ können soll. So kann Matrix etwa per SMS kommunizieren oder könnte als Mittler zwischen Slack und Discord eintreten. Riot hat also keine aus den Kernpunkten ableitbare abgegrenzte Zielgruppe, sondern erhebt den Anspruch für möglichst alle Menschen zugänglich zu sein.

Trotzdem ist auch bei Riot wieder eine Dominanz IRC-artiger Paradigmen festzustellen. Da auszuschließen ist, dass diese Art der Organisation von Kommunikation die einzig sinnvolle ist, stellt sich die Frage, warum diese Organisationsart auch hier wieder Verwendung findet. Einen Hinweis liefert die Betrachtung der Plattformen, mit denen Riot bisher kommunizieren kann.

Bei all diesen Plattformen ist abermals eine IRC-artige Grundstruktur aus Channelliste und Nachrichtenverlauf zu erkennen. Es erscheint zunächst logisch, dass Riot, in seiner vorgesehenen Rolle als Aggregator von Nachrichten der anderen Plattformen, deren Funktionsparadigmen aufgreift. Technisch zwingend ist es allerdings nicht. Riot könnte etwa auch die Daten der anderen Plattformen sammeln, mit eigenen Konzepten anreichern und den Nutzer:innen Wahlmöglichkeiten in der Aufbereitung der Daten anbieten.

Die immer wiederkehrende Reproduktion des IRC-Schemas sehe ich vielmehr als in der im Softwarebereich stark ausgeprägten Kultur von Wiederverwendung und Modularisierung begründet. Doug McIlroy formulierte etwa bereits 1978 im „Bell System Technical Journal“ einen Satz von Maximen für Programmentwicklung, der unter anderem auf Interoperabilität und Kleinteiligkeit abzielt¹⁶. Diese sogenannten Unix Prinzipien werden heute noch immer wertgeschätzt und Wiederverwendbarkeit gilt als „one of the [...] major software quality factors“¹⁷. Aus dieser Tradition heraus hat sich eine Anzahl von „software design patterns“ gebildet. Also bewährte Logiken, die in Standardwerken wie „Design Patterns: Elements of Reusable Object-Oriented Software“¹⁸ gesammelt und puliziert werden, und im übertragenen Sinne ein „Grundvokabular“ für die Softwareentwicklung darstellen. Üblicherweise wird überall dort auf solche design patterns zurückgegriffen, wo das jeweilige Problem nicht explizit nach einer anderen Lösung verlangt, allein schon, weil eine andauernde Neuentwicklung von Logiken zu aufwendig und fehlerträchtig wäre. Durch diese Praxis entfalten design patterns eine normative Wirkung.

Die Kultur der Wiederverwendung hat mit dem Aufkommen grafischer Programmoberflächen auch in deren Entwicklung Einzug gehalten. So ist es kaum ehrenrührig das Aussehen eines anderen Programms nahezu exakt

zu kopieren, wenn die spezifische Innovation eines zu erstellenden Programms auf rein technischer Ebene liegt. Im Gegenteil: Der Rückgriff auf Gestaltungssysteme, sogenannte „UI frameworks“, sowie auf etablierte Interaktionsschemata gilt als „best practice“ wenn eine veränderte Art der Gestaltung oder Interaktion nicht Kernkonzept einer Softwareentwicklung ist. Dies ist nicht zuletzt mit der Hoffnung verbunden, dass die Nutzer:innen der jeweiligen Software den Umgang mit den Anwendung findenden UI frameworks und Funktionsprinzipien bereits in einer anderen Software erlernt haben und damit die Zugänglichkeit verbessert wird.

Das IRC-Protokoll hat, als eines der ersten digitalen Kommunikationsmittel mit weiter Verbreitung, design patterns etabliert, die bis heute weit verbreitet sind, obwohl das heutige Umfeld andere Ansätze ermöglicht. Im Kontext der damals geringeren Rechengeschwindigkeiten und Internetbandbreiten hat der Fokus auf textbasierten Austausch Sinn ergeben. Eine nicht gesicherte synchrone Verbindung der Kommunikationsteilnehmer:innen gebietet den Einsatz von Servern als Zwischenspeicher und Vermittler, welche in der Oberfläche als Chaträume repräsentiert werden. Die zeitlich geordnete Darstellung aller bisherigen Nachrichten ist dabei eine pragmatische Ansicht der Datenbank. Die damals geringere Benutzerdichte und stärkere Anonymität bot weniger Potenzial für das Entstehen komplexerer sozialer Strukturen und hat ein geringeres Bedürfnis nach Abbildbarkeit solcher Strukturen erzeugt.

Vor dem Hintergrund einer deutlich höheren Zugangsbarriere zu Computersystemen zu Zeiten der Entwicklung des IRC-Protokolls, etwa durch höhere Preise solcher Systeme oder das komplexere und weniger gut aufbereitete Wissen, das nötig war um diese Systeme zu nutzen, ist davon auszugehen, dass die damaligen Nutzer:innen, um die sich das IRC-Protokoll geformt hat, deutlich technisch affiner waren oder zumindest weit mehr mit den eingesetzten Computersystemen vertraut.

In der Weiterentwicklung der Anwendungen setzt ein Großteil der Entwickler:innen zur Diskussion von Nutzer:innenfeedback – wenn überhaupt – auf für technisch weniger gebildete Menschen relativ obskure „Ticket Systeme“, etwa auf Entwickler:innenplattformen wie Github, oder auf Feedbackkanäle innerhalb der eigenen Kommunikationsanwendung, was das Entwicklungsbias in Richtung einer technisch affinen Nutzer:innengruppe weiter verstärkt.

Die Gestaltungsansätze von IRC haben sich im damaligen Umfeld als funktional erwiesen, wurden oft aufgegriffen und haben sich so zur Norm ent-

wickelt. Bereits leicht abweichende Konzepte werden als drastische Änderungen wahrgenommen. So wirken die eigentlich relativ banalen Evolutionsschritte etwa bei Instagram, durch den Fokus auf Kommunikation per Bild statt Text und Bevorzugung eines Hauptkommunikationskanals, oder bei Snapchat, durch nur zeitlich begrenzt einsehbare Nachrichten, bereits sehr neuartig.

Zwischenfazit

Die Umstände der Softwareentwicklung begünstigen die Herstellung von Kommunikationsplattformen die Menschen bevorzugen, die den Umgang mit Technologie und Digitalität gewöhnt sind. Sowohl die undeklarierte Zielgruppe vor der Produktwerdung einer Software in inkrementellen Entwicklungsprozessen als auch die Nutzer:innengruppe früher digitaler Kommunikationsformen, deren Adoption IRC-artiger Austauschformate dessen Paradigmen zu design patterns haben werden lassen, bestehen aus Menschen, die als solche charakterisiert werden können.

Die eingangs beschriebenen Kooperationsformen, die durch die zu untersuchenden Softwareprodukte ermöglicht werden, schließen aber auch Personen ein, die nicht oder wenig mit Effekten von Digitalität sozialisiert sind. Im Folgenden werde ich auf Grundlage der IRC-Paradigmen einzelne dieser Effekte herausstellen und Untersuchen, ob und wie die einzelnen Applikationen darauf reagieren.

Örtlichkeit

I Analog zum physischen Raum, der durch die Dimensionen von Länge, Breite und Höhe aufgespannt wird oder dem sozialen Raum, den etwa Pierre Bourdieu durch die Dimensionen von Kapitalvolumen, Kapitalstruktur und sozialer Laufbahn definiert¹⁹, gibt es auch im Digitalen eine Raumwahrnehmung, in der sich Orte durch eine jeweils distinktive soziale und funktionale Konfiguration abgrenzen lassen.

Eine Besonderheit des digitalen Raums ist die regelmäßig fast vollständige Nichtexistenz von Zeitkosten eines Ortswechsels. Wurde ein digitaler Ort, beispielsweise eine Website oder eine Ansicht in einer Software, einmal lokalisiert, kann er üblicherweise per eindeutiger Referenz in Form einer Webadresse oder per leicht reproduzierbarer Vorgehensweisen innerhalb der Software schnell wieder aufgefunden werden. Durch das parallele Ausführen mehrerer Programme oder das Öffnen verschiedener Webansichten in mehreren Browserinstanzen kann die Transitionszeit zwischen den Orten bis auf die Refokussierungsdauer der Sinnesorgane verkürzt werden.

In IRC-artigen Kommunikationsplattformen bilden die einzelnen Channels jeweils eigene digitale Orte, die jederzeit verfügbar sind. Im Gegensatz zur Kommunikation in physischen Räumen, wo der Wechsel zwischen Gesprächssituationen einen körperlichen Akt eines oder mehrerer Teilnehmender voraussetzt, kann ein solcher Wechsel im Digitalen instantan erfolgen. Prozesse, die im Zwischenraum zwischen Gesprächen stattfinden, etwa die Reflexion über Gesagtes oder die mentale Vorbereitung auf das nächste Gespräch, fallen damit weg. Darüber hinaus befördern die geringen Transitionskosten einen häufigen Ortswechsel bis hin zu einer quasi-parallelen Präsenz in mehreren Gesprächen oder, bei geringer Relevanz des Gesagten für die oder den Ankommende:n, einer nur sekundlichen Anwesenheit. Der Ortswechsel verliert dadurch an Signifikanz und die dem physischen Beitreten oder Verlassen einer Gesprächssituation eingeschriebenen Interaktionsmuster, wie Begrüßung und Einführung in die Gruppe oder inhaltlicher Abschluss und Verabschiedung, fallen weg oder müssen aktiv herbeigeführt werden.

II

Durch die Möglichkeit des unmittelbaren Wechsels der Gesprächsorte per direkter Referenzen in diesen digitalen Kommunikationsplattformen entfällt zudem die Durchquerung öffentlicherer Orte, denen eine wichtige sozialisierende Funktion zukommt, da sie zufällige Begegnungen und informellere Kommunikation ermöglichen. Vaiou und Kalandides etwa führen aus, dass „the borders between familiar and strange, insider and outsider are re-negotiated and even challenged, while public space acquires new meanings“²⁰. Eine übliche Reaktion auf den Bedarf von Öffentlichkeit und informellerer Begegnung in IRC-artigen Plattformen ist die Einrichtung verschiedener Räume mit identischen Nutzer:innen, aber verschiedenen Ansprüchen an die Art der Gesprächsführung. In Räumen, die als Öffentlichkeit deklariert sind, findet etwa der persönlichere Austausch über alltägliche Vorkommnisse mehr Akzeptanz als in themenbezogenen Räumen. Im Gegensatz zum physischen Raum gibt es aber keinen natürlichen Automatismus, der zu dem Besuch dieser Orte führt. Nutzer:innen müssen diesen Besuch selbst initiieren.

Ergo

Die IRC-Paradigmen bevorzugen durch die Möglichkeit des direkten Ortswechsels Menschen, denen der schnelle Wechsel zwischen Gesprächskontexten leichter fällt und deren Kommunikationsverhalten weniger auf Interaktionsmustern basiert. Eine Person, die beispielsweise gewöhnt ist ihr Verhalten stark davon abhängig zu machen wie sie von ihrem Gesprächspartner empfangen wird, ist hier im Nachteil. Die hier zu untersuchenden Programme weichen in diesem Punkt nicht von den IRC-Paradigmen ab. Da in einem kollaborativen Prozess üblicherweise nur die Anwesenheit in direkt projektbezogenen Kanälen vorausgesetzt wird, setzt das Besuchen oben genannter als öffentlich deklarerter Räume eine intrinsische Motivation und damit einen lustvollen Umgang mit digitalen Werkzeugen voraus. Slack und Discord begegnen dem Bedarf von Öffentlichkeit durch das Anzeigen einer Liste aller Beteiligter der jeweiligen Raumsammlung (bei Slack „workspace“ genannt; bei Discord „server“). In dieser Liste kann angezeigt werden, ob die einzelnen Beteiligten gerade „aktiv“ oder „online“ sind, also ob die Beteiligten das Programm gerade ausführen. Abgesehen von der technischen Schwierigkeit für das Programm festzustellen, ob es tatsächlich benutzt wird und nicht nur geöffnet ist, kann bei den Nutzer:innen somit durch bewusst gemachte gleichzeitige Aktivität der Eindruck einer öffentlichen Begegnung entstehen, welcher gegebenenfalls eine Interaktion provoziert.

I
In der Nutzung von IRC kann durch den Einsatz von „Bouncern“, also Hilfsprogrammen, die kontinuierlich Nachrichten empfangen können, oder durch Servereinstellungen, die das Einsehen des „Backlogs“, also der Nachrichtenhistorie, erlauben, der Nachrichtenverlauf eines Raumes eingesehen werden; im Falle des Bouncers der Nachrichtenverlauf ab dem erstmaligen Betreten des Raums, im Falle des Backlogs der gesamte Nachrichtenverlauf eines Raumes. Dadurch werden die jeweiligen Nachrichtentexte persistent und eindeutig festgehalten. Gespräche werden so zu einer durchsuchbaren Datensammlung, in der Aussagen von Gesprächsteilnehmer:innen konkret nachgeschlagen oder komplette Gesprächstexte zu einem späteren Zeitpunkt nachverfolgt werden können. Das hat den Vorteil, dass ungenaue Erinnerungen an Inhalte im Nachhinein aufgefrischt werden können und Neuankömmlinge das Ausbleiben einer Einführung (siehe Örtlichkeit I) selbstständig durch Nachvollzug der Historie kompensieren können.

Allerdings kann die Persistenz von Nachrichtentexten leicht als Persistenz der Nachricht selbst aufgefasst werden, die neben dem Text aber auch aus dem sozialen Kontext besteht, also „der Beziehung zwischen Sprecher[in] und Hörer[in], ihrem Wissen und ihren Einstellungen“²¹, welcher nicht festgehalten wird. Es ist davon auszugehen, dass das Wissen um diesen Kontext mit der zeitlichen Entfernung abnimmt, eine Fehlinterpretation wahrscheinlicher wird und eine Art falsche Geschichtsschreibung auftreten kann. Ein Neuankömmling kann etwa bei fehlendem Wissen um die Reputation anderer Teilnehmer:innen die Gewichtigkeit von Äußerungen schwer einschätzen; Vorschläge, die ursprünglich in einem schnellen Nachrichtenwechsel untergegangen sind, können nachträglich als besprochen und verabredet deklariert werden. Durch den so auftretenden Verlust der Hoheit über eigene Aussagen, welcher noch dadurch verstärkt wird, dass sich Nachrichten im Nachhinein praktisch nicht löschen lassen, kann ein „Chilling Effect“ auftreten, also ein „störende[r], einschüchternde[r] Einfluss auf die Ausübung eines Frei-

heitsgrundrechts“²², „fast immer [dem der] Meinungsäußerungsfreiheit“²³. Beispielsweise kann es bei mangelndem Vertrauen in die jeweilige Gesprächsgruppe folglich dazu kommen, dass Aussagen, die später missbräuchlich umdefiniert werden könnten, zurückgehalten werden.

II

Besteht in der jeweiligen Gruppe die Konvention, dass alle Nichtanwesenden die in ihrer Abwesenheit erfolgte Kommunikation im Nachhinein nachvollziehen, dann führt dies zu einer Adressierbarkeit der Abwesenden. Aus der Sicht der sendenden Person nehme ich jeder Zeit meine Rolle als Nachrichtenempfänger:in wahr, da in der technisch erfolgten Übermittlung der Nachricht in meinen Sichtbarkeitsbereich immer das Versprechen meiner Rezeption angelegt ist, somit bin ich virtuell immer anwesend.

Gespräche können so zwischen allen Beteiligten geführt werden, auch wenn immer jeweils nur eine oder ein Beteiligte:r zur Zeit tatsächlich am digitalen Gesprächsort zugegen ist. Dies hat den Vorteil einer größeren zeitlichen Unabhängigkeit, aber nimmt den Gesprächen auch die Rahmung, die die zeitliche Verfügbarkeit der Gesprächsteilnehmer:innen im physischen Raum setzen kann. Inhaltliche Bögen von Gesprächen im digitalen Raum können so über einen deutlich längeren Zeitraum gezogen werden, was einer inhaltlichen Tiefe zugutekommen, aber etwa auch die Verschleppung von Entscheidungen und inhaltlichen Stillstand erzeugen kann oder bei längerer Abwesenheit der Empfänger:innen deren Aufnahmekapazität sprengen kann.

Die Ansprache einer nur virtuell anwesenden Person kann darüber hinaus bei dem Sender den Eindruck einer erfolgreichen Kommunikation erwecken, wodurch aus dessen Sicht die Verantwortung der weiteren Gesprächsführung auf den oder die Empfänger:in übergeht. So entsteht gegebenenfalls der Erwartungsdruck eines regelmäßigen Nachvollzugs aller Nachrichten in einem Gesprächsraum durch alle Beteiligten.

III

Ein weiterer zeitlicher Aspekt ist der Unterschied zwischen der Dauer des Lesens und Schreibens von Nachrichten. Unter der Annahme, dass das Verhältnis dieser beiden Zeitspannen in IRC-artigen Plattformen ähnlich zu dem von E-Mails ist, so dauert das Verfassen einer Nachricht durchschnittlich rund fünfmal so lange wie das Lesen einer Nachricht²⁴. Einerseits bedeutet dies im Bezug auf die einzelnen Gesprächsteilnehmer:innen einen deutlichen zeitlichen Mehraufwand für Personen, die aufgrund ihres Kommunikationsstils oder

ihrer sozialen Rolle ein größeres Sendevolumen als andere Teilnehmer:innen haben. Andererseits erzeugt die kommunizierende Person in Gruppen, die in ihrer Größe über fünf Personen hinausgehen, durchschnittlich einen höheren kumulativen Zeitaufwand für die Menge der Empfänger:innen. Unter der Annahme, dass das Verhältnis von Sende- zu Empfangszeit mit geringerer Informationsdichte und -brisanz abnimmt, kann so beispielsweise „Smalltalk“ zwischen wenigen Teilnehmer:innen einer großen Gesprächsgruppe eine überproportionale Belastung für alle Beteiligten darstellen.

Ergo

Die IRC-Paradigmen in der dargestellten Nutzungsart per Bouncer und Backlogs bevorzugen Menschen mit hoher zeitlicher Verfügbarkeit und einer angewöhnten Regelmäßigkeit der Nutzung der jeweiligen Kommunikationsplattform, deren Kommunikationsstil strukturiert, sachlich und inhaltlich eindeutig ist; sowie Gruppen mit einem größeren gegenseitigem Vertrauen.

Slack, Riot und Discord haben die Funktion des Bouncers verinnerlicht. Wer Gesprächsgruppen beitrifft, bleibt in diesen Gruppen auf unbestimmte Zeit virtuell anwesend so lange kein bewusster Wiederaustritt stattfindet; temporäre Besuche einzelner Räume werden durch die jeweiligen Bedienungskonzepte erschwert. Riot und Discord lassen noch die Wahl, ob Neuankömmlinge das Backlog einsehen dürfen. Bei Slack ist diese Einsehbarkeit immer gegeben. Slack und Discord bieten zudem mächtigere Suchfunktion, die über das bloße Auffinden eines Wortes in der Historie hinausgehen.

Die Plattformen eint außerdem die vorhandene „Erwähnungs-Funktion“: Per Sonderschreibweise können andere Nutzer:innen deutlicher adressiert und bei Adressierungen benachrichtigt werden, wodurch diese deutlich transparenter werden, aber bei erfolgter Erlaubnis der Benachrichtigung außerhalb der Plattform, etwa per E-Mail, einen höheren Reaktionsdruck erzeugen.

Alle drei Plattformen bieten darüber hinaus auch nicht-textliche Kommunikationsformen, wie etwa den Austausch per Audio- oder Videosignale. Die Inhalte dieser Kommunikationsformen werden nicht in die Gesprächshistorie eingeschrieben und deren Verwendung setzt zeitgleiche Anwesenheit der interagierenden Nutzer:innen voraus, wodurch deren Nutzung die hier beschriebenen Effekte deutlich mindert. Allerdings ist nur bei Discord, mit dessen entstehungsgeschichtlichem Ursprung in der Audiokommunikation, eine Gleichrangigkeit der Kommunikationsformen in der Oberfläche gegeben.

Körperlichkeit

I Kommunikation bezeichnet im sprachwissenschaftlichen Sinn die zwischenmenschliche Verständigung anhand sprachlicher, aber auch nichtsprachlicher Mittel²⁵. Dem Kommunikationsmodell nach Schulz von Thun folgend, hat eine Äußerung in der zwischenmenschlichen Verständigung stets vier Aspekte: einen Sachinhalt, eine Selbstkundgabe, einen Beziehungshinweis und einen Appell²⁶. Abgesehen vom Sachinhalt, der sich üblicherweise klar durch den textlichen Inhalt einer Äußerung transportieren lässt, ist die korrekte Interpretation dieser Aspekte oft nur unter Einbeziehung des situativen Verhaltens des Sendenden möglich. Eine Äußerung kann, bei gleichem textlichen Inhalt, durch Mimik, Gestik oder Betonung sehr unterschiedlich konnotiert werden.

Diese situativen Verhaltensweisen gehen in IRC-artigen Kommunikationsmitteln überwiegend verloren, da die Vermittlung dieser Verhaltensweisen stark auf Körperfunktionen der Gesprächsteilnehmer:innen basiert und eine Körperlichkeit durch den rein textlichen Austausch kaum gegeben ist. Dieser Verlust kann teilweise durch die Kodierung der Körperlichkeit in textlichen Formen ausgeglichen werden. Im Umgang mit IRC-artigen Kommunikationsmitteln haben sich etwa Gebrauchsmuster wie das „Emoticon“, also Zeichenfolgen die Mimik repräsentieren, oder die Beschreibung einer virtuellen Handlung, die das eigentliche Gesagte kontextualisiert (beispielsweise „/me shakes fist in the air“), etabliert. Durch den begrenzten Zeichenvorrat in textlicher Kommunikation, sowie der begrenzten Effizienz und Unmittelbarkeit solcher Beschreibungen kann dies aber nur einen bedingten Ersatz schaffen. Außerdem setzt eine solche Vermittlung von Emotionalität durch den Sendenden ein Bewusstsein über seine Emotionen voraus, wohingegen bei Kommunikation im physischen Raum oft auch dem Sendenden unbewusste Kommunikation stattfindet.

Weiterhin lässt die Körperlosigkeit IRC-artiger Kommunikation gegebenenfalls ein Abweichen von einem der von Paul Watzlawick aufgestellten Axiome der menschlichen Kommunikation zu, welches besagt, dass

man „nicht nicht kommunizieren“⁴²⁷ kann. Da die Anwesenheit des oder der jeweiligen Empfänger-in und der Erfolg der Übermittlung einer Nachricht auf technischer Ebene per se nicht nachgewiesen werden können, ist es möglich, sich einer Kommunikation zu verweigern.

II

Nach der Sapir-Whorf-Hypothese, „derzufolge die Sprachen Denken und Wahrnehmung ihrer jeweiligen Sprecher determinieren“⁴²⁸, ist davon auszugehen, dass der Einfluss der Körperlosigkeit auf die Semantik der Sprache einen Rückkopplungseffekt zur Denkweise der Teilnehmenden an IRC-artigen Kommunikationsformen hat. Wenn körperliche Effekte im Sprachgebrauch regelmäßig ausbleiben und so ein Wandel der Sprache zu einer Form eintritt, die weniger auf dem Vermittlungskontext von Körperlichkeit beruht, dann erzeugt dies eine verminderte Kongruenz von Körper und Person im Denken über Andere, die wiederum einen weniger körperdependenten Kommunikationsstil erzeugt. Dadurch treten textuell nicht vermittelbare Kommunikationsanteile und daraus resultierende Kommunikationsstörungen weniger stark auf.

Bei Menschen, die eine starke Sozialisation durch digitale Kommunikation erfahren haben, kann dies auch auf deren Kommunikationsverhalten im physischen Raum rückwirken. Eine Reihe von Studien belegen, dass etwa Kinder der „Internet-Generation“ ein anderes Interaktionsverhalten erlernt haben. So folgern etwa Thomas Wells Brignall und Thomas Van Valey in einer Metastudie: „if students develop unique interaction rituals based on online communication without enough experience or understanding of traditional face-to-face interaction rituals, the likelihood for friction and/or conflict to occur undoubtedly increases.“⁴²⁹

III

Viele Botschaften werden in physischer Kommunikation nicht bloß körpersprachlich ergänzt, sondern ausschließlich per Körpersprache vermittelt. Diese Meta-Botschaften sind überwiegend sozialer und weniger thematischer Natur, beispielsweise Zustimmung und Ablehnung, das Bedürfnis etwas Beizutragen oder das Ende einer Aussage. Durch den Ausdruck per Körpersprache können diese Botschaften vermittelt werden ohne den Konversationsfluss zu unterbrechen. IRC-artige Kommunikationsplattformen hingegen bieten keine verschiedenen „Lautstärken“ der Äußerungen. Alle Nachrichten erscheinen in gleicher Formatierung nach ausschließlich zeitlichem Ordnungsprinzip, wodurch die Einordnung der Nachrichten zwischen thematischen Inhalten

und Meta-Botschaften stark erschwert wird und die Übersichtlichkeit sowie Nachvollziehbarkeit von Gesprächen leidet. Dies kann zu einer Zurückhaltung dieser Meta-Botschaften führen. Dadurch wird ein gegebenenfalls bestehender Bedarf nach Feedback auf Aussagen oft nicht befriedigt oder es kommt etwa zu einer Vermischung von Nachrichtenketten mehrerer Beteiligter, wenn der Abschluss der Nachrichtenkette des jeweils Anderen falsch antizipiert wird.

Im Sonderfall der inhaltlichen Kommunikation per Gestik, die nicht durch textuelle Nachrichten ersetzt werden kann, etwa ein Austausch über Sprachbarrieren hinweg, kann innerhalb der IRC-Paradigmen keine Verständigung erfolgen.

Ergo

IRC-artige Kommunikationsformen bevorzugen Menschen, deren Kommunikationsverhalten weniger stark auf der Vermittlung ihrer Botschaften durch körperlichen Ausdruck basiert und die einen geringeren Bedarf der Aushandlung ihres Verhaltens über die Rückmeldung Anderer haben. Damit werden generell auch Menschen bevorzugt, die bereits stärker durch gleichartige Kommunikationsformen sozialisiert sind. Die bereits oben geschilderten Möglichkeiten in Slack, Riot und Discord auch nicht-textliche Kommunikationsformate einzubinden, vermindern auch die hier geschilderten Effekte der Körperlosigkeit. Schon die alleinige Nutzung von Audiokommunikation erlaubt durch die Übertragung von Stimmlage und Betonungen eine deutlich stärkere Kontextualisierung des Gesagten; kommt noch die Nutzung eines Videosignals hinzu, welches die Gesprächsteilnehmer-innen zeigt, so kann von einer fast vollständigen Mitigation der Körperlosigkeit ausgegangen werden.

Im IRC-ähnlichen Bereich der drei Plattformen hingegen gibt es keine Paradigmenänderung, sondern lediglich eine überschaubare Funktionserweiterung. Durch das Angebot von „Emojis“, also kleine grafische Abbildungen von beispielsweise Gesichtern, Tieren oder Gegenständen, wird der Zeichenvorrat der Emoticons ausgeweitet. Diese Emojis lassen sich in Slack und Discord als „reactions“ auch an den Nachrichten anderer Gesprächsteilnehmer-innen befestigen und bieten so eine rudimentäre Möglichkeit des dem Textfluss ausgelagerten Feedbacks. Riot zeigt Empfangsbestätigungen, also welche-r Beteiligte-r bis zu welcher Nachricht gelesen hat, was eine Nichtkommunikation verhindert.

Die größte Veränderung macht die Möglichkeit aus, multimediale Inhalte als Nachricht in den Textfluss einzulagern, da etwa ein Foto gegebenenfalls eine deutlich nuanciertere emotionale Bedeutung vermitteln kann, als es durch eine textuelle Nachricht möglich wäre.

Kommunikation im Allgemeinen und Gruppenkommunikation im Besonderen wird durch die Hierarchiestruktur der Teilnehmenden beeinflusst. So zeigen etwa Behrendt et al. in einer Fallstudie, dass eine als gleichwertig empfundene Hierarchieebene der Kommunikationsteilnehmer:innen deren Austausch insbesondere in der Gruppenkommunikation fördert, wohingegen als verschieden empfundene Hierarchieebenen eine Abnahme der Kommunikation zur Folge hat.³⁰ Außerdem wird beobachtet, dass Kommunikationsteilnehmer:innen „try to minimize communication with higher levels“³¹.

Jo Freeman argumentiert in „The Tyranny of Structurelessness“, dass es keine [hierarchie-]strukturlosen Gruppen gibt, da sich jede Menschengruppe unter allen Umständen eine Struktur gibt³². Im weiteren heißt es: „Thus 'structurelessness' becomes a way of masking power [...]. The rules of how decisions are made are known only to a few and awareness of power is curtailed to those who know the rules, as long as the structure of the group is informal.“³³ Darüber hinaus stellt sie fest, dass in Abwesenheit einer Entscheidung darüber, wer in der Gruppe Macht ausüben soll, die Mächtigkeit der Einzelnen an deren Personalität fest gemacht wird und nicht etwa an deren Kompetenz oder deren potentielltem Beitrag zum Ziel der Gruppe³⁴.

IRC-artige Kommunikationsplattformen bieten in ihrer Oberfläche keine Abbildbarkeit von Hierarchie und keine gestalterischen Elemente, die eine Formulierung der konkreten oder empfundenen Machtstrukturen begünstigen. Ein solches Definitionsvakuum wird, wie bei Freeman beschrieben, durch eine Analyse der persönlichen Attribute der Teilnehmenden gefüllt, was zwar problematisch sein kann, aber auf angelernten Mustern und damit auf erfolgreichen Attribuerungen in der Vergangenheit beruht. Durch den oben beschriebenen Wegfall der Körperlichkeit im IRC-Paradigma geht allerdings ein großer Teil des im Physischen gegebenen Zeichenraums der Attribution verloren. In einem Klassenraum etwa lässt sich der Lehrer als mächtigste Person leicht durch seine körperlichen Merkmale ausmachen. In IRC-artigen

Kommunikationsplattformen werden die Teilnehmer:innen aber zunächst nur durch ihren Namen repräsentiert.

Dies führt zu einer Überbetonung der verbleibenden Merkmale, also Kommunikationsstil und Kommunikationsinhalt. Der Kommunikationsstil lässt jedoch nur begrenzt Rückschlüsse auf den oder die jeweilige:n Gesprächspartner:in zu und Kommunikationsinhalte, etwa Selbstauskünfte über Fähigkeiten, sind nicht trivial belegbar. Die dadurch entstehende Instabilität des Beziehungsgefüges ist der Kommunikation und Kooperation abträglich. So bilden sich, wie Gerd Sebald herausarbeitet, „in komplexeren und länger anhaltenden Formen der Online-Kooperation spezifische Taktiken der Kommunikation [heraus], in denen Person und Vertrauen über vier Mechanismen konstruiert werden:

a) durch Probezeiten, b) durch multimediale Absicherungen der Konstruktionen, c) durch Rückgriff auf externe Institutionen und d) schließlich durch realweltliche Treffen.“³⁵ Es muss also ein Rückgriff auf Mechanismen erfolgen, die außerhalb der Kommunikationsplattformen wirken.

Bleiben diese Rückgriffe aus, so ist anzunehmen, dass die Grundannahmen der Kommunikationsteilnehmer:innen über die gegebene Machtstruktur weniger Korrektur in Richtung der tatsächlichen Verhältnisse erfährt. So werden Personen, die von einem starken Hierarchiegefälle in einer Gruppe ausgehen und sich nicht selbst im oberen Bereich wähen, eher in der Kommunikation gehemmt, und Personen, die eher eine flache Hierarchie erwarten oder sich selbst in höherer Stellung sehen, vergleichsweise stärker zur Kommunikation angeregt.

Zudem entsteht eine geringere hierarchische Mobilität, da mittels textueller Kommunikation deutlich weniger, als etwa mittels Körpersprache, zum Ausdruck gebracht werden kann, wenn sich die Einzelnen in verschiedenartigen Gruppenkonstellationen in verschiedener Hierarchieposition sehen; insbesondere wenn diese Personen einen zurückhaltenderen Kommunikationsstil pflegen.

Ergo

Die IRC-Paradigmen bevorzugen Menschen, deren Kommunikation weniger durch ihre hierarchische Stellung beeinflusst wird, generell Personen, die durch flachere Hierarchien sozialisiert sind und Gruppen, in denen eine höhere Frequenz des Austauschs stattfindet.

Riot, Slack und Discord ermöglichen eine Rollenverteilung und -darstellung innerhalb ihrer Oberflächen. Bei Riot und Slack sind diese Rollen vor allem organisatorischer Natur. So bestimmen sie etwa, wer welche Räume

betreten oder wer Raumnamen und -themen ändern darf. Die Rollen können damit deutlichere Hinweise auf die Machtstrukturen geben aber auch irreführend sein, da mehr technische Befugnisse nicht zwingend mit einer höheren sozialen Stellung einhergehen. In Slack sind die jeweiligen Befugnisse dabei fest an fünf verschiedene, vordefinierte Rollen geknüpft, wodurch eine Abbildbarkeit der tatsächlichen Verhältnisse erschwert wird. In Riot hingegen erfolgt die Rollendefinition per „Power-Level“ zwischen null und einhundert, wobei die vorgegebenen Berechtigungen pro Level frei erteilt werden können. In der Annahme, dass eine tatsächliche Hierarchie kaum per kontinuierlichem Spektrum abbildbar ist, kann auch diese Definitionsart in die Irre führen, aber sie kann zumindest eine genauere Annäherung an die realen Bedingungen liefern.

Discord bietet hier die mächtigsten Werkzeuge an. Rollen können frei benannt werden und die Befugnisse, die wie bei Riot frei mit den einzelnen Rollen verknüpfbar sind, gehen hier über organisatorische Festlegungen hinaus. So kann beispielsweise einzelnen Nutzer·innen ein Vorzugsrecht in der Sprachkommunikation eingeräumt werden, welches bei Bedarf die Lautstärke der Sprachkommunikation anderer Nutzer·innen senkt. Besonders die freie Namensgebung der Rollen lässt eine klare Kommunikation von Machtstrukturen zu, da über die Rollenbezeichnung ein Anknüpfen an kulturelles Wissen möglich ist. Darüber hinaus fordert Discord in ihren Richtlinien für die „Verifizierung“ einer „Community“, welche eine Art Gütesiegel darstellt, eine zumindest in Grundzügen abgebildete soziale Struktur³⁶ und bietet Informationsmaterial zu erprobten Praktiken.

Fazit

Ausgangspunkt meiner Arbeit war die Annahme, dass Kollaborationssoftware, die eine Form kollektiver Arbeit ohne realweltliche Treffen ermöglichen soll, dem daraus erwachsenden Anspruch der Inklusion eines diversen Felds potenzieller Nutzer·innen nicht gerecht wird. Davon ausgehend, dass verschiedene Anwendung durch deren Eigenheiten jeweils verschiedene Nutzer·innen benachteiligen würden, sollten die jeweiligen Eigenheiten durch den Vergleich dreier Anwendungen herausgestellt werden.

Trotz vorheriger Bekanntschaft mit den Anwendungen ist mir erst durch eine nähere Betrachtung der Plattformen bewusst geworden, dass diese in ihren Kernmechaniken sehr ähnlich sind und sich nur in sekundären Implementationsdetails unterscheiden. Obwohl ich bereits selbst Kommunikationssoftware entwickelt habe, ist mir erst durch diese Betrachtung klar geworden, dass die ständige Reproduktion gleicher Kommunikationsparadigmen in unzähligen Anwendungen, die sich alle auf einen gemeinsamen Vorfahren zurückführen lassen, eine faktische Norm des digitalen Austauschs hervorgebracht hat, welche nicht genügend hinterfragt wird.

Wie sich gezeigt hat, liegt diesem Mangel an Reflektion im Entwicklungsprozess von Software zum einen dessen inkrementelles Vorgehen zugrunde, in dessen Verlauf die jeweiligen Entwickler·innen zunächst sich selbst als Nutzer·innen annehmen. Die Betrachtung einzelner Aspekte der IRC-Paradigmen zeigt, dass viele der bevorteilten menschlichen Eigenschaften auch die Arbeit als Entwickler·in begünstigen oder aus dieser hervorgehen. So trainiert die Herstellung von Code den schnellen Wechsel zwischen logischen Kontexten und den strukturierten sowie eindeutigen Umgang mit Text. Die Beschäftigung mit Software setzt einen lustvollen Umgang mit informationstechnischen Systemen voraus, geht üblicherweise aus einer starken Sozialisation durch digitale Umgebungen hervor und bedingt eine ausgiebige Präsenz in diesen Umgebungen. Vor diesem Hintergrund ist die Erstellung IRC-artiger Anwendungen eine logische Konsequenz aus der Entwicklung für den eigenen Bedarf.

Zum anderen hat sich in der Betrachtung des entwicklungskultu-

rellen Rückgriffs auf design patterns gezeigt, dass der Reproduktion bereits anderweitig Verwendung findender Funktionsprinzipien ein Wert an sich beigemessen wird, da somit bereits angesammeltes Wissen der Nutzer·innen wieder aufgegriffen werden kann. Damit entfällt sowohl eine eventuell abschreckende Einführung in neuartige Mechaniken als auch das Risiko, dass diese Mechaniken sich nur für einen Teil der Nutzer·innen als vorteilhaft gegenüber des „Status quo“ der IRC-Paradigmen erweisen. So wird in Konsequenz einer Art kapitalistischen Verwertungslogik gefolgt, in der aus einer knappen Entwicklungszeit ein Produkt entstehen soll, dass bei möglichst vielen „Verbraucher·innen“ ein Mindestmaß an Zufriedenheit erzeugt, anstatt die Bedürfnisse jeweils kleinerer Nutzergruppen besser zu erfüllen.

So erweist sich der Ansatz einer zentralisierten Kommunikationslösung als Ersatz für den Einsatz verschiedenartiger Austauschformen generell als problematisch, so lange deren Entwicklungsziel und auszeichnende Eigenschaft das Angebot einer einzigen, vermeintlich auf Zugänglichkeit optimierten Kommunikationsform darstellt. Die betrachteten Anwendungen folgen aber genau diesem Muster, indem sie nicht etwa einen Rahmen schaffen, in dem die Nutzer·innen zwischen gleichberechtigten Kommunikationsformen die für sie passendste auswählen können, und die Interoperabilität zwischen diesen Kommunikationsformen verbessern. Vielmehr basieren die Anwendungen jeweils grundlegend auf einer Kommunikationsform, beziehungsweise im Falle Discords auf zwei verschiedenen, und erweitern diese mit neuen Funktionen, welche sich aber immer auf diese Kernkommunikationsformen rückbeziehen. So bieten alle drei Anwendungen durchaus alternative Austauschmöglichkeiten wie Videoanrufe, doch sind diese in der optischen und funktionalen Gestaltung immer den Kernmechanismen untergeordnet. Die Nutzung der Kommunikation per IRC-Paradigmen ist damit stets Voraussetzung in der Nutzung der Kollaborationsplattformen, die Stellung des „chattens“ als wahrgenommener Normalzustand digitaler Kommunikation wird weiter zementiert und die dargestellten Effekte werden, wenn auch in teils abgemilderter Form, repliziert.

Eine einzelne Kommunikationsform wird immer bestimmte Nutzer·innen bevorteilen und andere benachteiligen. Es braucht ein stärkeres Bewusstsein für diesen Umstand bei den Gestalter·innen neuer Kommunikationsformen und kollaborativer Kommunikationsumgebungen. In Anlehnung an die Redewendung, dass soziale Probleme nicht mit Technik lösbar sind, sollte das Ziel zukünftiger Entwicklungen weniger in dem Versuch bestehen, eine Kommunikationsform zu schaffen die allen Nutzer·innen gerecht

wird und damit eine einzelne, „perfekte“ Antwort auf das Problem digitaler Kommunikation gibt, sondern mehr in der Verstärkung der spezifischen Vorteile einzelner Kommunikationsformen und der Herstellung eines pluralen Umfelds liegen, in dem stets die dem jeweiligen Anwendungszweck und jeweiligen persönlichen Bedürfnissen dienende Kommunikationsform gewählt werden kann.

- 1 Vgl. John Koetsier, Venturebeat.com: Flickr founder Stewart Butterfield's new Slack signed up 8,000 companies in 24 hours [Internet]. 15.08.2013. URL: <https://venturebeat.com/2013/08/15/flickr-founder-stewart-butterfields-new-slack-signedup-8000-companies-in-24-hours/>
- 1 Vgl. John Koetsier, Venturebeat.com: Flickr founder Stewart Butterfield's new Slack signed up 8,000 companies in 24 hours [Internet]. 15.08.2013. URL: <https://venturebeat.com/2013/08/15/flickr-founder-stewart-butterfields-new-slack-signedup-8000-companies-in-24-hours/>
- 2 Vgl. Barbara Barkhausen, gruenderszene.de: Slack ist mehr als sieben Milliarden Dollar wert [Internet]. 22.08.2018. URL: <https://www.gruenderszene.de/business/slack-netflix-apple-pure-facebook>
- 3 Vgl. Tazos Lazarides, toucharcade.com: Ex-'Fates Forever' Developers Making 'Discord', a Voice Comm App For Multiplayer Mobile Games [Internet]. 14.09.2015. URL: <https://toucharcade.com/2015/09/14/ex-fates-forever-developers-making-discord-a-voice-comm-app-for-multiplayer-mobile-games/>
- 4 Vgl. Stephan Mündges, zdf.de: Wahlkampf: Wo sich die Rechte sammelt [Internet]. 20.09.2017. URL: <http://web.archive.org/web/20171125002900/https://www.zdf.de/nachrichten>
- 5 Vgl. Gerhard Pahl, Wolfgang Beitz: Konstruktionslehre: Grundlagen erfolgreicher Produktentwicklung Methoden und Anwendung. 7. Auflage. Berlin; Heidelberg; New York: Springer 2007, S.10
- 6 Vgl. Ingo Schulz-Schaeffer, Matthias Bottel: Regulierung durch Technik: Arbeitsverteilung und Arbeitsorganisation in Projekten transnational verteilter Softwareentwicklung. in: Stephen Lessenich (Hg.) 2017: Geschlossene Gesellschaften. Verhandlungen des 38. Kongresses der Deutschen Gesellschaft für Soziologie in Bamberg 2016. S. 4
- 7 Agile Alliance, agilealliance.org: Agile 101: Key Agile Concepts [Internet]. 2018. URL: <https://www.agilealliance.org/agile101/>
- 8 Eric Ries, startuplessonslearned.com: Minimum Viable Product: a guide [Internet]. 03.08.2009. URL: <http://www.startuplessonslearned.com/2009/08/minimum-viable-product-guide.html>
- 9 Vgl. Walter Schönwandt: Denkfallen beim Planen. Braunschweig: Friedr. Vieweg & Sohn Verlagsgesellschaft mbH 1986. S. 21
- 10 Vgl. Owen Thomas, readwrite.com: Die, Email, Die! A Flickr Cofounder Aims To Cut Us All Some Slack [Internet]. 14.08.2013. URL: <https://readwrite.com/2013/08/14/stewart-butterfield-tiny-speck-slack/>
- 11 Vgl. Daniel Stenberg, haxx.se: History of IRC (Internet Relay Chat) [Internet]. Version 0.10. 29.03.2011. URL: <https://daniel.haxx.se/irchistory.html>
- 12 Vgl. Dean Takahashi, venturebeat.com: Hammer & Chisel pivots to voice comm app for multiplayer mobile games [Internet]. 10.09.2015. URL: <https://venturebeat.com/2015/09/10/hammer-chisel-pivots-to-voice-comm-app-for-multiplayer-mobile-games/>
- 13 Vgl. Nayam Amarshe, discordapp.com: Discord is not only for gamers! Change Discord's description [Internet]. 27.09.2017. URL: <https://feedback.discordapp.com/forums/326712-discord-dream-land/suggestions/31651072-discord-is-not-only-forgamers-change-discord-s-d>
- 14 Vgl. discordapp.com: Discord [Herz] Open Source [Internet]. URL: <https://discordapp.com/open-source>
- 15 Vgl. Matthew Hodgson, Electromagnetic Field, youtube.com: Matthew Hodgson: Power to the people: liberating online communication with Matrix.org [Internet]. 07.09.2018. URL: <https://www.youtube.com/watch?v=1TPICntbC5w>
- 16 Vgl. Eric Steven Raymond: The Art of Unix Programming [Internet]. 2003. URL: <http://www.catb.org/esr/writings/taoup/html/ch01s06.html>
- 17 William B. Frakes, Kyo Kang: Software Reuse Research: Status and Future. In: IEEE Transactions On Software Engineering Vol. 31 Nr.7. 07.2015. S. 1
- 18 Erich Gamma et al: Design patterns : elements of reusable object-oriented software. Addison-Wesley Professional. 1994
- 19 Vgl. Reemda Tieben, uni-muenster.de: Klassentheoretisches Modell vom sozialen Raum [Internet]. 2003. URL: <https://www.uni-muenster.de/FNZ-Online/theorien/modernisierung/unterpunkte/raum.htm>
- 20 Dina Vaiou, Ares Kalandides: Cities of „others“: public space and everyday practices. In: Geographica Helvetica. Jg. 64 2009/I. Zürich. S. 18
- 21 Hadumod Bußmann: Lexikon der Sprachwissenschaft. 2. Aufl. Stuttgart: Kröner 1990. S. 416
- 22 Simon Assion: Überwachung und CHilling Effects. In: Telemedicus e.V. (Hrsg.): Überwachung und Recht Tagungsband zur Telemedicus Sommerkonferenz 2014. Berlin: epubli GmbH 2014. S. 32
- 23 Ebd.
- 24 Vgl. James Gleick: Die Information: Geschichte, Theorie, Flut. München: Redline Verlag 2011. S. 440
- 25 Vgl. Hadumod Bußmann: Lexikon der Sprachwissenschaft. 2. Aufl. Stuttgart: Kröner 1990. S. 392
- 26 Vgl. Friedemann Schulz von Thun: Miteinander reden: Lexikon der Kommunikationspsychologie. Rowohlt Digitalbuch 2012 S. 175
- 27 Paul Watzlawick, paulwatzlawick.de: Die Axiome von Paul Watzlawick [Internet]. URL: <https://www.paul-watzlawick.de/axiome.html>
- 28 Hadumod Bußmann: Lexikon der Sprachwissenschaft. 2. Aufl. Stuttgart: Kröner 1990. S. 657
- 29 Thomas Wells Brignal III, Thomas Van Valey: The Impact of Internet Communications on Social Interaction. In: Mid-South Sociological Association(Hrsg.): Sociological Spectrum Vol. 25. 2005. S. 15
- 30 Vgl. Sebastian Behrendt et al: The Impact of Formal Hierarchies on Enterprise Social Networking Behavior: Completed Research Paper. In: Thirty Sixth International Conference on Information Systems. Fort Worth 2015 S.16
- 31 Ebd. S. 14
- 32 Jo Freeman: The Tyranny of Structurelessness. Nachdruck von Black Rose No 1. 1974 S. 4
- 33 Ebd. S. 5
- 34 Vgl. Ebd.
- 35 Gerd Sebald: Person und Vertrauen. Mediale Konstruktionen in den Online-Kooperationen der Free/Open-Source-Softwareentwicklung. In: Christian Stegbauer, Michael Jäckel (Hrsg): Social Software: Formen der Kooperation in computerbasierten Netzwerken. Wiesbaden: VS Verlag für Sozialwissenschaften 2008 S. 11
- 36 discordapp.com: Verified Server Moderation Guidelines [Internet]. 16.07.2018.URL: <https://support.discordapp.com/hc/en-us/articles/115001987272>

