

UNTERSUCHUNG DER POTENZIALE VON FREIER SOFTWARE UND OFFENEN STANDARDS

ABSCHLUSSBERICHT





Stadt Dortmund

vertreten durch:
Dortmunder Systemhaus

<https://dortmund.de>

Dieses Dokument steht unter der Creative-Commons-Lizenz CC-BY-SA 3.0 (Namensnennung und Weitergabe unter gleichen Bedingungen).



Stand: August 2022

INHALTSVERZEICHNIS

1	HANDLUNGSANLASS.....	5
2	ANALYSE DES POTENZIALS FREIER SOFTWARE UND OFFENER STANDARDS	6
2.1	Darstellung von Potenzialen	6
2.1.1	Transparenz und Vertrauen	6
2.1.2	Unabhängigkeit.....	7
2.1.3	Flexibilität durch Anpassungen	8
2.1.4	Ausfallsicherheit	9
2.1.5	Zusammenfassung	10
2.2	Best Practice.....	11
2.2.1	Ausgaben.....	11
2.2.2	Entwickler*innen und Administrator*innen.....	12
2.2.2.1	Notwendigkeit.....	12
2.2.2.2	Problematik	13
2.2.3	Geschwindigkeit.....	13
2.2.4	Einer für Alle	14
2.2.5	Beitrag zu Freier Software	15
2.2.5.1	Weiterverbreitung von Anpassungen.....	15
2.2.5.2	Unterstützung durch Pflege- und Supportverträge	15
2.2.6	Kooperation.....	16
2.2.6.1	Zwischen Verwaltungen.....	16
2.2.6.2	Organisationen und Bürger*innen.....	17
2.2.7	Beispiele	18
2.2.7.1	OParl	18
2.2.7.2	Ein Ort für öffentlichen Code	18
2.2.7.3	Geo-Software Esri und QGis.....	19
2.2.7.4	Bayerische Ehrenamtskarte.....	20
2.2.7.5	Palim! Palim! – Stadt Bühl	21
2.2.7.6	Offene Daten Portal Stadt Frankfurt am Main.....	21
2.2.8	Zusammenfassung	21
2.3	Kritische Erfolgsfaktoren	23
2.3.1	IT-Sicherheit und Datenschutz	23
2.3.2	Interoperabilität mit bestehender Software	24
2.3.3	Hilfe bei Benutzung	25

2.3.4	Qualität.....	26
2.3.5	Zusammenfassung	27
3	BEWERTUNG DER LIZENZMODELLE FREIER SOFTWARE	28
3.1	Bewertung	28
3.1.1	Allgemein.....	28
3.1.2	Marktstrukturen	29
3.1.3	Anwendungsbereiche	30
3.2	Fazit.....	31
4	BEWERTUNG VON DATEI-FORMATEN.....	32
4.1	Analyse von Datei-Formaten	32
4.2	Festlegung von Standard-Datei-Formaten.....	32
4.2.1	Formate für Office-Dokumente.....	33
4.2.1.1	Textdokumente zum Informationsaustausch.....	33
4.2.1.2	Textdokumente zur Weiterverarbeitung.....	34
4.2.1.3	Tabellenkalkulationen zum Informationsaustausch	36
4.2.1.4	Tabellenkalkulationen zur Weiterverarbeitung	36
4.2.1.5	Präsentationen zum Informationsaustausch	37
4.2.1.6	Präsentationen zur Weiterverarbeitung	38
4.2.2	Formate für Grafiken	39
4.2.3	Formate für Multimediadaten.....	40
4.2.4	Formate für Datenkomprimierung	41
5	RÜCKSCHLÜSSE FÜR DIE STADT DORTMUND	42

1 HANDLUNGSANLASS

Mit Beschluss vom 07.07.2016 wurde der Masterplan „Digitales Dortmund“ (DS-Nr.: 03339-15) vom Rat der Stadt Dortmund verabschiedet. Dieser wurde am 01.06.2017 in den Masterplan „Digitale Stadtverwaltung“ (DS-Nr.: 07035-17) und schließlich mit Beschluss vom 15.11.2018 in den Masterplan „Digitale Verwaltung – Arbeiten 4.0“ (DS-Nr.: 11784-18) überführt. Mit diesen Beschlüssen wird die Digitalisierung in der Dortmunder Stadtverwaltung auf der Grundlage eines möglichst breiten Denk- und Handlungsansatzes neu ausgerichtet. Da die heutigen informationsverarbeitenden Systeme überwiegend digital arbeiten (d.h. diskrete Informationen verarbeiten), wird unter dem Begriff der Digitalisierung häufig auch der Softwareeinsatz verstanden. Allgemein ist ein vielfältiger gesellschaftlicher Trend zu digitaler Vernetzung zwischen Menschen, Dingen und Arbeitsprozessen zu erkennen. Die wachsende Bandbreite an digital erschlossenen Anwendungsfeldern wird dabei von einer stetig wachsenden Anzahl von Softwareprodukten und Geräteklassen bewältigt.

Die Digitalisierung ist dabei der Treiber, um einfacher zu werden und den Zugang zu vielen Verwaltungsleistungen noch stärker online anzubieten als bislang schon. Nicht nur, weil gesetzliche Grundlagen, wie das E-Government-Gesetz (EGovG), das Onlinezugangsgesetz (OZG) oder die Corona-Pandemie es so fordern, sondern weil digitale Services ein selbstverständlicher Teil eines immer digitaler werdenden Lebens sind.

Einen Beitrag zur Realisierung kann dabei Open Source leisten. Das „Memorandum zur Digitalisierung 2020 bis 2025“, das durch den Rat der Stadt Dortmund bereits eine Zielsetzung an die Verwaltung formuliert, verweist auf die hohe Bedeutung von Open Source für die digitale Souveränität der Kommune. Es setzt das Ziel, Open Source Software zu nutzen, wo es möglich ist und die von der Verwaltung erarbeitete oder zur Entwicklung beauftragte Software der Allgemeinheit zur Verfügung zu stellen.

Die Stadtverwaltung Dortmund hat daher untersucht, welche Potenziale Freie Software und Offene Standards bei der Optimierung des Zusammenspiels zwischen Akteuren und Softwarekomponenten haben und welche notwendige Flexibilität sie bei gleichzeitig hoher Steuerungsfähigkeit bieten könnten. Die Zielvorstellungen waren dabei sehr weitläufig. Zu den Kernpunkten dieser Untersuchung zählten unter anderem die langfristige Reduzierung der Herstellerabhängigkeit, eine Erhöhung der Flexibilität und der Erhalt der Steuerungsfähigkeit beim Softwareeinsatz sowie das Potenzial zur Stärkung der IT-Sicherheit.

2 ANALYSE DES POTENZIALS FREIER SOFTWARE UND OFFENER STANDARDS

Dieser Abschnitt soll eine Analyse des Potenzials der einzelnen Faktoren beim Einsatz von Freier Software und Offenen Standards geben. Darüber hinaus sollen mit diesem Kapitel Einblicke in das Themenfeld der Best Practices gegeben und die kritischen Erfolgsfaktoren herausgestellt werden.

2.1 Darstellung von Potenzialen

2.1.1 Transparenz und Vertrauen

Freie Software bietet ein hohes Maß an Transparenz und hilft Bürger*innen, Abläufe besser zu verstehen, indem der Quelltext (engl. Source Code) von jedem uneingeschränkt eingesehen, nachvollzogen und verändert werden kann. Es ist jeder in der Lage, die Verwaltung bei ihrer Arbeit zu verstehen, zu begleiten und zu unterstützen. Bürger*innen sowie Unternehmen können sich ein Bild über interne Prozesse machen, neue Funktionen vorschlagen und potenzielle Fehler und Sicherheitslücken frühzeitig melden oder selbst beheben. Diese Veränderungen können anschließend von Projektinhabern zu ihrer freien Software hinzugefügt werden.

Proprietäre Software bietet hingegen oftmals keine Möglichkeit, den Quelltext einzusehen, also auch keine Möglichkeit, diesen zu bewerten und nachzuvollziehen. Außenstehende haben keine Chance, an eine Kopie des Quelltextes zu gelangen. So kann Freie Software ein zusätzliches Vertrauen in die Digitalisierung und damit verbundene Abläufe vermitteln.

Ein Kerngeschäft der Verwaltungen ist die Verarbeitung von personenbezogenen Daten, daher wäre eine Transparenz nach außen von großem Nutzen. Misstrauen aufgrund proprietärer Softwaresysteme hilft nicht, Bürger*innen zu überzeugen. Zusätzlich können durch Freie Software gesammelte Daten oftmals, mittels eines freien Standards, in einem maschinenlesbaren Format öffentlich zugänglich gemacht werden. Maschinenlesbare Daten liegen strukturiert vor und können problemlos von Software gelesen und verstanden werden. Jeder*e Bürger*in, mit den notwendigen Kenntnissen, kann Software entwickeln, die diese Daten auswertet. Daher gilt es im Sinne der Transparenz, diese Daten öffentlich und kostenlos zugänglich und unter freien Lizenzvereinbarungen zur Verfügung zu stellen. So werden die Transparenz und das Vertrauen in die Verwaltung weiter gesteigert.

Offene Daten können unter anderem dazu beitragen, Falschinformationen zu vermeiden, indem diese kurz nach Veröffentlichung durch offene, offizielle Daten entkräftet werden können. Außerdem können durch offene Daten Innovationen gefördert werden. Neue Produkte können mithilfe der offen zugänglichen Daten erstellt und dank freier Lizenzen kommerzialisiert werden. Diese Produkte können beispielsweise auf geografische Daten zugreifen und in einer Kartenansicht dargestellt werden. Neue Visualisierungsformen der Daten, erstellt durch Bürger*innen sowie Unternehmen, können hochspezifische Bereiche abdecken und die offenen Daten so verständlicher vermitteln. Meinungen werden auf Datenbasis gebildet und Entscheidungsprozesse können objektiver ablaufen.

2.1.2 Unabhängigkeit

Durch Freie Software und freie Standards, die zum Teil bereits existieren, entsteht eine Unabhängigkeit von einzelnen Herstellern und Herstellerfirmen. Diese Unabhängigkeit kann Verwaltungen helfen, Entscheidungen freier treffen zu können. Sie werden digital souverän und können weitere, neue Faktoren bei der Softwareauswahl beachten, die bisher aufgrund einer Herstellerbindung durch proprietäre Software nicht berücksichtigt werden konnten.

Zum Beispiel kann ein größeres Augenmerk auf Benutzerfreundlichkeit, Barrierefreiheit und Aussehen der Software gelegt werden, da durch freie Standards kein „Quasi-Monopol“ eines Herstellers mehr besteht, der die Nutzung einer proprietären Software und deren Erscheinungsbild unveränderlich erzwingt. Außerdem bekommen durch offene Standards vermehrt kleinere Softwarehersteller die Möglichkeit, eigene Konkurrenzprodukte zu entwickeln, da diese nicht mehr von einzelnen großen Herstellern durch geschlossene Industriestandards ausgeschlossen und vom Markt verdrängt werden. Auch kann Freie Software durch offene Standards entwickelt, getestet und eingesetzt werden.

Freie Software unterstützt oftmals viele freie Standards. Der Quelltext ist frei einsehbar und veränderbar, sodass auch Dritte Veränderungen vornehmen können. Durch Beauftragung dieser Anbieter müssen Verwaltungen neue Funktionen nicht eigenhändig einbauen, dennoch können sie aber von Herstellern unabhängig bleiben, da der entstandene Quelltext erneut veröffentlicht werden muss und Veränderungen weiterhin gestattet sind. Zukünftige Anpassungen können daher trotzdem vorgenommen werden, ohne die Software wechseln zu müssen oder denselben Drittanbieter erneut zu konsultieren.

Mit proprietärer Software können Anpassungen häufig nur von dem Hersteller selbst implementiert werden, von dem die Verwaltung dann also abhängig ist. Dieser kann

jederzeit Lizenzmodelle ändern oder Funktionen entfernen, was zu einem Ausfall und einem erzwungenen Wechsel zu einer Konkurrenzsoftware führen kann. Ist die Software frei, werden Funktionen teils länger unterstützt, da sich mitunter ein Community-Mitglied findet, das die Funktionen selbst benötigt und daher weiterentwickelt.

2.1.3 Flexibilität durch Anpassungen

Freie Software bietet die Möglichkeit, Veränderungen in Form von sogenannten Forks (engl. *Gabelung*) vorzunehmen. Forks sind eine Abspaltung von dem ursprünglichen Projekt, welche in separaten Folgeprojekten weiterentwickelt werden. Hierbei wird eine Kopie des gesamten Quelltextes erstellt, sodass zwei unabhängige Versionen desselben Quelltextes vorliegen und voneinander getrennt bearbeitet werden können.

Forks müssen oftmals unter derselben Lizenz, die das ursprüngliche Projekt trägt, veröffentlicht werden. Die Software bleibt also trotz Veränderungen frei. Hierbei kann die Software an die konkreten Bedürfnisse der Behörde angepasst werden. So kann beispielsweise die lokale Rechtslage berücksichtigt werden, um bestehende offene Software verwenden zu können. Durch Anpassungen kann von einer Eigenentwicklung oder einer proprietären Lösung abgesehen werden. Offene Software kann außerdem in bestehende Systeme mit wenigen Anpassungen integriert werden, da offene Standards hinzugefügt werden können.

Aktuell sind die Integration und Kompatibilität mit bestehenden Systemen ein Ausschlusskriterium für proprietäre Software, die keine der bereits verwendeten Standards verwendet. Dieses Problem kann durch die Nutzung offener Software teilweise umgangen werden. Zusätzlich kann das gesamte System durch kleine Änderungen am Quelltext einheitlich gestaltet werden, um somit benutzerfreundlicher und wiedererkennbarer zu werden.

In Krisensituationen besteht schließlich die Möglichkeit, die Software an neue Gegebenheiten anzupassen. So kann mit hoher Geschwindigkeit eine Lösung für plötzlich auftretende Probleme gefunden und schnellstmöglich implementiert werden. Freie Software wird durch rapide wechselnde Umstände nicht unbrauchbar und wertlos, sondern erlangt weitere Funktionen, die bei der späteren Benutzung helfen können. Bei der Nutzung von proprietärer Software muss auf den Hersteller vertraut werden, der genügend Kapazitäten benötigt, um die Software schnellstmöglich an neue Gegebenheiten anzupassen. Trotz bestehender Kapazitäten kann das entstehende Produkt nicht mitentwickelt werden, wodurch Verzögerungen

entstehen können. Anpassungen von proprietärer Software können außerdem sehr teuer sein, wenn nur der Hersteller selbst Änderungen vornimmt.

(vgl. [Open Source in Kommunen \(5/2021\) \(kgst.de\)](https://www.kgst.de/documents/20181/34177/KGSt-Bericht-5-2021_Open-Source.pdf/994d10d0-ec25-f8ed-91af-1a12518c27d3) - https://www.kgst.de/documents/20181/34177/KGSt-Bericht-5-2021_Open-Source.pdf/994d10d0-ec25-f8ed-91af-1a12518c27d3, 2.2.2)

2.1.4 Ausfallsicherheit

Bei dem Einsatz eines jeglichen Softwareprodukts ist die Ausfallsicherheit ein wichtiges Merkmal. Häufig wird angenommen, dass Freie Software mehr Probleme mit Ausfällen hat als proprietäre Konkurrenzsoftware. Es gibt jedoch keinen Anlass, die Ausfallsicherheit von Freier Software, die von großen Communities (Entwicklergemeinschaften, die die Software betreuen und warten) gemeinsam entwickelt und getestet wird anders zu beurteilen als die von proprietärer Software. Softwareprodukte aus beiden Lizenzformen können als ähnlich stabil und funktional angesehen werden.

Freie Software bietet allerdings die Möglichkeit, dass auffallende Fehler von jedem behoben werden und so keinen Schaden verursacht wird. Experten sind in der Lage, den Quelltext zu bewerten und nachzuvollziehen, um Freie Software von außen einzuschätzen.

Offene Software kann einfach in mehreren Rechenzentren und auf verschiedenen Servern gleichzeitig eingesetzt werden, da keine zusätzlichen Lizenzgebühren anfallen und Konfigurationen problemlos übernommen werden können. So können bei einem Ausfall in einem Rechenzentrum die Server in dem anderen Rechenzentrum übernehmen und so ein stabileres System sichern. Diese Sicherheit durch den Einsatz von Clusterlösungen ist ein wesentlicher Bestandteil der Ausfallsicherheit für Software (freie und proprietär). Die entsprechenden Lizenzkosten machen in den Gesamtbetriebskosten jedoch nur einen geringen Anteil aus und wären den in der Regel anfallenden Subskriptionskosten für Open-Source-Software gegenüberzustellen.

Trotz des Vorteils, dass Freie Software nicht an Softwarehersteller gebunden ist, die ihr Produkt ausschließlich als SaaS (engl. Software as a Service), also nur in der Cloud, anbieten, wird für die Betriebsform im eigenen Rechenzentrum ein hoher Personalbedarf bei Administratoren und Entwicklern benötigt, der aufgrund des Fachkräftemangels nur schwer zu erfüllen ist. Strategisch ist daher die Betriebsform der Cloud ein zentrales strategisches Instrument der IT-Landschaft, in die sich Angebote von Freier Software integrieren müssen.

Hinzu kommt, dass bei Problemstellungen in proprietärer Software immer der Hersteller verantwortlich für die Problembhebung ist. Dieser muss den gemeldeten

Fehler zunächst reproduzieren, um ihn zu verstehen und zu lösen. Einen Fehler zu reproduzieren, kann mitunter viel Zeit in Anspruch nehmen, die bei Freier Software in die Problembeseitigung selbst investiert werden kann. Verwaltungen können durch den Einsatz von Freier Software größere Kontrolle bei Systemabstürzen und auftretenden Problemen erlangen, da sie die Problemquelle selbst identifizieren und lösen können. Hierzu ist jedoch eine hohe Expertise erforderlich, die zunächst neben den bestehenden Betriebsaufgaben aufgebaut werden muss. Entwickler und Administratoren müssen die Freie Software also verstehen und verändern können. Hierzu sind Schulungen und Weiterbildungen ein essenzielles Werkzeug, um Expertise innerhalb der Verwaltung aufzubauen.

2.1.5 Zusammenfassung

Transparenz und Vertrauen können bei Bürger*innen und Unternehmen geschaffen werden, sodass diese in die Lage versetzt werden, Verwaltungsprozesse und Abläufe zu verstehen. Darüber hinaus haben sie die Möglichkeit, sich einzubringen. Durch die Bereitstellung von offenen Daten auf öffentlichen Plattformen kann zusätzlich eine Transparenz geschaffen werden und Daten können zur weiteren Verarbeitung durch Dritte offen genutzt werden, zum Beispiel um eigene Erhebungen vorzunehmen und diese in Berichte einfließen zu lassen. Darüber hinaus können Falschinformationen durch diese Transparenz vermieden und Vertrauen in die Verwaltung und ihre Kernprozesse geschaffen werden.

Unabhängigkeit von Herstellern kann durch den Einsatz von Freier Software gewonnen werden. Dadurch entsteht die Möglichkeit, bei der Auswahl von Software andere Kriterien in den Fokus zu rücken und somit auch kleinere Software-Hersteller, die auf Freie Software setzen, zu berücksichtigen. Darüber hinaus ist man bei der Weiterentwicklung unabhängig, da die Software durch eigene Entwicklungen ergänzt oder verändert werden kann, wenn Sie benötigt werden.

Es besteht bei der Verwendung von Freier Software eine **Flexibilität durch Anpassungen**, die auf bestimmte Situationen und Anforderungen hin durchgeführt werden können. Bei der Anpassung von Rechtslagen muss proprietäre Software durch den Hersteller angepasst werden, was zu Verzögerungen oder hohem Aufwand führen kann. Bei der Nutzung von Freier Software kann auf diese Situationen schnellstmöglich reagiert werden, da die Anpassungen ggf. selbsttätig durchgeführt werden können. Auch auf Krisensituationen kann so reagiert werden und bestehende Software um Funktionen erweitert werden.

Häufig wird angenommen, dass Freie Software nur in einem geringen Maße eine **Ausfallsicherheit** bietet. Aufgrund der Lizenz-Gestaltung können beliebig viele

Instanzen dieser Software betreiben werden, jedoch bietet auch proprietäre Software einen hohen Grad an Ausfallsicherheit und die Lizenz-Mehrkosten machen nur einen geringen Teil der gesamten Betriebskosten aus. Dass auf Probleme schneller reagiert wird, da diese eigenverantwortlich oder durch eine Entwicklergemeinschaft behoben werden können und eine Problem-Meldung nicht erst an den Software-Hersteller gegeben werden muss, ist von Vorteil. Hersteller proprietärer Software müssen zunächst eine aufwändige Analyse durchführen, um mit der Problembehebung zu beginnen. Diese freie Zeit könnte durch entsprechende Experten bereits genutzt werden, um die Fehleranalyse selbst durchzuführen.

Für alle diese Punkte ist jedoch ein hoher Spezialisierungsgrad notwendig und eine hohe Expertise gefordert. Entwickler*innen und Administrator*innen müssten also zunächst geschult oder eingestellt werden und notwendige Werkzeuge müssen kennengelernt werden, um innerhalb der Verwaltung von diesen Potenzialen Gebrauch zu machen.

2.2 Best Practice

2.2.1 Ausgaben

Häufig werden die Einsparungen als eines der wichtigsten Argumente für den Einsatz von Freier Software genannt. Hier werden die fehlenden Lizenzgebühren aufgeführt, die bei proprietärer Software sehr hoch ausfallen können.

Lizenzgebühren sind jedoch nicht die einzigen Kostenpunkte im Einsatz von Freier Software. Da Freie Software von der Community oft als Prototyp ohne dauerhafte Funktionsgarantie veröffentlicht und offen kommuniziert wird, dass die Software „so, wie sie ist“ („as is“) bereitgestellt wird, muss eine stabile Version dieser Freien Software angefertigt werden.

Um eine Prototypensoftware für den dauerhaften Betrieb anzupassen, werden jedoch neue Entwicklerkapazitäten benötigt. Hier entstehen Mehrkosten, die bei proprietärer Software vermieden werden können. Außerdem müssen weitere Administrator*innen und Supportmitarbeiter*innen eingestellt werden, wann immer eine neue Freie Software eingeführt wird, um eine Überlastung der Mitarbeiter*innen zu vermeiden, die für die Freie Software verantwortlich sind. Die Erstellung neuer Schulungsunterlagen kann ebenso Kosten verursachen, die bei der Nutzung von proprietärer Software vermieden werden können. Support- und Pflegeverträge, sog. „Subscriptions“, sind weitere Ausgaben, die bei Freier Software getätigt werden müssen, um einen reibungslosen Einsatz dieser sicherzustellen. Bei weniger essenzieller Freier Software kann gegebenenfalls auf Support- und Pflegeverträge verzichtet werden. Eine Umstellung auf Freie Software ist zu Beginn demnach teurer

als die Nutzung proprietärer Lösungen, jedoch kann sich die Verwaltung durch Freie Software unabhängig von großen Anbietern machen und auf lange Sicht Kosten reduzieren, indem freiere Entscheidungen getroffen werden können. Mehrkosten müssten also zunächst hingenommen werden, um die Einführung von Freier Software gewissenhaft durchführen zu können.

(vgl. [OpenData: bisschen Prototyp, und das war's dann? - binary butterfly \(binary-butterfly.de\) - https://binary-butterfly.de/artikel/opendata-bisschen-prototyp-und-das-wars-dann/](https://binary-butterfly.de/artikel/opendata-bisschen-prototyp-und-das-wars-dann/), abgerufen am 28.01.2022)

2.2.2 Entwickler*innen und Administrator*innen

2.2.2.1 Notwendigkeit

Mit dem geplanten Einsatz von Freier Software und der damit verbundenen Einführung und Administration (Betreuung von Softwareprodukten) der neuen Systeme entsteht eine Notwendigkeit für neues Fachpersonal, da der Simultanbetrieb der Freien Software und der proprietären Software in der Einführungsphase eine Doppelbelastung für Administrator*innen und Entwickler*innen bedeutet. Ein Simultanbetrieb ist zunächst notwendig, um alle Informationen und Dateien auf die Freie Software migrieren zu können. Außerdem benötigen Nutzer*innen Zeit, um sich an die Änderungen zu gewöhnen und diese anzunehmen. Neue Stellen müssten geschaffen oder entwickelt werden, um die großen Belastungen, die mit Freier Software einhergehen, bewältigen zu können. Wenn entsprechende Expertise in der Verwaltung fehlt, ist davon auszugehen, dass Freie Software weniger stabil funktioniert als proprietäre Konkurrenzprodukte. Freie Software wird in den meisten Fällen in kleineren, spezialisierten Paketen veröffentlicht.

Eine proprietäre Lösung kann weniger Aufwand bedeuten, da sie viele Anwendungsfälle auf einmal löst. Weniger Fachpersonal ist gefordert, da insgesamt weniger Software installiert und betrieben werden muss. Um Freie Software also anwenden zu können, braucht eine Verwaltung neue Entwickler*innen und Administrator*innen.

Sobald Freie Software jedoch in Benutzung ist, hat die Verwaltung größere Möglichkeiten und Freiheiten in der Auswahl und Konfiguration der Software. So könnten verschiedene kleine Softwarepakete so zusammengestellt werden, dass das System exakt auf die Anwendungsfälle in der entsprechenden Verwaltung zugeschnitten wäre. Außerdem kann bei auftretenden Fehlersituationen eine interne Problemlösung stattfinden, da Entwickler*innen und Administrator*innen entsprechende Erfahrungen mitbringen und mit dem Quelltext vertraut sind. Probleme können so schneller und effizienter gelöst werden.

2.2.2.2 Problematik

Um die notwendigen, neuen Entwickler*innen, die entsprechende Qualifikationen mitbringen, anzuwerben, muss die öffentliche Verwaltung Werbung machen, um nicht in der Fülle großer IT-Konzerne (Google, Facebook, Microsoft, SAP, T-Systems) unterzugehen. Da in Bildungseinrichtungen immer mehr mit Freier Software gearbeitet wird, beginnen Entwickler*innen schon früh in ihrer Ausbildung, Freie Software zu verwenden. Viele Entwickler*innen bevorzugen schon heute Freie Software und wählen ihren zukünftigen Arbeitgeber dementsprechend aus. Viele junge Entwickler*innen engagieren sich in freien Software-Communities und entwickeln Freie Software aktiv mit.

Damit die öffentliche Verwaltung diesen Trend nicht verpasst, muss Freie Software auch bei diesen weitreichend adaptiert werden. Die Industrie hat diesen Trend erkannt und wirbt mit eigenen freien Softwareprojekten, wie „Tensorflow“ von Google, „React“ von Facebook oder „Visual Studio Code“ von Microsoft. Die Unternehmen profitieren selbst von diesen Projekten, da Freiwillige an ihrer Infrastruktur mitarbeiten und so neue Entwickler*innen angeworben werden.

Die öffentliche Verwaltung kann von diesem Trend profitieren, indem sie ebenfalls in der Freien Software aktiv wird, um dadurch Digitale Souveränität zu erhalten.

Freie Software ist ein Weg, für sich zu werben und auf sich aufmerksam zu machen. Neben der fehlenden Freien Software fallen branchenübliche Gehälter oft höher aus als in den Behörden. Um weiterhin fähiges Personal anwerben zu können, müssen Gehälter angepasst werden. Kann die Verwaltung zukünftig nicht für sich werben, führt der Fachkräftemangel in die Abhängigkeit von großen Herstellerunternehmen, die mitunter hohe Preise für Software verlangen können.

2.2.3 Geschwindigkeit

Aufgrund der Bedeutung von Freier Software sind die Integration in bestehende Systeme und der ausschließliche Einsatz dieser in naher Zukunft äußerst wünschenswert. Hierbei ist jedoch anzumerken, dass eine Einführung über einen zu kurzen Zeitraum zu Problemen führen wird. Ist eine Einführung verfrüht oder in einem zu kurzen Zeitraum angedacht, kann das System nicht korrekt konzipiert und aufgesetzt werden. Wenn das System vor einer flächendeckenden Einführung, wie sie immer wieder gefordert wird, nicht gründlich getestet und erprobt wird, kann ein stabiler Betrieb der Freien Software nicht sichergestellt werden.

Fehlende Tests und Bewertungen können zu Ausfällen und Systemabstürzen führen. Auch muss die Datenintegrität vor Einführung geprüft werden, da die Daten unter

keinen Umständen von Unbefugten abrufbar oder veränderbar sein dürfen. Zudem muss die Akzeptanz der Nutzer*innen nach und nach erarbeitet werden, da andernfalls eine Ablehnungshaltung entstehen könnte. Administrator*innen benötigen ebenfalls Zeit, um sich mit der neuen Software vertraut zu machen und den Umgang mit ihr zu erlernen. Sie müssen das neue System genauso gut kennen wie das alte, um im Fall einer Frage zügig helfen zu können. Da in der Einführungsphase zwei Softwaresysteme gleichzeitig betrieben werden müssen, sollte dringend zusätzliches Personal eingestellt werden.

Der Parallelbetrieb der Softwareprodukte ist in der Übergangsphase oft notwendig, um alle Daten auf das neue System übertragen zu können. Weitere Administrator*innen bedeuten weitere Kostenpunkte, die für den Einsatz von Freier Software unumgänglich sind. Eine Freie Software erstmals einzuführen, beansprucht mehr Zeit als proprietäre Software, jedoch wird Expertise in der eigenen Verwaltung aufgebaut, was langfristig zu höherer Qualität und schnellerer Problembehebung führt. Eine Einführung von Freier Software ist demnach ein langsamer, fortwährender Prozess, der Zeit benötigt, um korrekt durchgeführt zu werden. Ein Umstellen der Verwaltung auf Freie Software kann nicht über Nacht durchgeführt werden.

2.2.4 Einer für Alle

Das Einer-für-Alle-Prinzip (EfA-Prinzip) beschreibt die zentrale Entwicklung einer Software und einen anschließenden zentralen Betrieb dieser. Bund, Länder und Kommunen stimmen sich ab, wer welche Leistung digitalisiert. Anschließend können fertige Softwarelösungen von anderen genutzt werden. Zur Nutzung müssen nur noch Schnittstellen miteinander verbunden werden, sodass benötigte Daten ausgetauscht werden können.

Dieses Prinzip wurde beispielsweise bereits vom Main-Taunus-Kreis angewandt. Es wurde eine digitale Führerscheinlösung von der Stadt Mannheim in Baden-Württemberg nachgenutzt. Freie Software bietet sich besonders gut an, da die zentralen Lösungen somit von Außenstehenden mitentwickelt werden können und sofort auf Feedback eingegangen werden kann, um das bestmögliche Produkt zu erschaffen. Nachnutzende Behörden können nachvollziehen, was mit Daten passiert und wie mit diesen umgegangen wird.

Da eine große Menge an OZG-Leistungen digitalisiert werden muss und eigene Lösungen von jeder Verwaltung zu aufwändig und zu kostenintensiv sind, bietet sich im Zusammenhang mit der Umsetzung des OZG das EfA-Prinzip an. Um es später produktiv nutzen zu können, muss sichergestellt werden, dass jede dieser Anwendungen gleich aussieht und funktioniert. Andernfalls entstehen viele separate

Lösungen, die nur schwer gemeinsam nutzbar sind. Benutzeroberflächen sollten also genormt werden und einem einheitlichen Design folgen. Auch müssen Standards eingehalten und Anbindungen an verschiedenen Schnittstellen bedacht werden. Eine Nachnutzung macht nur Sinn, wenn die Software in das bestehende System integriert werden kann. Bei der Entwicklung einer solchen Software nach dem EfA-Prinzip müssen viele verschiedene Voraussetzungen von verschiedenen Kommunen, Kreisen und Ländern berücksichtigt werden.

2.2.5 Beitrag zu Freier Software

2.2.5.1 Weiterverbreitung von Anpassungen

Da Verwaltungen oftmals ähnliche Aufgaben erfüllen, kann das Recht, Freie Software weiterzuverbreiten, von großem Nutzen sein. Nachdem eine Anpassung am Quelltext vorgenommen wurde, wird dieser als ein neuer Fork veröffentlicht und anderen Verwaltungen zur Verfügung gestellt. Anpassungen dienen nicht ausschließlich der eigenen Verwaltung, sondern jeder Organisation, die die Software einsetzen möchte. So kann in kleineren Verwaltungen Mehraufwand reduziert werden, da bereits angepasste Systeme zu finden sind. Entstandene Forks können dem Ausgangsprojekt wieder hinzugefügt werden, um die Fähigkeiten der ursprünglichen Software zu steigern und die neuen Funktionen allen Nutzern zur Verfügung zu stellen. Um eine Integration in die ursprüngliche Software zu ermöglichen, ist eine enge Kooperation mit den Entwickler*innen der Software erforderlich.

Diese Kooperation sollte von Beginn an gepflegt werden, da Entwickler*innen auch bei der Problemlösung helfen können. Die eigene Verwaltung kann schließlich auch auf fertig angepasste Software zurückgreifen und so Kosten reduzieren. Diese angepassten Versionen von Freier Software können zum Beispiel auf den Servern des Projektes „Ein Ort für öffentlichen Code“ geteilt werden. Verwaltungen sollen sich auf dieser Plattform über Freie Software austauschen können und Erfahrungen miteinander teilen.

Wenn eine Verwaltung also eine angepasste Version einer freien Software teilt, sollten alle Verwaltungen mit dem gleichen Problem auf dieser Plattform die entsprechende Software finden. Wenn eine bestehende Software gefunden wurde, muss diese nur noch installiert werden, da kein oder nur sehr geringer Anpassungsaufwand besteht.

2.2.5.2 Unterstützung durch Pflege- und Supportverträge

Um die weitere Entwicklung und den Fortbestand einer Freien Software zu sichern, sind Pflege- und Supportverträge in Betracht zu ziehen. Pflege- und Supportverträge können mit Drittanbietern, aber auch mit dem freien Softwareprojekt selbst, abgeschlossen werden. In diesen Verträgen können zum Beispiel Ausfallsituationen

oder Erweiterungen geregelt werden. Wenn eine Freie Software essenziell für die Aufrechterhaltung einer Verwaltung ist, so zum Beispiel Serverbetriebssysteme auf Basis von Linux, können Pflege- und Supportverträge als Absicherung dienen. Sie sind hilfreich bei der Sicherung der Funktionalität der Software, da die Möglichkeit, auf professionelle Expertise zurückzugreifen, bei der Problemfindung und Problemlösung notwendig sein kann.

Obwohl Freie Software weiterhin kostenfrei bleibt, können durch diese Verträge die Community, die hinter einem Projekt steht und dieses aktiv entwickelt, aber auch die Anwender*innen, die eventuell auf die Hilfe und den Support der Entwickler*innen angewiesen sind, unterstützt werden. Support- und Pflegeverträge sind ein Werkzeug, um die Nutzung Freier Software möglich zu machen.

Der Einsatz der Freien Software ist demnach weiterhin mit Kosten verbunden, jedoch sind diese Kosten oftmals geringer als bei proprietärer Software, die von großen Anbietern entwickelt wird. Zudem kann von diesen Investitionen die gesamte Community und jede Verwaltung profitieren, die die Software einsetzen, da durch eine Finanzierung des Projektes höhere Qualität und neue Funktionen ermöglicht werden.

2.2.6 Kooperation

2.2.6.1 Zwischen Verwaltungen

Sollte sich herausstellen, dass keine Freie oder proprietäre Software für die Lösung eines Problems in Frage kommt, sollte bei der Entwicklung von Freier Software immer eine Kooperation mit anderen Kommunen und Stadtverwaltungen in Betracht gezogen werden. Häufig bestehen ähnliche Probleme und Anforderungen, die durch eine gemeinsam entwickelte oder angepasste Software gelöst werden können. Eine Kooperation kann über verschiedene, bereits bestehende freie Plattformen durchgeführt werden.

Eine solche Plattform ist „ein Ort für öffentlichen Code“, die Verwaltungen erlaubt, Ergebnisse ihrer Freien Software zu veröffentlichen. Essenziell für die gemeinsame Entwicklung ist die Nutzung einer Versionsverwaltung, wie zum Beispiel „git“, die unter der „GNU General Public Licence Version 2“, einer offenen Lizenz, verfügbar ist. Die Versionsverwaltung sorgt für eine geordnete Zusammenarbeit, indem einzelne Veränderungen am Quelltext dokumentiert und gespeichert werden. Bei der Entwicklung kann jeder*e Entwickler*in erkennen, wer für welche Änderung verantwortlich ist und diese Änderung notfalls zurücksetzen.

Eine Versionsverwaltung wird von den Communities, die Freie Software entwickeln, bereits verwendet, da die meisten Projekte auf einer Plattform wie „GitHub“ oder

„Iterate faster, innovate together | GitLab“ entwickelt werden. Durch eine Kooperation kann Arbeit zwischen kleineren, aber auch größeren Verwaltungen aufgeteilt werden. Die Kooperationspartner stellen jeweils Ressourcen und Personal zur Verfügung, die an der neuen Software arbeiten. Kosten können durch eine gemeinsame Entwicklung verringert werden, da sie von vielen gleichzeitig getragen werden und eine entsprechende Software auch ohne Kooperation entwickelt werden muss. Durch Zusammenarbeit können außerdem größere, zeitaufwändige Projekte durchgeführt werden, die andernfalls nur schwer umsetzbar sind. Entwicklungsressourcen werden effizienter verwendet und pro Verwaltung sind weniger Entwickler*innen notwendig.

(<https://about.gitlab.com/>)

2.2.6.2 Organisationen und Bürger*innen

Um Freie Software besser integrieren zu können, sollten unabhängige Organisationen und Bürger*innen in den Entwicklungs- und Auswahlprozess bestimmter Software mit einbezogen werden. Durch die Unterstützung Außenstehender, können Konzepte schneller erdacht und die Qualität der Freien Software gesichert werden. Um eine gute Zusammenarbeit zu ermöglichen, müssen alle Beteiligten auf Augenhöhe miteinander kommunizieren und bei jedem Austausch und bei jeder Forderung die Machbarkeit berücksichtigen.

Schon seit einiger Zeit existieren Organisationen, die sich intensiv mit der Digitalisierung von Behörden und Verwaltung auseinandersetzen. Diese Organisationen in die Entwicklung neuer Konzepte und Freier Software mit einzubeziehen, kann Verwaltungen entlasten und eine höhere Qualität sichern. Da viele Mitglieder in entsprechenden Organisationen Expert*innen auf Gebieten in der IT-Branche sind, können sie wichtige Hilfestellungen leisten und hilfreiche Einblicke gewähren. Unabhängige Organisationen können zudem helfen, bestehende Freie Software zu untersuchen und zu bewerten, bevor sie eingesetzt wird.

Durch die Unterstützung können Bürger*innen direkten Einfluss nehmen, was Vertrauen und Transparenz steigert. Designer für Benutzerschnittstellen („UX-Designer“) können die Abläufe und Routinen bei der Programmnutzung analysieren und optimieren und so helfen, die bestmögliche Nutzungserfahrung zu ermöglichen.

Eine solche Zusammenarbeit lässt sich bereits bei der „Corona-Warn-App“ beobachten, die gemeinsam mit Privatpersonen und Organisationen entwickelt und konzipiert wurde. So wurde von Beginn an Vertrauen in die App aufgebaut. Dieses Vertrauen ist in vielen Fällen wichtig, gerade wenn die Software mit personenbezogenen Daten arbeiten soll. Diese Einbeziehung von Organisationen und Bürger*innen kann die Entwicklung und den Auswahlprozess Freier Software deutlich

verlangsamen, jedoch hat das Produkt eine höhere Qualität und die Auswahlprozesse sind öffentlich und transparent.

(Open-Source-Projekt Corona-Warn-App (coronawarn.app) - <https://www.coronawarn.app/de/>)

2.2.7 Beispiele

2.2.7.1 OParl

OParl ist ein freier Standard, der eine maschinenlesbare Schnittstelle zu Ratsinformationssystemen bietet. Der Standard wurde 2018 von der „Open Knowledge Foundation Deutschland e.V.“ in der Version 1.1 vorgestellt. Die Dokumentation ist öffentlich abrufbar und kann frei implementiert werden. So haben Bürger*innen die Möglichkeit, alle Daten jederzeit auszulesen und zu verarbeiten.

Informationen können effektiver aus dem Ratsinformationssystem herausgezogen werden und zur Meinungsbildung beitragen. Aktuelle Ratsinformationssysteme sind oftmals kompliziert zu benutzen und bieten nur begrenzte Zugangsmöglichkeiten.

Trotz Nutzung von OParl muss sichergestellt werden, dass keine sensiblen oder personenbezogenen Daten bereitgestellt und öffentlich abrufbar gemacht werden. OParl kommt bereits in verschiedenen Verwaltungen, wie zum Beispiel München, zum Einsatz. Oftmals müssen Verwaltungen lediglich bei dem Softwarehersteller, der das verwendete Ratsinformationssystem entwickelt, eine OParl-Einbindung hinzubuchen.

Verwaltungen bieten durch OParl eine effektive freie Datenschnittstelle, die von Bürger*innen verwendet werden kann. Auch die Verwaltung selbst kann von OParl profitieren, wenn zum Beispiel die Oberfläche, zum Abrufen der Daten, von einem anderen Hersteller entwickelt wird oder Freie Software eingesetzt wird.

„Ziel von OParl ist somit, Anwendungen zu ermöglichen, die mehr Transparenz schaffen und durch die nun für Bürger besser verfügbaren Informationen mehr Mitbestimmung möglich zu machen. Bürger*innen können mit OParl-basierten Anwendungen eine geeignetere Entscheidungsgrundlage für ihr gesellschaftliches Handeln bekommen und frühzeitig Planungsprozesse z.B. von Großprojekten begleiten.“

(OParl für Kommunen - OParl. - <https://oparl.org/oparl-fuer-kommunen/>, abgerufen am 25.01.2022)

2.2.7.2 Ein Ort für öffentlichen Code

„Ein Ort für öffentlichen Code“ ist ein Projekt, das sich zum Ziel gemacht hat, eine Plattform zum Austausch von Freier Software zu erschaffen, die von Verwaltungen, Fachleuten, aber auch von Bürger*innen eingesehen werden kann. Die Plattform soll

eine Alternative zu aktuellen Plattformen, wie „GitHub“, bieten, bei denen viele rechtliche Bedenken bestehen.

Benutzer*innen müssen unter anderem den AGB des Anbieters zustimmen. Das Projekt soll neben dem Austausch von Quelltext auch einen Austausch von Erfahrungen und Listen mit bestehender Freier Software ermöglichen. Das Projekt wurde im Februar 2022 der Öffentlichkeit zugänglich gemacht. Um Quelltext effektiv austauschen zu können, ist eine Versionsverwaltung von großer Wichtigkeit. Durch sie sind Updateprozesse transparent einsehbar und es kann sichergestellt werden, dass dauerhaft die richtige Version verwendet und veröffentlicht wird.

Damit Kooperationen auf der Plattform stattfinden können, müssen Entwickler*innen und Anwender*innen ein tiefes Verständnis der Versionsverwaltung haben. Die bekannteste Versionsverwaltung ist „git“. Die Plattform löst jedoch nicht das Problem, dass der Quelltext dokumentiert werden muss, um in fremden Verwaltungen eingesetzt und verändert werden zu können. Einen Quelltext zu dokumentieren, bedeutet, im Quelltext selbst Anweisungen mit Informationen zu versehen, die die Software nicht beeinflussen, aber Aufschluss über die Funktion des Programmes geben. Ist der Quelltext nicht dokumentiert, können Entwicklerteams diesen nicht nachvollziehen und verändern. Auch die Mitarbeit der Öffentlichkeit wäre sonst ausgeschlossen. Veränderungen am Quelltext sollten demnach gewissenhaft dokumentiert werden.

„Ziel des Interessenverbands ist die Wegbereitung für ein Portal, durch das die öffentliche Verwaltung in Deutschland Open-Source-Software (OSS), oft auch als "Freie Software" oder "FOSS" bezeichnet, in adäquater und rechtssicherer Weise austauschen und gemeinsam entwickeln kann.“

(KGSt | Ein Ort für öffentlichen Code: Ein Code Repository für die öffentliche Hand - <https://www.kgst.de/ein-ort-fur-oeffentlichen-code>, abgerufen am 25.01.2022)

(Bundesregierung plant Portal für Open-Source-Software in der Verwaltung | heise online - <https://www.heise.de/news/Bundesregierung-plant-Portal-fuer-Open-Source-Software-in-der-Verwaltung-6124712.html>)

2.2.7.3 **Geo-Software Esri und QGis**

QGis ist ein freies geographisches Informationssystem mit vielen professionellen Funktionen. Die Software ist erweiterbar und bietet eine Schnittstelle für Entwickler*innen. QGis verfügt über eine starke Community, die sich mit der Entwicklung des Projektes beschäftigt. Das Projekt wird von verschiedenen Organisationen, Verwaltungen und Firmen unterstützt und wird aktiv weiterentwickelt. Eine Adaption könnte das „Quasi-Monopol“ der Herstellerfirma ESRI mit ihren „ArcGIS“ Produkten lösen und so die Unabhängigkeit steigern.

Bei einer Migration ist wichtig zu beachten, dass Nutzer*innen den direkten Vergleich zwischen QGIS und ArcGIS machen können und Unterschiede aufgelistet werden. Problematisch ist das aktuelle „Quasi-Monopol“ bei einer Migration dennoch, da ESRI nicht nur die Anwendungssoftware, sondern auch Cloud- und Serversoftware anbietet. Einige der Dienste sind durch Freie Software zu ermöglichen, jedoch ist bei der Nutzung von Freier Software mit einer Vielzahl von verschiedenen Projekten zu rechnen. Dennoch kann QGIS mit entsprechenden Zusatzprojekten vergleichbare Funktionen bieten.

([Die Nutzung von QGIS in der Geodateninfrastruktur der Stadt Uster](https://www.qgis.org/de/site/about/case_studies/suisse_uster.html)
https://www.qgis.org/de/site/about/case_studies/suisse_uster.html)

([Willkommen beim QGIS Projekt! - https://qgis.org/de/site/](https://qgis.org/de/site/))

2.2.7.4 **Bayerische Ehrenamtskarte**

Die Bayerische Ehrenamtskarte ist eine der ersten Leistungen im FIT-Store. Der FIT-Store wird von der Föderalen IT-Kooperation (kurz: FITKO) betrieben und soll eine zentrale Sammelstelle für Einer-für-Alle-Software werden. Die App für die Ehrenamtskarte wurde von einem Unternehmen in Zusammenarbeit mit sechs Studenten der Universität Augsburg entwickelt und unter der freien MIT-Lizenz auf der Plattform „GitHub“ veröffentlicht. Zur Installation benötigt es keine Anmeldung bei der Plattform „GitHub“. Eine Ehrenamtskarte kann in der App angefragt werden. Die Verwaltung kann anschließend den Antrag auf einem Web-Portal bearbeiten. Personenbezogene Daten werden auf dem privaten Smartphone und zur Verifikation nur in gehashter Form auf einem Server gespeichert. Das Hashen von Daten funktioniert nur in eine Richtung, sodass keine Rückschlüsse zu personenbezogenen Daten gemacht werden können.

Die App kann außerdem Geschäfte, die eine Vergünstigung für Personen mit Ehrenamtskarte anbieten, in einer Kartenansicht darstellen. Bei der Umsetzung der Software haben Studenten mit einem privaten Unternehmen und einer Behörde zusammengearbeitet. Eine solche Kooperation bietet großes Potenzial, da jede Partei profitiert. Die Studenten werden beim Erstellen einer freien Software unterstützt, das Unternehmen kann die Freie Software als SaaS-Lösung anbieten und die Behörde die Software einsetzen.

([Ehrenamtskarte OpenSource Leistungsbeschreibung.pdf \(fitko.de\)](https://www.fitko.de/fileadmin/fitko/produktmanagement/fit-store/leistungen/Ehrenamtskarte_OpenSource_Leistungsbeschreibung.pdf) -
https://www.fitko.de/fileadmin/fitko/produktmanagement/fit-store/leistungen/Ehrenamtskarte_OpenSource_Leistungsbeschreibung.pdf)

2.2.7.5 **Palim! Palim! – Stadt Bühl**

Die Stadt Bühl veröffentlichte im Zuge der Corona-Pandemie und den damit einhergehenden Kontaktbeschränkungen die Online-Videokonferenzplattform „Palim! Palim!“. Sie soll Bürger*innen die Möglichkeit geben, sich trotz Kontaktbeschränkungen zu treffen. Die Plattform basiert auf dem Open-Source-Projekt „Jitsi“.

„Jitsi“ ist eine freie Videokonferenz-Software basierend auf dem WebRTC-Standard, der eine Verbindung zweier Geräte untereinander beschreibt. „Jitsi“ ist ein Unternehmen, das sein Projekt im Jahr 2003 begonnen hat und die Videokonferenz-Plattform auch als SaaS anbietet. Von der Stadt Bühl wurde eine eigene Benutzeroberfläche mit dem Logo der Stadt entwickelt. Allgemeine Daten zu dem Projekt sind öffentlich über ein Echtzeit-Portal einsehbar und zeigen Auslastung und Nutzerzahlen. Die Videokonferenzplattform wird gut angenommen und von vielen Bürger*innen verwendet. Projekte, die jede*n Bürger*in betreffen, haben demnach ein hohes Potenzial und große Erfolgchancen. Städte können zukünftig Webseiten und Dienste bauen, die jedem helfen können.

Im Zuge der Corona-Pandemie entstehen vermehrt konkrete Probleme, die die Kommunen lösen können, wie die Videokonferenzplattform „Palim! Palim!“ zeigt.

(Jitsi Meet - <https://meet.jit.si/>)

(Palim! Palim! | Stadt Bühl - <https://www.buehl.de/digitalisierung/palim-palim>)

2.2.7.6 **Offene Daten Portal Stadt Frankfurt am Main**

Die Stadt Frankfurt stellt auf der Seite Willkommen - Offene Daten Portal der Stadt Frankfurt/Main offene Datensätze in einem maschinenlesbaren Format zur Verfügung. Die Daten sind frei nutzbar und eine API-Schnittstelle wird angeboten. Insgesamt wurden bereits mehr als 100 Datensätze veröffentlicht. Eine ähnliche Plattform würde sich auch in Dortmund anbieten. Datensätze können veröffentlicht und über eine freie Schnittstelle abrufbar gemacht werden. Offene Daten können zum Beispiel für neue Apps, Bildungszwecke oder Forschungsprojekte eingesetzt werden.

([Willkommen - Offene Daten Portal der Stadt Frankfurt/Main](#), abgerufen am 03.02.2022)

2.2.8 **Zusammenfassung**

Ein häufig aufgezeigter Vorteil Freier Software ist auch, dass im Bereich der **Ausgaben** diese gegenüber proprietärer Software günstiger ist. Dies ist jedoch detailliert im Einzelfall zu betrachten. So machen die Lizenzkosten in der Regel nur einen geringen Bruchteil der Gesamtkosten aus. Bei der Erstellung und dem Betrieb von Freier Software kann es durchaus zu Mehrkosten kommen, da

Schulungsunterlagen, Anpassungen der Software an die eigenen Bedürfnisse sowie Support- und Pflegekosten hinzugerechnet werden müssen und diese insgesamt höher ausfallen können als bei proprietärer Software. Auch die Kosten für die eigenen Administrator*innen, Entwickler*innen und Supportmitarbeiter*innen können bei der Gesamtbetrachtung dazu führen, dass Freie Software erst auf lange Sicht und bei einem langfristigen Einsatz zu Kostenersparnissen führt.

Insbesondere der letztgenannte Punkt des Personals im Bereich der **Administrator*innen und Entwickler*innen** stellt Verwaltungen bereits heute vor große Herausforderungen. Um Freie Software für die Verwaltung effizient einsetzen zu können, besteht die **Notwendigkeit**, dieses Fachpersonal anzuwerben. Damit dies gelingen kann, muss die Verwaltung die **Problematik** erkennen und auflösen, dass potenzielle Mitarbeiter*innen durch große Technologieunternehmen oder Unternehmen der freien Wirtschaft frühzeitig angeworben werden. Mit der Einführung von Freier Software muss die Verwaltung für sich werben und bereits frühzeitig potenzielle Mitarbeiter*innen anwerben. Die Zusammenarbeit mit Entwicklungsgemeinschaften und Universitäten muss daher intensiviert werden. Aber auch die branchenüblichen Gehälter, die gezahlt werden, müssen durch die Behörden adaptiert werden. Kompetentes Personal kann nur gewonnen werden, wenn Behörden in Konkurrenz mit Unternehmen aus der Technologiebranche treten können. Um das Interesse dieser Personengruppe zu bekommen, kann die Nutzung und eine flächendeckende Strategie für Freie Software ein Weg sein, auch um dem Fachkräftemangel in diesem Bereich entgegenzutreten.

Ein weiterer Aspekt, der Betrachtung finden muss, ist, dass Freie Software in einer annehmbaren **Geschwindigkeit** eingeführt wird. Eine Umstellung der gesamten Infrastruktur über einen zu kurzen Zeitraum kann zu Problemen führen. Daher ist es sicherlich notwendig, einen Parallelbetrieb und eine Übergangsphase einzuplanen, da davon maßgeblich der Erfolg von Freier Software abhängt.

Ein Prinzip, das dazu beitragen kann, ist das **Einer-für-Alle**-Prinzip, wonach interkommunal eine Leistung abgestimmt und durch einen Akteur entwickelt wird. Durch die Entwicklung auf Basis von Freier Software müssen zur Nutzung und Adaption für die eigene Verwaltung nur noch Schnittstellen angepasst werden. Diese Variante der Entwicklung bietet viele Möglichkeiten, jedoch müssen vorab durch die verschiedensten Stellen auf allen Ebenen die entsprechenden Voraussetzungen und "Leitplanken" für diese Art von Entwicklung definiert werden.

Nicht zuletzt kann damit auch ein **Beitrag zur Freien Software** geschaffen werden, schließlich haben Verwaltungen oftmals ähnliche Aufgaben. Daher kann durch die **Weiterverbreitung von Anpassungen** der Freien Software ein Nutzen für andere

entstehen. Eine gute Plattform wurde erst kürzlich dafür geschaffen (Open-Code-Plattform der KGSt). Sollte darüber hinaus der Bedarf an **Unterstützung durch Pflege- und Supportverträge** bestehen, lassen sich diese mit Drittanbietern oder Organisationen schließen. Durch diese haben Verwaltungen eine erweiterte Sicherheit und es lassen sich Ausfälle absichern. Darüber hinaus wird durch diese Verträge die Entwicklung der jeweiligen Software sichergestellt, obwohl die Software weiterhin kostenfrei bleibt.

Abschließend kann also hervorgehoben werden, dass **Kooperationen zwischen Verwaltungen** oder **Organisationen und Bürger*innen** immer in Betracht gezogen werden sollten. Oft stehen Verwaltungen vor den gleichen Herausforderungen und eine Kooperation kann dazu führen, dass eine Software gemeinsam entwickelt wird. Hierzu stehen bereits mehrere Plattformen zur Verfügung. Weiterhin können bei diesen Entwicklungen Organisationen und Bürger*innen hinzugezogen werden. Dies schafft einerseits Vertrauen und Transparenz, andererseits kann auf das Wissen über die Entwicklung Freier Software von bestimmten Organisationen zurückgegriffen werden. Ein Beispiel für diese Kooperation ist die Corona-Warn-App, die in diesem Format entwickelt wurde.

2.3 Kritische Erfolgsfaktoren

2.3.1 IT-Sicherheit und Datenschutz

Wichtige Erfolgsfaktoren sind die IT-Sicherheit und der Datenschutz der Freien Software. Hierzu zählen sämtliche rechtliche Rahmenbedingungen im Zusammenhang mit Datenschutz, so zum Beispiel die DSGVO der Europäischen Union oder das Datenschutzgesetz NRW (DSG NRW). Vor dem Einsatz der Software muss demnach bewertet werden, ob diese den rechtlichen Ansprüchen genügt und im Zweifelsfall muss entweder ein Ersatz gefunden oder es müssen Änderungen an der Software, in Form eines Forks, vorgenommen werden. Positiv ist, dass Bürger*innen und Expert*innen beim Einsatz von Freier Software selbst eine Bewertung der Software vornehmen können. Hintertüren können also nicht von einem Hersteller unbemerkt eingebaut werden.

Bei proprietären Lösungen ist eine Hintertür nicht auszuschließen, da der Hersteller allein Zugriff auf den Quelltext hat, da dieser als Betriebsgeheimnis gewertet wird. Bei Cloud-Lösungen ist das Risiko, dass Daten von Unbefugten aufgerufen werden, besonders hoch, da die Daten hier direkt auf den Geräten des Herstellers liegen und dieser häufig dauerhaften Zugriff hat, um die Cloud-Dienste zu warten. Hier wären besondere Absicherungen, wie z.B. eine effektive Verschlüsselung, erforderlich. Daher

können Bürger*innen den Cloud Lösungen nur schwer trauen, da auch sie nicht wissen können, was mit den Daten passiert.

Zu IT-Sicherheit zählt hier sowohl die Stabilität, die das Softwaresystem aufweist, als auch die Zuverlässigkeit, mit der sie bedient werden kann. Ist die Software nicht dauerhaft funktionsfähig oder gibt es Zuverlässigkeitsprobleme mit der Anwendungssoftware, die sich auf den Desktop-PCs befindet, so kann dies zu Fehlern oder zu einem Ausfall der Software führen.

Weiterhin umfasst IT-Sicherheit die Integrität der Daten. Die Daten müssen vor einer Manipulation geschützt und der Zugriff von Unbefugten darf nicht möglich sein. Um einer Manipulation vorzubeugen, können IT-Sicherheitsaudits vorgenommen werden. Bei einem Sicherheitsaudit wird das System von Expert*innen angegriffen und geprüft. Sicherheitslücken werden frühzeitig entdeckt und Mängel können behoben werden. IT-Sicherheitsaudits regelmäßig durchzuführen, ist eine sinnvolle Maßnahme, um Freie Software sicherer zu machen. IT-Sicherheit befasst sich mit grundlegenden Gefahren, die mit der Digitalisierung einhergehen, der Datenschutz hingegen nur mit der Sicherheit der Datenverarbeitung von Bürger*innen.

2.3.2 Interoperabilität mit bestehender Software

Bei der Integration von Freier Software in bestehende Systeme ist die Interoperabilität ein wichtiger Erfolgsfaktor. Die Interoperabilität beschreibt das Kommunizieren verschiedener Softwareprodukte mittels Schnittstellen. Diese Schnittstellen sind eine gemeinsame Sprache, die beide Produkte sprechen. So können Informationen austauschen werden. Die Freie Software muss reibungslos mit dem restlichen System zusammenarbeiten können, das eventuell auf proprietärer Software aufbaut und bereits Schnittstellen verwendet, die geschlossene Standards verwenden. Beim Einsatz Freier Software muss zunächst geprüft werden, ob die Schnittstellen miteinander kommunizieren können. Die Benutzerfreundlichkeit darf hierbei nicht eingeschränkt werden.

So muss die Software zum Beispiel mit aktuellen Benutzerverwaltungen kompatibel sein, um keine separate Anmeldung und Registrierung notwendig zu machen. Diese müsste parallel zu bestehenden Systemen verwendet werden, was einen Mehraufwand für Nutzer*innen bedeutet. Schließlich muss die Freie Software auf bestehender Serversoftware und Hardware funktionieren, damit keine indirekte Abhängigkeit von proprietärer Software entsteht.

Sind Teile der Freien Software proprietär ist der Einsatz dieser Software genauer zu betrachten, da eine Weiterverbreitung trotz freier Lizenzen eventuell nicht möglich ist. Um dem entgegenzuwirken, könnten Freie Softwareprodukte von unabhängigen

Organisationen bevorzugt werden, da diese weniger wirtschaftliche Interessen haben als Unternehmen. Oftmals werden Freie Softwareprodukte von Organisationen entwickelt, die Entwickler*innen Ressourcen zuweisen, Werbung verbreiten und Spendenzahlungen koordinieren und an Stellen verteilen, die Finanzierung benötigen.

Behörden können solche Organisationen in geeigneter Weise unterstützen und bei einigen Organisationen auch einzelne geplante Funktionen explizit unterstützen. Benötigt eine Verwaltung also eine Funktion, um die Software einsetzen zu können, beispielsweise eine Möglichkeit, mit bestehenden Anmeldeverfahren zu kommunizieren, kann diese Funktion aktiv gefördert werden und eine Implementation wird eingeleitet.

2.3.3 Hilfe bei Benutzung

Um den reibungslosen Einsatz der freien Software in der Verwaltung sicherzustellen, müssen Nutzer*innen der Software vor Inbetriebnahme eingehend geschult und weitergebildet werden, damit der Wechsel ohne Ausfälle stattfinden kann. Wird die Software ohne weitere Informationen eingeführt, wird sie keine Akzeptanz finden. Vor der Einführung Freier Software sollten zudem Vorzüge und Entscheidungsgründe dem Nutzerkreis mitgeteilt werden. In der Anfangsphase, nach der Inbetriebnahme der Software, sollte außerdem technischer Support zur Verfügung stehen, der bei spezifischen Fragen aushelfen kann.

Hierzu sind besonders geschulte Administrator*innen und Entwickler*innen notwendig, die Probleme mit der Freien Software schnell und effizient beheben können. Außerdem sollten Nutzer*innen gehört und Kritik und Verbesserungsvorschläge berücksichtigt werden. Gegebenenfalls muss die Software durch einen eigenen Fork angepasst werden, bis die Nutzung reibungslos möglich ist. Hierbei ist zu beachten, dass auch nach der ersten Anpassung Kritikpunkte wahrgenommen werden.

Softwareentwicklung ist ein kontinuierlicher Prozess, der auf Verbesserung und Veränderung angewiesen ist. Nutzer*innen sollten daher zusätzlich regelmäßig befragt werden, wie sie zu der Freien Software stehen und ob sie konkrete Probleme erkennen. Schließlich ist zu beachten, dass eine Umstellung zu Freier Software vorrangig als positiv eingestuft wird, wenn diese mit Arbeitserleichterungen einhergeht. Macht eine Freie Software einfache Arbeitsabläufe komplexer und länger, so sehen Nutzer*innen keinen Zweck in der neuen Software und verwenden diese nur ungern. Neue Software sollte immer ein Fortschritt, nie ein Rückschritt sein. Freie Software nur einzuführen, weil sie frei ist, dabei jedoch auf weitere Faktoren, wie Nutzbarkeit, nicht weiter einzugehen, ist problematisch und sollte vermieden werden.

2.3.4 Qualität

Auch die Qualität einer Freien Software entscheidet über den Erfolg. Vorab sollte demnach eine eingehende Bewertung der Software durchgeführt werden, bei der verschiedene Produkte miteinander verglichen werden. Ist die Nutzung einer Freien Software absolut undenkbar, so sollte eine proprietäre Lösung dennoch in Betracht gezogen werden, sofern diese keine weiteren Einschränkungen mitbringt, um Digitalisierungsanstrengungen voranzutreiben.

Um eine Freie Software zu bewerten, sollte darauf geachtet werden, ob eine Freie Software über viele, aktive Entwickler*innen oder eine Hersteller-Organisation verfügt, die die Software warten, aktualisieren und Fehler beheben. Andernfalls kann es nach der Inbetriebnahme der Software zu Sicherheitslücken kommen, die nicht behoben werden. Auch werden mit einer zu kleinen Community neue Funktionen nur schleichend eingeführt. In diesem Fall ist ein eigener Fork, der von der Verwaltung betreut wird, in Betracht zu ziehen.

Hinter einigen Projekten steht auch ein Herstellerunternehmen, das die Entwicklung der Software beaufsichtigt und finanzielle Absichten verfolgt. Einige Softwareprodukte sind so zum Beispiel als Freie Software und als erweiterte proprietäre Software gleichzeitig erhältlich. Das Freie Softwareprodukt hat in diesem Fall häufig weniger Funktionen. Trotzdem sollte die Freie Software der proprietären vorgezogen werden, wenn die erweiterten Funktionsumfänge nicht zwingend benötigt werden. Ein weiteres Indiz für eine gute Freie Software kann der Zeitraum sein, über den das Projekt bereits existiert. Wenn ein Projekt erst wenige Tage oder Wochen alt ist, ist davon auszugehen, dass das Projekt unvollständig und fehleranfällig ist. Vermutlich befindet sich das Projekt weiterhin in der Entwicklung und benötigt noch Zeit, um alle Funktionen zu erhalten.

Eine Mitarbeit an einer solchen, unvollständigen Software kann die Entwicklungszeiten verkürzen und ein Produkt nach eigenen Vorstellungen garantieren. Hier können auch Erfahrungsberichte anderer Verwaltungen aufschlussreich sein, wenn diese die Software bereits benutzen. Solche Erfahrungsberichte können auch unter freien Lizenzen mit der Öffentlichkeit geteilt werden. Das Projekt „Ein Ort für öffentlichen Code“ kann in diesem Kontext auch als Plattform zum Austausch von Erfahrungsberichten eingesetzt werden.

Berichte zu Freier Software sollten frei verfügbar sein, um eine weitere, unnötige Bewertung von einer weiteren Verwaltung zu vermeiden. Schließlich sollte vor dem Einsatz die Qualität des Quelltextes berücksichtigt werden, da komplexe, unleserliche Quelltexte zukünftig unwahrscheinlich weiterentwickelt werden. Auch können eigene Anpassungen nur schwer stattfinden, da sich eigene Entwickler*innen nicht in dem

Quelltext zurechtfinden und demnach Schwierigkeiten haben diesen zu bearbeiten.

2.3.5 Zusammenfassung

Vor dem Einsatz von Freier Software muss bewertet werden, ob diese den **rechtlichen Ansprüchen**, u. a. der **IT-Sicherheit** sowie dem **Datenschutz** genügt. Sollten daran Zweifel bestehen, müssen entsprechende Veränderungen vorgenommen oder Alternativen gesucht werden. Auch die Integrität der Daten muss gewährleistet werden. Bei Freier Software kann erkannt werden, ob Manipulationen oder Sicherheitslücken vorhanden sind, die dann geschlossen werden müssen. Bei proprietärer Software wäre die Überprüfung nicht einfach möglich.

Eine Bewertung von Freier Software ist allerdings nicht ausreichend, insbesondere muss die **Interoperabilität mit bestehender Software** gegeben sein. Freie Software muss also Schnittstellen zu Fachverfahren, proprietärer Software und dem restlichen System bieten. Auch muss sie mit aktuellen Verwaltungslösungen (z.B. Benutzerverwaltung) kompatibel sein, damit z.B. Single-Sign-On ermöglicht wird, was wiederum Einfluss auf die Benutzerfreundlichkeit hat. Um dies zu gewährleisten, können Teile der Infrastruktur (Server und Betriebssysteme) ebenfalls auf Freier Software basieren, sodass hier bereits Hürden entfallen.

Auch hängt der Erfolg von Freier Software maßgeblich von den Benutzer*innen ab. Diese benötigen insbesondere in der Einführungsphase **Hilfe bei der Benutzung** und müssen auf die Vorteile von Freier Software hingewiesen werden. Insbesondere wenn diese Kritik und Verbesserungsvorschläge liefern, sollte dies Berücksichtigung in der Entwicklung oder bei der Anpassung in der Einführungsphase finden. Mit der Einführung von Software sollte immer ein Fortschritt erzielt werden, sodass Faktoren, wie Nutzbarkeit und **Qualität**, ebenso maßgebend sind, wie die Freiheiten, welche die Software bietet.

3 BEWERTUNG DER LIZENZMODELLE FREIER SOFTWARE

Freie Software unterliegt bestimmten Lizenzbedingungen, die ihr die folgenden Rechte bzw. Freiheiten gewähren:

1. die Freiheit, das Programm für jeden Zweck auszuführen,
2. die Freiheit, die Funktionsweise eines Programms zu untersuchen und es an eigene Bedürfnisse anzupassen,
3. die Freiheit, Kopien weiterzugeben und
4. die Freiheit, ein Programm zu verbessern und die Verbesserungen an die Öffentlichkeit weiterzugeben.

[\(https://fsfe.org/freesoftware/\)](https://fsfe.org/freesoftware/)

3.1 Bewertung

Dieser Abschnitt wird eine Bewertung der Lizenzmodelle Freier Software für die Stadtverwaltung Dortmund durchführen und dabei einen Bezug zu den Anwendungsbereichen herstellen, in denen Freie Software besonders gut eingesetzt werden kann.

Die Einsatzmöglichkeiten Freier Software sind heutzutage praktisch endlos. In nahezu jedem Anwendungsbereich kann Freie Software zum Einsatz kommen, um die Anforderungen für dieses Anwendungsgebiet zu unterstützen.

3.1.1 Allgemein

Grundsätzlich gilt, dass Urheber Freier Software die Lizenz für den erstellten Code frei wählen können. Sie können sich einer bereits existierenden Lizenz (z.B. der General Public Licence – GPL) anschließen oder für ihre Anwendung eine eigene Abwandlung dieser Lizenzen schaffen. Nach deutschem Recht sind diese Lizenzen zwar nicht immer mit den Prinzipien des Urheberrechts und den zugehörigen restriktiven Vorgaben für die klare Definition von Nutzungs- und Verwertungsrechten vereinbar, es wird jedoch anerkannt, dass durch den Download Freier Software die Lizenzbedingungen des Urhebers bereits rechtswirksam akzeptiert werden. Eine Nichtbeachtung dieser Bedingungen kann dazu führen, dass die gewährten Nutzungs- und Verarbeitungsrichtlinien erlöschen (vgl. Rechtliche Fallstricke des Einsatzes von Open Source Software und Freier Software – Hinweise für die Praxis; Thomas Wilmer; 2021).

3.1.2 Marktstrukturen

Freie Software wird durch die unterschiedlichsten Projekte erstellt und verbreitet. Je nach Nutzerzahl kann es vorkommen, dass Produkte unregelmäßig neue Versionen hervorbringen oder nach einiger Zeit keine Betrachtung mehr finden, da ein Alternativprodukt höhere Vorteile bietet. Dies kann häufig ein Indiz dafür sein, dass diese Projekte nicht ausreichend unterstützt werden und somit der Service und die Fehlerbehebung leiden.

In diesem Zusammenhang lohnt es sich, die einzelnen Sparten von freien Softwareprojekten zu betrachten und eine Grobstruktur aufzuzeigen:

- **Freie Teams oder Einzelentwickler:**

Lange Zeit wurde Freie Software ausschließlich von freien Entwicklern produziert, die in der Freizeit die Software entwickeln und meist Hauptnutzer ihrer eigenen Anwendung sind.

- **Institutionelle, non-Profit-Projekte:**

Größere Projekte haben in der Regel eine institutionelle Basis. So wird zum Beispiel der Web-Browser Firefox von der Mozilla-Foundation bzw. der Tochterfirma Mozilla Corporation entwickelt. Diese Institutionen und Projekte werden meist durch Spenden oder den Verkauf von Merchandising-Produkten finanziert. Ein Vorteil dieser institutionellen Entwicklung ist, dass in dem Projekt professionell entwickelt wird und dass die Ideale einer freien Softwareentwicklung gelten.

- **Kommerzielles Basisprodukt:**

Da es in der heutigen Zeit schwierig ist, neue Softwareprodukte am Markt zu platzieren, gehen viele Unternehmen den Weg, ihre Software als Freie Software zur Verfügung zu stellen und verlangen keine Lizenzgebühren, da sie sich Service- und Wartungsvereinbarungen erhoffen.

(vgl. Positionspapier Freie Software/Open Source; Dr. Michael Neubauer; Vitako)

Auch kommt darunter das Modell des Dual Licensing, also die zusätzliche Freigabe des Programmcodes unter einer Freien Lizenz, neben der Veröffentlichung unter einer Kauflizenz zum Einsatz. Hierbei entstehen zwei unterschiedliche Lizenzversionen der gleichen Software. Schwierig ist hierbei für den Hersteller, dass die Wiedereingliederung von Weiterentwicklungen aus dem OSS-Bereich nicht erfolgen darf (vgl. Rechtliche Fallstricke des Einsatzes

von Open Source Software und Freier Software – Hinweise für die Praxis; Thomas Wilmer; 2021).

Zusammengefasst lässt sich festhalten, dass die Entwicklung einer Freien Software einen individuellen Zweck verfolgt und somit aus mehreren Richtungen initiiert werden kann. Auch kann sich ein Projekt, das durch eine Entwicklergemeinschaft initiiert wurde, im Verlaufe der Weiterentwicklung durch die Community zu einem Projekt entwickeln, das unter einer Stiftung oder einem Verein formiert wird um die Entwicklungsressourcen zu bündeln. Auch können Projekte durch Firmen oder Kommunen initiiert werden und somit die Basis für ein Freies-Software-Projekt bilden.

3.1.3 Anwendungsbereiche

Freie Software kann gemäß organisatorischen Bedürfnissen in folgenden Kategorien vorkommen:

- **Infrastruktur**
- **Basis-Dienste**
- **Anwendungen**
- **Entwicklungswerkzeuge**

Insbesondere in der letztgenannten Kategorie, den Entwicklungswerkzeugen, besteht eine große Anzahl an Tools, die sich speziell an Softwareentwickler*innen wenden. Dies ist darauf zurückzuführen, dass Entwicklungstools insbesondere für proprietäre Software mit hohen Kosten einhergehen und die Freien Softwareentwickler sich damit selbst Werkzeuge schaffen mussten, um Ihre Software zu entwickeln.

Den Kern Freier Software bilden jedoch die Dienste. Es wurden Dienste, wie NFS oder BIND, entwickelt und als Open-Source-Programmpakete zur Verfügung gestellt, die heute einen wesentlichen Teil der IT-Infrastruktur vereinfachen. Daher ist es auch nicht verwunderlich, dass wesentliche Funktionen in Rechenzentren auf Basis von Freien Softwarekomponenten betrieben werden und somit heute schon einen wichtigen Bestandteil der IT-Infrastruktur ausmachen (vgl. Positionspapier Freie Software/Open Source; Dr. Michael Neubauer; Vitako).

Problematischer fällt jedoch die Betrachtung der Kategorie Anwendungen aus. Zwar besteht ein großes Angebot von Freier Software, die bestimmte Lösungen bieten (z.B. als Browser das Produkt Firefox, im Bereich Office-Anwendungen das Produkt Open Office oder im Bereich der Zeichnungs-/Bildbearbeitung das Produkt GIMP), jedoch sind diese Anwendungen in der Regel Werkzeuge, um ein Problem zu bearbeiten. Je spezifischer daher die Anwendungen werden, umso schmaler wird das Angebot von

Lösungen auf Basis von Freier Software. Auch gibt es heute kaum Anwendungen in Bereichen der kommunalen Kernaufgaben der Verwaltung, die auf Basis von Freier Software zur Verfügung stehen. Da dieser Trend nicht nur auf die Branchen der Kommunalverwaltungen beschränkt ist, zeigen sie auch die Grenzen des Modells Freier Softwareentwicklung deutlich auf (vgl. Positionspapier Freie Software/Open Source; Dr. Michael Neubauer; Vitako).

3.2 Fazit

Die Bewertung der Lizenzmodelle Freier Software zeigt auf, dass der Einsatz von Freier Software in den Bereichen der Kommunen durchaus weitreichende Potenziale bietet.

In den Bereichen der Systemtechnik, Serverinfrastruktur, der Basisdienste und der Entwicklungsinfrastrukturen muss der heute bereits bestehende Trend weiter fortgeführt werden. Hier bietet Freie Software viele Schnittstellen und Standards, die, wenn sie eingehalten werden, eine langfristige Stabilität der Infrastruktur gewährleisten können.

Jedoch bergen der Einsatz und auch die Entwicklung von eigenen Lösungen auf Basis von Freier Software Gefahren, die nicht vernachlässigt werden dürfen. Im Bereich der Anwendungen muss es zunächst einmal Freie Software geben, die für den gewünschten Einsatzzweck passend ist. Bei der Ablösung und der Neueinführung von Software spielt auch die Akzeptanz der Mitarbeiter*innen eine wesentliche Rolle. Sollte eine Software ausgewählt werden, weil Freie Software als ein KO-Kriterium definiert wird, die Lösung aber nicht die weiteren Aspekte erfüllt, besteht das Risiko, dass eine Unzufriedenheit bei den Anwender*innen dazu führt, dass dieses Arbeitsmedium nicht akzeptiert wird.

Daher gilt es, im Rahmen der bestehenden Produktverantwortungen des Prozesses zur Einführung von Fachsoftware und bei der Eigenentwicklung von Software darauf zu achten, Freie Software zu berücksichtigen und bei der Betrachtung und Entwicklung einzubeziehen.

Dies kann unter anderem gelingen, indem die Zusammenarbeit mit Open-Source-Partnern, wie der Free Software Foundation Europe, der Open Source Business Alliance oder den Bestrebungen der KGSt zur Schaffung einer Open Code Plattform (einem Open Source Code Repository) für die Öffentliche Verwaltung, intensiviert und durch eine zentrale Stelle innerhalb der Stadtverwaltung Dortmund koordiniert wird.

4 BEWERTUNG VON DATEI-FORMATEN

Im Rahmen der Bestandsaufnahme zur „Potenzialanalyse Freie Software und Offene Standards“ wurden innerhalb der Stadtverwaltung Dortmund die geläufigsten Dateierendungen und -formate erfasst. Im Rahmen einer ersten Analyse wurden diese Endungen und Formate in Kategorien zusammengefasst.

4.1 Analyse von Datei-Formaten

Um eine Analyse von Datei-Formaten durchführen zu können, wurde zunächst definiert, welches Ziel mit dieser Analyse erreicht werden soll. Auf Basis der Bestandsaufnahme und der erhobenen Daten der Stadtverwaltung Dortmund wurde festgestellt, dass bereits ein Trend hin zu Offenen Formaten eingesetzt hat. Diesen Trend kann man forcieren und verstärken, indem die Vorteile von Offenen Formaten Berücksichtigung finden und herausgestellt werden.

Vorteile von Offenen Formaten sind,

- dass ihre Spezifikationen öffentlich publiziert und zugänglich sind
- dass es keine technischen und rechtlichen Einschränkungen für die Nutzung gibt
- dass diese unter der Aufsicht von offen zugänglichen Gremien entwickelt werden
- dass diese durch jegliche Software (proprietär oder frei) implementiert werden kann.

Auf Basis dieser vier Vorteile wird im Folgenden nun die Festlegung von Standard-Datei-Formaten für die Stadtverwaltung Dortmund herausgestellt.

4.2 Festlegung von Standard-Datei-Formaten

Für die weitere Festlegung der erfassten Datei-Formate der Stadtverwaltung Dortmund wurden die TOP-3-Kriterien bewertet und in diesem Standard zusammengefasst. Damit dieser eine wesentliche Basis für die Stadtverwaltung Dortmund bilden kann, wurden in der Erstellung auch Dateiformate aufgenommen, die den Kategorien Multimediadaten (Audio- und Videodaten) sowie Datenkompression zugehören.

Innerhalb der Stadtverwaltung Dortmund werden Datei-Formate verwendet, die aus den verschiedensten Anwendungen und Verfahren erstellt werden. Die Vereinheitlichung und somit die Festlegung eines Standard-Formates für den

jeweiligen Verwendungszweck hängt dabei von verschiedensten Faktoren ab. Mit dieser Standardisierung wird eine Empfehlung zum Einsatz von Datei-Formaten vorgenommen, die im Rahmen der weiteren Digitalisierungsprojekte, Ablösungsprojekte und Entwicklungen der Stadtverwaltung Dortmund zu beachten ist.

4.2.1 Formate für Office-Dokumente

Office-Dokumente sind Textdokumente, Tabellenkalkulationen und Präsentationen.

4.2.1.1 Textdokumente zum Informationsaustausch

Textdokumente, die nur dem Informationsaustausch dienen, werden in der Regel nicht weiterverarbeitet oder verändert. Dem Empfängerkreis sollten daher Dokumente in einer Form zur Verfügung gestellt werden, die eine Veränderung nicht ermöglichen.

Klassifizierung:	Format:
Empfehlung	Portable Document Format (PDF) Version größer 1.4
weitere Informationen:	https://www.pdfa.org/

Es wird empfohlen, PDF - Portable Document Format ab der Version 1.4 für diese Textdokumente zu verwenden. PDF ist ein Format zum Austausch und zur Ansicht elektronischer Dokumente. PDF wurde ursprünglich von Adobe Systems Inc. entwickelt. Seit der Version 1.7 ist PDF in der ISO unter ISO 32000 normiert.

Die Empfehlung gilt zur Verwendung der Version 1.4, als Profil davon, kann auch PDF/A verwendet werden. Erst mit dieser Version gibt es die Möglichkeit, Dokumente zu erzeugen, die barrierefrei sind. Höhere Versionen dieses Standards können in Abhängigkeit von zusätzlichen Funktionen, die benötigt werden, verwendet werden. Es ist insbesondere auf Kompatibilitäten bei der Verwendung von mobilen Geräten zu achten.

Klassifizierung:	Format:
Empfehlung	Plain Text (TXT)

Sollte es zum Austausch von Textdokumenten mit einfachem, unstrukturierten Text ohne Formatierungen oder Layout kommen, wird empfohlen, diesen als

„plain/text“ (meist als Dateityp .txt“) bereitzustellen. Wichtig sind dabei die Zeichensätze und die Zeichenkodierung zu beachten. Hierbei wird empfohlen, das UCS-Transformation-Format (UTF) in der Version 8 (UTF-8) zu verwenden.

4.2.1.2 **Textdokumente zur Weiterverarbeitung**

Textdokumente, die für eine weitere Bearbeitung durch verschiedenste Stellen oder Schnittstellen erstellt werden, müssen durch die verschiedensten Office-Programme verarbeitet werden können. Es ist dabei darauf zu achten, welcher Typ von Textdokumenten hier vorliegt. Komplexe Textdokumente enthalten Layout-Informationen und Formatierungen, die bei der Weiterverarbeitung elementar sein können. Einfache Textdokumente enthalten diese Zusatzinformationen nicht.

Klassifizierung:	Format:
Empfehlung	Open Document Format for Office Applications (Open Document) Version 1.2
weitere Informationen:	https://www.oasis-open.org/

Das „Open Document Format for Office Applications (OpenDocument)“, kurz ODF ist ein auf XML basierender, international genormter und quelloffener Standard für Dateiformate von Büroanwendungen. Dies können Textdokumente, Tabellendokumente, Präsentationen, Zeichnungen, Bilder oder Diagramme sein.

OpenDocument wird durch OASIS entwickelt und ist durch die ISO in der ISO 26300 genormt.

Da es sich bei ODF um einen quelloffenen Standard handelt, können Dokumente in diesem Format einfach weiterbearbeitet werden, da zusätzliche Informationen zum Layout oder zur Formatierung kompatibel gespeichert werden und somit von jeder Office-Anwendung verarbeitet werden können.

Klassifizierung:	Format:
Empfehlung	Office Open XML (OOXML) Transitional/Strict
weitere Informationen:	https://www.ecma-international.org/publications-and-standards/standards/ecma-376/

Sofern das quelloffene Format (ODF) nicht genutzt werden kann, da das vor- oder nachgelagerte System nicht verarbeitet wird, wird auf das Format Office Open XML zurückgegriffen.

OOXML ist ein ursprünglich von Microsoft entwickeltes auf XML-basierendes Format zur Speicherung von Dokumenten aus Büroanwendungen. Es wird in der ISO als ISO 29500 normiert.

Hierbei unterscheidet sich OOXML in zwei Versionen, der Transitional- und der Strict-Variante. Bei der Transitional-Variante liegt der Fokus auf der Kompatibilität zu dem älteren Microsoft Office Files der Microsoft Office Suite. Bei der Strict-Variante wurde der Transitional-Anteil ausgeschlossen, da dieser durch herstellerunabhängige Office-Anbieter nur schwer zu implementieren ist.

Es wird empfohlen, sollte kein quelloffenes Format genutzt werden, Dokumente im Office Open XML (OOXML) Strict zu erstellen und zur Weiterverarbeitung zu verwenden.

Klassifizierung:	Format:
Empfehlung	Rich Text Format (RTF)

Sofern das quelloffene Format (ODF) nicht genutzt werden kann, da das vor- oder nachgelagerte System nicht verarbeitet wird, kann auf das Format Rich Text Format (RTF) zurückgegriffen werden.

Bei einem Austausch mit diesem Format kann nicht gewährleistet werden, dass das Layout oder die Formatierungen im nachverarbeitenden Programm bestehen bleibt.

Klassifizierung:	Format:
Ablaufend	Microsoft Office File Formats
weitere Informationen:	https://docs.microsoft.com/en-us/openspecs/office_file_formats/ms-doc/ccd7b486-7881-484c-a137-51170af7cc22

Sofern das quelloffene Format (ODF) nicht genutzt werden kann, da das vor- oder nachgelagerte System nicht verarbeitet wird, wurde bislang auch auf das Format Microsoft Office File Formats zurückgegriffen.

Das Format wird als ablaufend in der Liste der eingesetzten Formate geführt, da es nur in der Vergangenheit eine hohe Relevanz hatte. Das Word Binary File

Format (DOC) kann für Textdokumente, die zur Weiterbearbeitung vorgesehen sind, verwendet werden.

4.2.1.3 **Tabellenkalkulationen zum Informationsaustausch**

Tabellendokumente, die nur dem Informationsaustausch dienen, werden in der Regel nicht weiterverarbeitet oder verändert. Dem Empfängerkreis sollten daher Dokumente in einer Form zur Verfügung gestellt werden, die eine Veränderung nicht ermöglichen.

Klassifizierung:	Format:
Empfehlung	Portable Document Format (PDF) Version größer 1.4
weitere Informationen:	https://www.pdfa.org/

Siehe Empfehlung zu Portable Documents Format (PDF) Version größer 1.4 im Abschnitt 4.1.1.1 „Textdokumente zum Informationsaustausch“.

Klassifizierung:	Format:
Empfehlung	Comma-Separated Values (CSV)
weitere Informationen:	https://datatracker.ietf.org/doc/html/rfc4180

Einfach strukturierte Tabellenkalkulationen ohne Berechnungen und Layout sollten im Format Comma-Separated Values (CSV) ausgetauscht werden. Das Format ist durch den RFC 4180 dokumentiert, jedoch nicht standardisiert.

4.2.1.4 **Tabellenkalkulationen zur Weiterverarbeitung**

Tabellenkalkulationen, die für eine weitere Bearbeitung durch verschiedenste Stellen oder Schnittstellen erstellt werden, müssen durch die verschiedensten Office-Programme verarbeitet werden können. Es ist dabei darauf zu achten, welcher Typ von Tabellenkalkulation hier vorliegt. Komplexe Tabellen enthalten Layout-Informationen, Berechnungen und Formatierungen, die bei der Weiterverarbeitung elementar sein können. Einfache Tabellen enthalten diese Zusatzinformationen nicht.

Klassifizierung:	Format:
Empfehlung	Open Document Format for Office Applications (Open Document) Version 1.2

weitere Informationen:	https://www.oasis-open.org/
------------------------	---

Siehe Empfehlung zu Open Document Format for Office Applications (Open Document) Version 1.2 im Abschnitt 4.1.1.2 „Textdokumente zur Weiterverarbeitung“.

Klassifizierung:	Format:
Empfehlung	Office Open XML (OOXML) Transitional/Strict
weitere Informationen:	https://www.ecma-international.org/publications-and-standards/standards/ecma-376/

Siehe Empfehlung zu Office Open XML (OOXML) Transitional/Strict im Abschnitt 4.1.1.2 „Textdokumente zur Weiterverarbeitung“.

Klassifizierung:	Format:
Empfehlung	Comma-Separated Values (CSV)
weitere Informationen:	https://datatracker.ietf.org/doc/html/rfc4180

Siehe Empfehlung zu Comma-Separated Values (CSV) im Abschnitt 4.1.1.3 „Tabellenkalkulationen zum Informationsaustausch“.

Klassifizierung:	Format:
Ablaufend	Microsoft Office File Formats
weitere Informationen:	https://docs.microsoft.com/en-us/openspecs/office_file_formats/ms-doc/ccd7b486-7881-484c-a137-51170af7cc22

Siehe Microsoft Office File Formats im Abschnitt 4.1.1.2 „Textdokumente zur Weiterverarbeitung“.

4.2.1.5 **Präsentationen zum Informationsaustausch**

Präsentationen, die nur dem Informationsaustausch dienen, werden in der Regel nicht weiterverarbeitet oder verändert. Dem Empfängerkreis sollten daher Dokumente in einer Form zur Verfügung gestellt werden, die eine Veränderung nicht ermöglichen.

Klassifizierung:	Format:
Empfehlung	Portable Document Format (PDF) Version größer 1.4
weitere Informationen:	https://www.pdffa.org/

Siehe Empfehlung zu Portable Document Format (PDF) Version größer 1.4 im Abschnitt 4.1.1.1 „Textdokumente zum Informationsaustausch“.

4.2.1.6 Präsentationen zur Weiterverarbeitung

Präsentationen, die für eine weitere Bearbeitung durch verschiedenste Stellen oder Schnittstellen erstellt werden, müssen durch die verschiedensten Office-Programme verarbeitet werden können.

Klassifizierung:	Format:
Empfehlung	Open Document Format for Office Applications (Open Document) Version 1.2
weitere Informationen:	https://www.oasis-open.org/

Siehe Empfehlung zu Open Document Format for Office Applications (Open Document) Version 1.2 im Abschnitt 4.1.1.2 „Textdokumente zur Weiterverarbeitung“.

Klassifizierung:	Format:
Empfehlung	Office Open XML (OOXML) Transitional/Strict
weitere Informationen:	https://www.ecma-international.org/publications-and-standards/standards/ecma-376/

Siehe Empfehlung zu Office Open XML (OOXML) Transitional/Strict im Abschnitt 4.1.1.2 „Textdokumente zur Weiterverarbeitung“.

Klassifizierung:	Format:
Ablaufend	Microsoft Office File Formats
weitere Informationen:	https://docs.microsoft.com/en-us/openspecs/office_file_formats/ms-doc/ccd7b486-7881-484c-a137-51170af7cc22

Siehe Microsoft Office File Formats im Abschnitt 4.1.1.2 „Textdokumente zur Weiterverarbeitung“.

4.2.2 Formate für Grafiken

Bei Grafik-Dokumenten handelt es sich um Fotos, Scans, Rastergrafiken oder um sonstige grafische Dokumente, die je nach Anwendungsgebiet in unterschiedlicher Komprimierung und Qualität verwendet werden.

Klassifizierung:	Format:
Empfehlung	Tagged Image File Format (TIFF oder TIF)
weitere Informationen:	https://tools.ietf.org/html/rfc3302

Das Tagged Image File Format (TIFF) speichert gerasterte Bilder und sollte insbesondere verwendet werden, wenn mehrseitige Grafiken oder Bilder verarbeitet werden. Dies ist insbesondere beim Scannen mehrseitiger Dokumente empfohlen.

Klassifizierung:	Format:
Empfehlung	Portable Network Graphics (PNG)
weitere Informationen:	https://tools.ietf.org/html/rfc2083

Das Portable Network Graphics (PNG) ist ein Grafikformat, das eine verlustfreie Komprimierung ermöglicht. Darüber hinaus unterstützt das Format transparente/halbtransparente Bilder. Das Format sollte verwendet werden, wenn Grafiken und Schaubilder verarbeitet werden.

Klassifizierung:	Format:
Empfehlung	WebP
weitere Informationen:	https://developers.google.com/speed/webp

Das Format WebP ist ein aktuelles und modernes Format zur Speicherung von Bilddateien im Internet. Es wurde entwickelt, um als neuer offener Standard für verlustbehaftete komprimierte Grafiken im Web zu dienen. Es bietet im Vergleich zu JPEG eine höhere Qualität bei kleinerer Datenmenge.

Klassifizierung:	Format:
Empfehlung	Joint Photographic Experts Group (JPEG)
weitere Informationen:	https://jpeg.org/index.html?langsel=de

Das Format Joint Photographic Experts Group (JPEG) ist ein Grafikformat zur Speicherung von Bildern, die nach der JPEG-Norm komprimiert wurden. Das Format wird insbesondere im Bereich von Fotos genutzt.

Klassifizierung:	Format:
Empfehlung	Graphics Interchange Format (GIF)
weitere Informationen:	https://www.w3.org/Graphics/GIF/spec-gif89a.txt

Das Graphics Interchange Format (GIF) ist ein Dateiformat zur Speicherung von Bildern. Das Format erlaubt auch die Speicherung mehrerer (übereinanderliegender) Einzelbilder, die als Animationen interpretiert werden. Das Format sollte verwendet werden, wenn Animationen oder Diagramme dargestellt werden.

4.2.3 Formate für Multimediadaten

Im Bereich der Multimediadaten kommen insbesondere Containerformate für Audio- und Videodaten zum Einsatz. Daher sollten mindestens folgende Containerformate im Bereich von Multimedia-Anwendungen zum Einsatz kommen.

Klassifizierung:	Format:
Empfehlung	Ogg Encapsulation Format (Ogg)
weitere Informationen:	https://datatracker.ietf.org/doc/html/rfc3533

Das Ogg Encapsulation Format (Ogg) ist ein Container-Dateiformat für Multimedia-Dateien, kann also gleichzeitig Audio-, Video- sowie Textdaten enthalten. Es wurde mit dem Ziel konzipiert, eine freie und von Softwarepatenten unabhängige Alternative zu proprietären Formaten zu bieten, um Multimedia-Inhalte effizient zu speichern und zu streamen.

Bei Audiodateien mit niedrigen Qualitätsanforderungen, zum Beispiel Sprachaufzeichnungen, ist das Codec Speex geeignet. Für Audiodateien mit normalen Qualitätsanforderungen bietet sich Vorbis an, dessen Qualität mit

MP3 vergleichbar ist. Für höchste Qualitätsansprüche kann das verlustfreie Audio-Codec FLAC verwendet werden. Für Videodateien sollte Ogg in Verbindung mit dem freien Codec Theora verwendet werden.

Klassifizierung:	Format:
Empfehlung	MPEG-4 Part 14 (MP4)
weitere Informationen:	https://www.iso.org/standard/68960.html

MP4 ist ein Video-Containerformat, das von der Moving Picture Experts Group (MPEG) für MPEG-4-Inhalte vorgesehen und in ISO 14496-12 bzw. -14 (MPEG-4 Teil 12 und 14) standardisiert wurde.

4.2.4 Formate für Datenkomprimierung

Klassifizierung:	Format:
Empfehlung	7z
weitere Informationen:	https://www.7-zip.org/7z.html

Das Format 7z ist ein offenes, modulares Archivformat mit einer hohen Komprimierungsrate. Es ist im 7-Zip-Programm implementiert, das unter den Bedingungen der GNU Lesser General Public Licence verfügbar ist. Die Nutzung des 7z-Formates wird empfohlen, um einen hohen Komprimierungsgrad bei der Dokumentenweitergabe zu erzielen.

Klassifizierung:	Format:
Empfehlung	Zipped File (ZIP)
weitere Informationen:	https://www.iana.org/assignments/media-types/application/zip

Das Format Zipped File (ZIP) ist ein Format, um Dateien verlustfrei in einem Container zu komprimieren und/oder zusammenzufassen. ZIP ist in der ISO unter ISO 21320 standardisiert.

5 RÜCKSCHLÜSSE FÜR DIE STADT DORTMUND

Die Stadtverwaltung Dortmund setzt bereits erfolgreich Lösungen auf Basis von Open-Source-Software ein. Durch die Potenzialanalyse Freie Software und Offene Standards, lassen sich darüber hinaus Handlungsempfehlungen und Rückschlüsse herleiten, die einen wesentlichen Einfluss auf die Digitalisierung und die weitere Entwicklung im Bereich der IT-Infrastruktur nehmen.

Das Memorandum zur Digitalisierung 2020 bis 2025 gibt hierzu bereits einige Handlungsempfehlungen vor:

„Wo möglich Nutzung von Open Source Software. Von der Verwaltung entwickelte oder zu Entwicklung beauftragte Software wird der Allgemeinheit zur Verfügung gestellt.“

Um dieser Aufgabe gerecht zu werden, gilt es, neben der Fortführung bestehender Maßnahmen, weitere Maßnahmen abzuleiten.

Bei der Einführung von Software und dem damit verbundenen Softwareeinführungsprozess werden fachliche Anforderungen, Sicherheit, Wirtschaftlichkeit, Zuverlässigkeit und Schnittstellen betrachtet. Freie Software ist bei der Betrachtung dieser Kriterien proprietärer Software gleichgestellt. In jedem Einzelfall ist bei der Gesamtbetrachtung zu prüfen, welche Anforderungen am besten für die Stadtverwaltung Dortmund in Betracht kommen.

Die Entwicklungen bezogen auf die Digitale Souveränität und die damit verbundene Nutzung von Open-Source-Software sind weiterhin einzubeziehen und zu beobachten. Hierzu ist es erforderlich, einen konkreten Ansprechpartner für Fragen zu Open Source Software und offene Standards innerhalb der Stadtverwaltung Dortmund und somit eine zentrale Stelle für die Belange der Digitalen Souveränität zu etablieren. Diese bildet einerseits eine Schnittstelle zwischen Organisationen und Projekten im Bereich Open Source und dient gleichzeitig als Ansprechpartner innerhalb der Stadtverwaltung Dortmund, um diese Entwicklungen in Strategien und Konzepten einzubringen.

Insbesondere bei der Betrachtung von Fachanwendungen im Bereich der Kommunalverwaltungen ist der Einsatz von Open-Source-Software eine große Herausforderung. Dies liegt insbesondere an der mangelnden Verfügbarkeit passgenauer und mit existierenden Fachanwendungen interoperablen Lösungen. Öffentliche Verwaltungen sind daher diesem Mangel unterlegen und es bieten sich oft keine Alternativen zu proprietärer Software. Durch die Koordinationsstelle Digitale Souveränität gilt, es in diesem Zusammenhang die Kommunikation mit anderen öffentlichen Stellen der Kommunen, Länder und des Bundes zu suchen, sodass

Synergien und gemeinsame Projekte entwickelt werden, um diesem Mangel entgegenzuwirken.

Bei der Eigenentwicklung von Software und Produkten für den Einsatz innerhalb der Stadtverwaltung Dortmund gilt es, wenn möglich, eine Kooperation mit anderen Kommunen einzugehen. Auch sind die Möglichkeiten zu prüfen, ob die zu nutzende Software bereits in einer Basisvariante als Open-Source-Software zur Verfügung steht und die Weiterentwicklung dieser Software in einer eigenen Fork sinnvoll für die Stadtverwaltung ist. Diese Entwicklungen sind allerdings nur möglich, wenn bereits vorhandenes Personal entsprechende Expertise im Bereich von Open-Source-Entwicklungen aufbaut und dabei die aktuellen Betriebsaufgaben nicht vernachlässigt. Auch müssen nach Abschluss einer Eigenentwicklung und der Veröffentlichung an entsprechenden Stellen die dann gestellten Support- und Beratungserwartungen berücksichtigt werden. Wenn eine Kommune eine Entwicklung erbringt und diese in einem Open-Code-Repository veröffentlicht, muss davon ausgegangen werden, dass die Kommune für seine Entwicklung und auch für die Weiterentwicklung als verantwortlicher Ansprechpartner agiert. Dieser Effekt würde dazu führen, dass Entwickler*innen nur wenige Produkte verantworten und auch von außen gerichtete Anfragen beantworten müssen. Ein Weg, diesem entgegenzuwirken, wäre, dass die Stadtverwaltung Dortmund für Entwicklungen externe Dienstleister hinzuzieht, die Produkte auf Open Source anbieten, entwickeln und entsprechenden Support anbieten. Dadurch würde einerseits sichergestellt, dass ein einheitlicher Ansprechpartner für Entwicklungen, Support und Änderungsanforderungen besteht und darüber hinaus die Veröffentlichungen durch einen Dritten erfolgen, sodass die Stadtverwaltung nicht als Herausgeber in Erscheinung tritt und somit keinerlei Erwartungen an diese herangetragen werden, die sonst bereits stark eingebundenes Fachpersonal und deren Expertise erfordern.

Kritisch zu betrachten ist der Gedanke, dass durch den Einsatz von Open Source die Digitale Souveränität im Ganzen geschaffen wird. Dies kann nicht allein durch den Einsatz von Open-Source-Software und offenen Standards erzielt werden, jedoch kann die Nutzung von Open-Source-Software an geeigneter Stelle und von Offenen Standards ein wesentlicher Baustein sein, um das Ziel der Digitalen Souveränität zu erreichen.