

Die Organisation von Dateien in einer dreidimensionalen Umgebung

Leander C. Suda



MASTERARBEIT

eingereicht am
Fachhochschul-Masterstudiengang

Interactive Media

in Hagenberg

im Juni 2018

© Copyright 2018 Leander C. Suda

Diese Arbeit wird unter den Bedingungen der Creative Commons Lizenz *Attribution-NonCommercial-NoDerivatives 4.0 International* (CC BY-NC-ND 4.0) veröffentlicht – siehe <https://creativecommons.org/licenses/by-nc-nd/4.0/>.

Erklärung

Ich erkläre eidesstattlich, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen nicht benutzt und die den benutzten Quellen entnommenen Stellen als solche gekennzeichnet habe. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt.

Hagenberg, am 25. Juni 2018

Leander C. Suda

Gender Erklärung

Aus Gründen der besseren Lesbarkeit wird in dieser Diplomarbeit die Sprachform des generischen Maskulinums angewendet. Es wird an dieser Stelle darauf hingewiesen, dass die ausschließliche Verwendung der männlichen Form geschlechtsunabhängig verstanden werden soll.

Inhaltsverzeichnis

Erklärung	iii
Gender Erklärung	iv
Kurzfassung	viii
Abstract	ix
1 Einleitung	1
1.1 Motivation	1
1.2 Ziele	2
1.3 Aufbau	2
2 Related Work	4
2.1 Geschichte der 2D Dateimanager	4
2.2 Geschichte der 3D-Dateimanager	5
2.2.1 3D-Dateimanager	6
2.2.2 3D Desktops	8
2.3 Grundlagen des Dateimanagements	9
2.3.1 Filing und Piling	10
2.3.2 Filing und Piling am Computer	13
2.4 Persönliche Organisationstypen	14
2.5 Räumliches Gedächtnis und Suchstrategien	15
2.5.1 Klassifizierung von Dateien nach Verwendungsart	17
2.6 2D vs. 3D	18
3 Anforderungen und Konzepte	20
3.1 Faktoren	21
3.1.1 Dateimenge	22
3.1.2 Dateityp	23
3.1.3 Visuelles Feedback über Metainformationen	24
3.2 Realitätsnähe und Virtual Reality	25
3.2.1 Virtual Reality-Handschuhe	25
3.2.2 Bewegungssteuerung	26
3.2.3 Metaphern	26
3.3 Augmented und Mixed Reality	27

4	Designziele	29
4.1	Grundlagen der Applikation	29
4.2	Funktionen zur Dateimanipulation	30
4.2.1	Verschieben von Daten	30
4.2.2	Kopieren von Daten	30
4.2.3	Löschen von Daten	30
4.2.4	Weitere grundlegende Funktionen	31
4.2.5	Optionale Funktionen	31
5	Implementierung	33
5.1	Grundlagen	33
5.2	Umgebung	34
5.3	Statische Objekte	35
5.4	Dynamische Objekte	35
5.5	Instanziierung und Programmstart	36
5.5.1	Ordnerstruktur und -initiierung	37
5.6	Funktionen des Dateimanagers	37
5.6.1	Verschieben und Umbenennen	38
5.6.2	Kopieren	39
5.6.3	Löschen	39
6	Benutzerstudie	40
6.1	Aufgabe und Vorbereitung	40
6.2	Ablauf	41
6.3	Analyse der gewonnenen Daten	42
6.3.1	Demografische Daten	42
6.3.2	Auswertung des Fragebogens	42
6.3.3	Auswertung des praktischen Teils	43
6.3.4	Zeitmessungen	46
6.4	Zusammenfassung	46
6.5	Resultat	47
7	Diskussion	48
7.1	Benutzerstudie	48
7.1.1	Effizienz	49
7.1.2	Metaphern	49
7.1.3	Orientierung	50
8	Schlussbemerkungen	51
8.1	Aus der Sicht des Entwicklers	51
A	Inhalt der CD-ROM/DVD	53
A.1	Generell	53
A.2	Kopien der Onlinequellen	53
A.3	Bilder	53
	Quellenverzeichnis	57

Literatur	57
Audiovisuelle Medien	60
Software	60
Online-Quellen	60

Kurzfassung

Wenn man sich die Dateimanager der letzten 20 bis 30 Jahre ansieht, fällt auf, dass diese sich in ihren Funktionen kaum verändert haben, trotz des vorhandenen Potentials zur Verbesserung. Ein Dateimanager verhält sich wie ein reales Archivierungssystem, in dem Ordner nach Kategorien in Aktenschränke eingeordnet und über längere Zeit gelagert werden. Wenn man nun diese Metapher aus dem Dateisystem zurück in eine reale Arbeitssituation überträgt, wäre das so, als würden Dokumente sofort nach der Verwendung wieder in den Schrank eingeordnet werden. Dies entspricht allerdings nicht der natürlichen Arbeitsweise der meisten Personen; stattdessen werden relevante Dokumente in Stapeln auf oder in der Nähe des Arbeitsplatzes platziert. Dabei erzeugt bereits der Ort und die Reihenfolge der Stapel subtile Hinweise auf deren Inhalt und Wichtigkeit, welche in einem virtuellen Dateisystem in den meisten Fällen aber verloren geht.

Der Hauptfokus dieser Arbeit liegt auf der Verwendung von 3D in einem Dateimanager, aber auf eine andere Art, als die in den bisherigen verschiedenen Versuchen umgesetzte. Während die meisten dieser Versuche die dritte Dimension auf bestehenden Dateimanagern „aufgesetzt“ haben, sollte dieses Unterfangen das Augenmerk auf die vielseitigen neuen Möglichkeiten und Konzepte legen, die dadurch ermöglicht werden. Metaphern können etwa noch repräsentativer und zahlreicher eingesetzt werden. Das Dateisystem könnte in Bezug auf Dateimanagement als virtuelle Darstellung der Realität, ausgestattet mit einer Palette an praktischen Werkzeugen und Hilfsmitteln, dienen.

Da das Thema der Dateimanager und des -managements ein sehr umfassendes ist, beschränkt sich diese Thesis auf die Unterschiede des Organisationsverhaltens in 2D und 3D. Um diese vergleichen zu können, wurde ein Prototyp implementiert, in dem Dateien frei beweglich in einer 3D Umgebung bewegt und sortiert werden können. Eine Auswahl an Möbelstücken stehen dabei zum Sortieren zur Verfügung. Das Ziel dieses Prototypen war es, als Grundlage für eine Benutzerstudie zu dienen, in der die Teilnehmer aufgefordert wurden, Fotos in einen Schrank zu sortieren und diese später wieder zu finden. Ein begleitender Fragebogen sollte dabei das Organisationsverhalten in 2D Dateimanagern erschließen.

Abstract

The way our file managers work has not changed in the last 20 or 30 years, despite being not perfect and eligible for improvement. As the name already suggests, file managers work like a real life filing cabinet, where files are categorized and stored and later to be retrieved. According to this metaphor, a user would have to return a file to the cabinet immediately after he uses it, which would be highly unlikely in a real life situation. Instead, the already researched method of piling is used by the vast majority of people, where files and documents are stored in piles on various surfaces on or around the workplace. The placement and order of piles already gives subtle hints about the documents' content and importance, which is often lost in virtual file storage.

The main topic of this thesis is the addition of the third dimension to a file manager, but in a new, different way than in previous approaches. While most of those added 3D “on top” of the old file manager, there should be a much stronger focus und the new possibilities and concepts this step can unlock. Metaphors can be taken to the next level, and be even closer to reality. In a way, it can be implemented as a virtual depiction of reality, with a palette of handy gadgets and functions to ease the organization of files.

As the topic of file managers and file organization covers a wide range, this thesis concentrates on the aspect of how the organization of files in 2D compares to the organization of files in 3D. To compare these, a prototype was implemented that lets users sort files in a 3D space in a realistic manner. An array of furniture articles is provided to store files or place them onto. This prototype was then used to hold a user study, where participants where observed sorting, placing files in and retrieving files from a single virtual cupboard. An accompanying questionnaire was used to determine the participants' 2D organizational behaviour.

Kapitel 1

Einleitung

1.1 Motivation

Wenn man sich die Geschichte von Dateimanagern ansieht, fällt schnell auf, dass sich die grundlegende Funktionalität seit den 1980er Jahren nicht merklich verändert hat. Man könnte meinen, das läge daran, dass die Umsetzung der Dateimanager schon ausgereift sei; Tatsache ist aber, dass das Design und die Umsetzung auf die Technik der 80er abgestimmt wurde und die umsetzbaren Möglichkeiten durch diese stark beschränkt waren. Zusätzlich sind die Speichermethoden von Computersystemen an reale Archivsysteme, die mit Ordnern voller Dokumente gefüllt werden, angelehnt (ersichtlich etwa an den Ordnersymbolen). Es ist aber so, dass kaum jemand ausschließlich mit Archiven arbeitet, sondern lockerere Arbeitsmethoden bevorzugt werden. Es gibt Studien, die die Arbeitsweise in realen Büros [25, 32] und an virtuellen Arbeitsplätzen [3, 4, 26] untersuchen, und diese stellen einstimmig fest, dass das Arbeiten mit Archiven nicht den natürlichen und tatsächlichen Arbeitsvorgang widerspiegelt.

Im Lauf dieser Masterthesis wird detailliert auf die Alternativen zum sog. *Filing* eingegangen. Die wichtigste, und von fast allen Personen, die mit Dokumenten verschiedenster Art arbeiten, verwendete ist das *Piling*. Sie beschreibt den Vorgang (in einem realen Arbeitsumfeld) Dokumente in Stößen auf dem Schreibtisch und anderen Oberflächen im direkten Umfeld zu lagern. Das besondere an dieser Methode ist, dass subtile Informationen in der Platzierung der Stöße und der Sortierung und der Größe der Stöße versteckt sind. Da es keine vergleichbare Methode in einem virtuellen Dateisystem gibt, gehen solche Informationen verloren, bzw. gibt es diese erst gar nicht. Eine Ausnahme dafür ist der Desktop, der, wie der Name schon vermuten lässt, die Oberfläche des Schreibtisches als Metapher darstellen soll. Der Platz dort ist jedoch sehr begrenzt und wird nur teilweise für Dokumente verwendet.

Hier ist also offensichtlich Aufholbedarf gegeben, die reine Einführung von Piles geht aber meiner Meinung nach nicht weit genug. Da auch Heimcomputer im Vergleich zu früher extrem leistungsstark und trotzdem leistungsfähig sind, ist es nicht abwegig, Dateien in einer dreidimensionalen Umgebung zu speichern und wiederzugewinnen. Besonders die stärkere Einbindung des räumlichen Gedächtnisses kann weitere Vorteile bringen. Mehrere Studien [2, 11, 34] behandeln die Vor- und Nachteile von dreidimensionalen Anwendungen gegenüber zweidimensionalen in Bezug auf grafische Darstellung, Effizienz

und persönliches Empfinden. Studien, die sich dabei rein auf das Sortieren von Dateien konzentrieren, sind dabei rar. In [11] wird dieses Thema behandelt, jedoch ist diese Studie inzwischen über 17 Jahre alt und unterlag wie die damals entwickelten grafischen Dateimanager den technischen Beschränkungen, was besonders in 3D Applikationen zu visuell unästhetischen Ergebnissen führt. Aus diesem Grund ist es meiner Meinung nach an der Zeit, dieses Thema neu aufzugreifen, die Werkzeuge zum Sortieren von Daten mit neuen Mitteln zu rekreieren und die Ergebnisse der Studie(n) zu reevaluiieren.

1.2 Ziele

Das übergreifende Ziel bzw. der etwas weiter gefasste Fokus der Masterthesis ist die Vorbereitung der Computertechnologie und der Menschen, die diese Technologie verwenden, auf eine dreidimensionale Zukunft. Es gibt eine Vielzahl an vielversprechenden Forschungen, die teilweise unvorstellbare Möglichkeiten der Interaktion mit und des Feedbacks von Computern liefern. Noch befinden wir uns auf dem Weg dorthin, trotzdem sollte ein Teil der Forschung schon jetzt durchgeführt werden. Teilaspekte der später wichtigen Themen können heute bereits simuliert und evaluiert werden. Im Fall dieser Arbeit ist das die Eignung von 3D Applikationen zum Sortieren von Dateien, bzw. im späteren Verlauf als eigenständige Dateimanager. Zu diesem Zweck soll die benötigte Literatur zusammengetragen und evaluiert werden, sowie ein begleitender Prototyp zum eigenständigen Beantworten dieser Problematik entstehen. Da das Thema sehr subjektiv und die Performanz der Benutzer ebenso sehr unterschiedlich sein kann, findet der Großteil dieser Evaluierung im Zuge einer Benutzerstudie statt. So kann gewährleistet werden, dass auch ein größerer Fokus als der des Verfassers der Arbeit in Betracht gezogen wird.

1.3 Aufbau

Kapitel 2 liefert eine ausführliche Recherche zur gesamten Thematik, mit einer Auflistung der relevanten bisherigen Forschungsarbeit, sowie eine generelle Abhandlung zu dem umfassenden Thema. Die Elaboration reicht dabei von der Geschichte der Dateimanager an sich über die psychologischen Aspekte von dreidimensionalen Applikationen bis zu den verschiedenen Arten der Dokumentorganisation.

Nachdem mit Kapitel 2 ein genügend großes Grundverständnis für die Materie geschaffen worden sein sollte, werden in Kapitel 3 zuerst die grundlegenden Anforderungen an eine derartige Applikation durchgegangen. Die fertige Entwicklung eines funktionsreichen Dateimanagers birgt viele Stolperfallen, weshalb es wichtig ist, die Kernthemen zu isolieren und separat genauestens zu betrachten. Im zweiten Teil des Kapitels werden verschiedene Konzepte, mögliche Lösungen und zu beachtende Themen vorgestellt. Hier wird teilweise aufgezeigt, was alles möglich sein könnte, nicht nur mit den heute üblichen Mitteln, sondern bereits mit Blick in die Zukunft. Außerdem werden einige Aspekte aufgelistet, die im Prinzip zwar optional, für eine positive Benutzererfahrung aber überaus wichtig sind.

In Kapitel 4 werden die tatsächlich umzusetzenden Ziele für den neben der Arbeit entwickelten Prototypen aufgezählt. Da die in Kapitel 3 aufgelisteten Funktionen den Rahmen eines Masterprojektes um ein Vielfaches übersteigen würden, wird hier

beschrieben, welche davon essentiell sind, und warum andere nicht umgesetzt wurden.

In Kapitel 5 schließlich wird auf die Implementierung der im vorigen Kapitel festgelegten Ziele eingegangen, mit technischen Details aus der Sicht des Entwicklers. Die Beschreibung reicht von der Wahl des Frameworks bis hin zu den einzelnen umgesetzten Funktionen der Dateimanager.

Neben einer Sandbox für die Funktionen eines Dateimanagers war das zweite Hauptziel des Prototypen, als Testprogramm für eine Benutzerstudie zu dienen. Die Ergebnisse werden in Kapitel 6 aufgeführt und ausführlich diskutiert. Neben den organisatorischen Details der durchgeführten Studie werden hier die Ergebnisse der praktischen Studie und des beiliegenden Fragebogens ausgewertet.

In Kapitel 7 schließlich wird die gesamte Thematik aus einem subjektiveren Standpunkt noch ein weiteres Mal durchgegangen und diskutiert. Hier werden die gewonnenen Erkenntnisse aus der vorhergehenden Fachliteratur mit den selbst gewonnenen Erkenntnissen bei der Implementierung des Projektes und der Durchführung der Benutzerstudie verglichen. Ein wichtiger Punkt ist auch der Blick in die Zukunft und wie die „Vision“ hinter der Masterthesis – eine beinahe komplett dreidimensionale Interaktionsmethode mit Computern – mit dieser möglichen Zukunft zusammenspielen könnte.

Kapitel 2

Related Work

Nur wer die Vergangenheit kennt, hat
eine Zukunft.

Wilhelm von Humboldt

Diese Arbeit beginnt mit einem Rückblick auf die Geschichte der Dateimanager an sich, um einen besseren Einblick in die Designentscheidungen der Vergangenheit und Zukunft zu bekommen, und somit ein solideres Verständnis für das Thema zu schaffen. Eine Analyse der Vor- und Nachteile, und vor allem der Gründe, warum sich manche Ansätze gegenüber der Konkurrenz durchgesetzt haben, ist essentiell für das Erstellen einer derartigen Anwendung. So wie der Herstellung eines neuen Produktes eine Konkurrenzanalyse vorhergeht, geht dieser wissenschaftlichen Arbeit ein Exkurs in die Geschichte des Themas vor. Gerade in der Entwicklung eines Dateimanagers, einem der zentralsten Elemente beim Arbeiten mit einem Computer, ist eine lupenreine Umsetzung unumgänglich.

2.1 Geschichte der 2D Dateimanager

In den 1960ern war das grafische User Interface auf für heutige Verhältnisse schlichte Kommandozeilen ohne grafische Darstellung im heutigen Sinn beschränkt. Die Funktionalität war und ist immer noch die selbe, man konnte Dateien kopieren, löschen, deren Informationen auslesen, etc. All das geschieht jedoch ohne ständiges visuelles Feedback, stattdessen ist pro Aktion nur eine Ausgabe zu sehen. Den Anfang der Kommandozeile bildeten sogenannte Directory Editors, so z.B. DIREDD [43], entwickelt um 1974 von Stan Kugell am Stanford Artificial Intelligence Laboratory.

Wie in Abb. 2.1 (a) zu sehen ist, werden Dateien und Dateiinformationen zur besseren Unterscheidung farblich gekennzeichnet und angezeigt. Bei den allerersten Bildschirmen war das jedoch noch nicht möglich. Zusätzlich gibt es bei späteren Directory Editors die Option, mehrere Directories gleichzeitig zu betrachten. Wie der Name bereits vermuten lässt, liegt der Schwerpunkt beim Durchsuchen der Ordnerstrukturen, was sich vor allem durch die Beschränkung der GUI auf die Kommandozeile zeigt.

Der nächste Evolutionsschritt waren sog. orthodoxe Dateimanager, wie der 1984 herausgegebene *PathMinder* und der 1986 herausgegebene *Norton Commander*. Sie be-

stehen laut Standard [50] aus zwei Fenstern, die Verzeichnisse darstellen, und einem Fenster, das eine Kommandozeile beinhaltet. Der große Vorteil daran ist, dass der Inhalt des Verzeichnisses ständig sichtbar ist und nicht nur auf Befehl aufgelistet wird. Siehe dazu Abb. 2.1 (b). Wie auch die Directory Editors sind die orthodoxen Dateimanager darauf ausgelegt, hauptsächlich oder ausschließlich mit der Tastatur bedient zu werden. Der Unterschied zu den oben genannten Directory Editors ist, dass sich der Inhalt der Fenster beliebig wählen lässt, also auch gerade nicht direkt verwendete Verzeichnisse angezeigt werden können.

Als nächstes wurden mit den Fortschritten der grafischen Darstellung navigatorische Dateimanager eingeführt, die erstmals den Pfad bzw. die Hierarchie eines Dateisystems grafisch mit Hilfe von Icons darstellten. Ein Fenster besteht aus einer Navigationsleiste mit der Ordnerhierarchie und einem inhaltlichen Fenster für den gerade geöffneten Ordner. Es gibt auch die Möglichkeit, mehrere Fenster nebeneinander geöffnet zu haben und Dateien per Drag&Drop zwischen diesen zu verschieben. Daraus schließt sich, dass die Maus erstmals als primäres Eingabegerät gewählt wurde. Beispiele, die teilweise auch heute noch verwendet werden, sind der *MacOS Finder* und *GNOME Files* [54] auch bekannt als *Nautilus* unter Linux. Letzterer ist in Abb. 2.1 (c) und (d) zu sehen.

Die neueste Entwicklung ist schließlich die Erweiterung der navigatorischen Dateimanager zu räumlichen Dateimanagern. Der räumliche Dateimanager stellt mit jedem geöffneten Fenster immer nur einen eindeutig zuordenbaren Ordner dar, in dem Dateien wieder durch Icons dargestellt werden. Die Fenster und Icons behalten dabei ihre pseudo-physischen Eigenschaften, wie z.B. ihre Position oder Größe, bei. Es können beliebig viele Ordner dargestellt werden, diese unterliegen aber alle den folgenden Regeln:

- Ein Fenster kann immer nur einen Ordner auf einmal darstellen – keine zwei Ordner im selben Fenster.
- Jedes Fenster muss immer einem bestimmten Ordner zugewiesen sein – keine leeren Fenster.
- Alle Dateien behalten ihre Eigenschaften (Position im Ordner, Größe, Farbe, etc.) – Gebot der Konsistenz.

Die am weitesten verbreitete (= die in modernen Windows Systemen verwendete) Methode ist eine Mischung aus einem navigatorischen und räumlichen Dateimanager, die so seit *Windows 95* existiert. Auf der linken Seite eine Verzeichnisübersicht zu sehen, auf der rechten Seite der Ordnerinhalt. Beispiele dafür sind der *Windows Explorer*, der *MacOS Finder* oder das *Linux KDE* [47]. Jeder Mensch, der heute einen handelsüblichen Computer verwendet, dürfte mit dieser Methode vertraut sein, da dieser sich unter der Allgemeinheit durchgesetzt hat.

2.2 Geschichte der 3D-Dateimanager

Der Mensch ist daran gewohnt, sich im dreidimensionalen Raum fortzubewegen, hat eine ausgeprägte räumliche Vorstellungskraft und verbindet Erinnerungen oft mit Orten. Die logische Schlussfolgerung ist, dass die Navigation und Merkfähigkeit gegenüber zweidimensionalen Dateimanagern stark verbessert sein sollte; demnach gab bereits mehrere Versuche, diese Theorie in die Tat umzusetzen. Wenn man bisherige dreidimensionale Dateimanager betrachtet, erkennt man eine grobe Unterteilung in reine Dateimanager

```

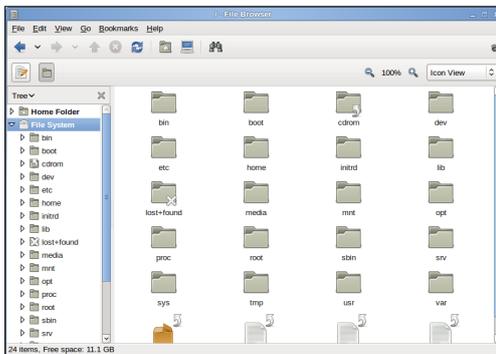
-rw-r--r-- 1 jixiuf 7.7K 11-04-04$13:00 prj...el
-rw-r--r-- 1 jixiuf 0 11-04-04$13:01 asdasd.exe
-rw-r--r-- 1 jixiuf 0 11-04-04$13:02 adsa.gz
-rw-r--r-- 1 jixiuf 1.7K 11-04-04$12:59 index.html
-rw-r--r-- 1 jixiuf 0 11-04-04$13:01 asdas.iso
-rwxr-xr-x 1 jixiuf 603K 11-04-04$12:59 server.jar
-rw-r--r-- 1 jixiuf 7 11-04-04$11:32 a.java
-rw-r--r-- 1 jixiuf 0 11-04-04$13:01 adsa.java
-rw-r--r-- 1 jixiuf 0 11-04-04$13:01 adasd.jpg
-rw-r--r-- 1 jixiuf 0 11-04-04$13:01 wdewdwe.jsp
-rw-r--r-- 1 jixiuf 0 11-04-04$13:03 adsa.mid
-rw-r--r-- 1 jixiuf 0 11-04-04$13:03 adsa.mp3
-rw-r--r-- 1 jixiuf 0 11-04-04$13:00 asdas.mp4
-rw-r--r-- 1 jixiuf 0 11-04-04$13:01 asdfsa.msi
-rw-r--r-- 1 jixiuf 0 11-04-04$13:00 adsas.pdf
-rw-r--r-- 1 jixiuf 0 11-04-04$13:00 adasdas.ppt
-rw-r--r-- 1 jixiuf 0 11-04-04$13:02 adsa.rar
-rw-r--r-- 1 jixiuf 0 11-04-04$13:01 asdas.rmvb
-rw-r--r-- 1 jixiuf 4 11-04-04$12:59 maven2.rng
-rw-r--r-- 1 jixiuf 0 11-04-04$13:00 a.txt
-rw-r--r-- 1 jixiuf 0 11-04-04$13:00 b.txt
-rw-r--r-- 1 jixiuf 0 11-04-04$13:03 adsa.wav
-rw-r--r-- 1 jixiuf 4 11-04-04$12:26 a.xml
-rw-r--r-- 1 jixiuf 8.8K 11-04-04$13:00 b.xml

```

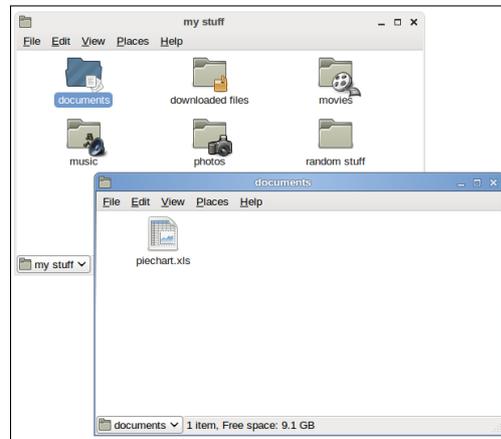
(a)



(b)



(c)



(d)

Abbildung 2.1: File Browser im Lauf der Zeit. *DIREDD*, ein Directory Editor, implementiert in *Emacs* [56] (a), *Norton Commander*, ein orthodoxer Dateimanager [50] (b), und *Nautilus File Manager* a.k.a. *GNOME*, im navigatorischen Modus (c) und im räumlichen Modus (d) [54].

und 3D-Desktops. Die letzteren beinhalten zwar teilweise auch eigene Dateimanager, jedoch liegt der Hauptfokus auf dem Desktop.

2.2.1 3D-Dateimanager

Die reinen Dateimanager wurden hauptsächlich vor dem Jahr 2000 entwickelt und unterliegen demnach auch den damals vorherrschenden Beschränkungen bezüglich Prozessor- und Grafikkartenleistung. Das wirkt sich natürlich negativ auf die Qualität der grafischen Darstellung aus, die am prominentesten ins Auge sticht und laut einer Studie [35] über 3D-Inhalte auf Websites ausschlaggebend für die Akzeptanz unter Benutzern ist. So bleiben Benutzer durchschnittlich länger auf Websites, wenn diese über qualitative 3D-Inhalte verfügen. Ein weiteres Manko dieser frühen Versuche ist die fehlende Einschränkung der sichtbaren Struktur, was zu Unübersichtlichkeit und Verwirrung führen kann. Während einige Programme zwar gezielt die größeren Strukturen darstellen möch-

ten, zerstört dieser Ansatz die Prämisse, die Navigation und die Performanz mit Hilfe der dritten Dimension zu erleichtern.

Es handelt sich hierbei natürlich um ein sehr subjektives Thema, aber meiner Meinung nach zählen Programme wie *ConeTrees* [28], *SemNet* [17] oder der *Cubix 3D File Manager* [45] zu dieser Kategorie, um nur einige zu nennen. Die Navigation durch die Dateistruktur wird hier erschwert, weil, bedingt durch Animationen, die Zeit beim Durchforsten der Ordner verlängert wird. Oft werden auch Dateien im 3D-Raum aufgereiht, sind aber nicht frei zugänglich. Sie müssen erst in den Fokus geschoben oder gedreht werden, ähnlich wie bei mit Tasten bedienten Mobiltelefonen und Controller-gesteuerten Konsolen. Es muss dabei jedes Mal auf den Abschluss der Bewegung der Kamera oder der Dateien gewartet werden, was vor allem bei älteren Systemen ruckelt und Zeit in Anspruch nimmt.

Folgend wird auf einige Beispiele näher eingegangen. Der Fokus von *ConeTrees* liegt darauf, Ordnerhierarchien im Ganzen darzustellen. Ein Level einer Hierarchie wird dabei in Form eines sich drehenden Kegels (Cone) dargestellt, und mit etwas Fantasie sieht eine so dargestellte Hierarchie wie ein stilisierter Nadelbaum aus. Laut eigener Beschreibung eignet sich *ConeTrees* hervorragend, um File Browser zu verbessern; meiner Meinung sind die Vorteile aber rein kosmetischer Natur. Von Weitem ist die grobe Struktur und Hierarchie des Systems zwar schön anzusehen, aus der Nähe ist die Darstellung aber unübersichtlich. Die Navigation wird ebenfalls nicht verbessert, eher das Gegenteil ist der Fall. Das ist vor allem durch die Überlagerung der auf der selben Stufe einer Hierarchie liegenden Kegel, zu sehen in Abb. 2.2 (a), bedingt. Die Kegel lassen sich zwar frei drehen, die im Vordergrund gelegenen Dateien werden aber nicht deutlich genug hervorgehoben und überlagern mit der die im Hintergrund liegenden, was zusammen mit der Opazität zu einer visuellen Unordnung führt. Auch in einer Studie [12] wurde evaluiert, dass Teilnehmer zwar erheblich länger brauchten, um Dateien zu finden; sie konnten sich aber viel schneller einen guten Überblick über die gesamte Struktur des Dateisystems verschaffen.

Ein weiterer früher Versuch für das *UNIX* System war *fsn* von *Silicon Graphics IRIX* [42], zu sehen in Abb. 2.2 (b), das durch seinen kurzen Auftritt in dem Film *Jurassic Park* an Bekanntheit gewann. Ordner werden durch Podeste, Dateien durch je nach Größe verschieden hohe Quader, und Ordnerrelationen durch weiße Linien dargestellt. Dadurch lassen sich gewisse Eigenschaften leicht auf einen Blick erfassen, genau wie bei *ConeTrees* ist der Vorteil gegenüber zweidimensionalen Anwendungen aber fraglich. Was im linken Teil der Abbildung besonders gut zu sehen ist, ist die relativ kontraintuitive Steuerung mit Buttons und Schiebereglern. Das Programm *fsv* [40] ist eine privat erstellte Weiterentwicklung für modernere Linux-Systeme, mit verbesserter Grafik und modernerem Design.

Der *Cubix 3D-Dateimanager* soll eine Erweiterung des Windows Explorers darstellen und Files nach zusätzlichen Kriterien sortieren, die auf die verschiedenen Achsen im dreidimensionalen Raum zugeordnet werden können. So soll die Suche nach bestimmten Dateien, von denen man den genauen Speicherort nicht mehr genau weiß, erleichtern. Die so gefilterten Daten werden schließlich als immer gleich eingefärbte Würfel in einem schwarzen Raum angezeigt, zu sehen in Abb. 2.2 (c). Während die in [45] angegebenen Filter die Suche tatsächlich verbessern könnten, ist für mich fraglich, warum die gefilterten Daten auf diese Art in 3D dargestellt werden müssen. Identisch aussehende Würfel

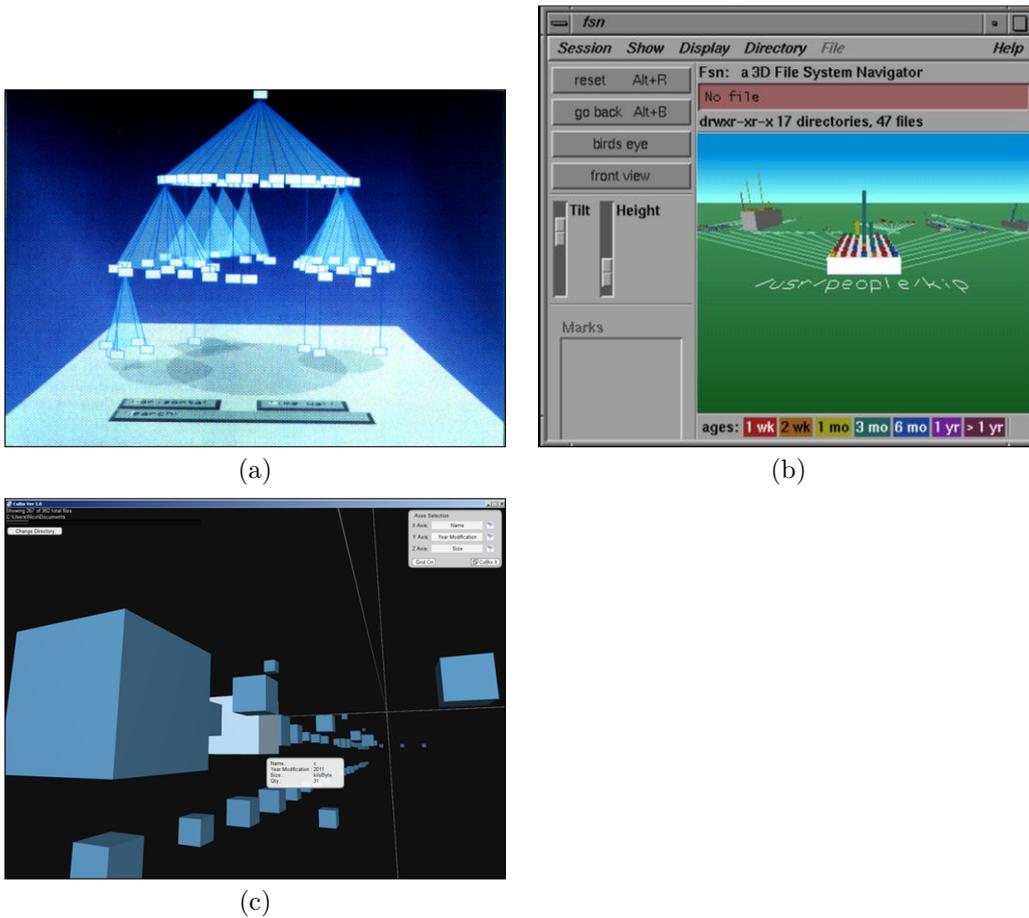


Abbildung 2.2: Beispiele für 3D-Dateimanager. Eine Ordnerstruktur, dargestellt mit *ConeTrees* [28] (a), *fsn File Manager* [42] (b), und die Navigationsansicht des *Cubix 3D File Managers* [45] (c).

auf einer unbegrenzt langen Achse verschieben nur das Problem einer unübersichtlichen, listenförmigen Dateisuche von der zweiten in die dritte Dimension.

2.2.2 3D Desktops

Viel interessanter sind in meinen Augen 3D-Desktops, da sie initial vom Einschalten des Computers an verwendet werden und somit nicht erst für die Dateisuche aktiviert werden müssen. Wie der Name vermuten lässt, wird man nach dem Starten des Computers zu einem Desktop weitergeleitet, der in 3D gestaltet ist. Bei den bisherigen Umsetzungen gibt es wie im zweidimensionalen Counterpart Icons und Widgets auf der Benutzeroberfläche. Die gesamte Applikation ist aber immersiver gestaltet und reduziert sich nicht nur auf einen einzelnen Aspekt der Verwendung von Computern wie bei den reinen Dateimanagern. Auch bei 3D-Desktops gibt es verschiedene Ansätze, auf die ich hier näher eingehen werde.

Die für Linux entwickelnde *X.org* Organisation [52] stellt den Source Code für das X Window System zur Verfügung und weist drei verschiedene 3D-Desktops vor. 2005 wurde z.B. in Frankreich *Metisse* [9] herausgegeben, das den Desktop 2.5D umsetzt. Laut eigener Definition ist *Metisse* jedoch kein 3D Desktop und hat dies auch nicht als Ziel, sondern ein Tool zum erstellen von neuen innovativen Desktop Umgebungen. Die *True3D*Shell* zeigt Icons in 3D und lässt sich mit der Tastatur durchforsten, ähnlich der Controller-Steuerung einer Spielkonsole.

Ein weitaus modernerer 3D Desktop von *X.org* ist der *Compiz Desktop Cube*, zu sehen in Abb. 2.3 (a). Dieser stellt aber ein Problem dar, das meiner Meinung nach einige 3D Desktops betrifft. Die Funktionalität eines zweidimensionalen Desktops wird mehr oder weniger eins zu eins in den dreidimensionalen Raum übertragen, was in der Abbildung gut zu sehen ist. Anstelle eines Desktop Fensters hat man nun sechs, aufgeteilt auf die Seiten eines Würfels. Das ist zwar ein netter visueller Effekt, der eigentliche Nutzen gegenüber zweidimensionalen Desktops ist aber marginal.

Ein sehr ähnlicher Ansatz, mit Fenstern auf den Seiten eines Würfels, war auch unter Windows Vista verfügbar. Wie einige der 3D-Dateimanager stellt das zwar eine nette Abwechslung gegenüber der gewohnten Umgebung dar, bindet aber die dritte Dimension viel zu wenig in den tatsächlichen Arbeitsprozess mit ein. Meiner Meinung nach sollte 3D kein Werkzeug zur kosmetischen Aufbesserung, sondern eine tiefgreifende Verbesserung der heutigen Ansätze sein.

Sehr viel interaktiver als die bis jetzt genannten Desktops ist zum Beispiel *Real-Desktop* [46], das den Desktop zusammen mit drei umschließenden Wänden in einer schrägen Ansicht von oben zeigt, zu sehen in Abb. 2.3 (b). Icons und Dateien werden als dünne Kacheln dargestellt und können frei im Raum bewegt und platziert werden. Sie weisen dabei sehr realistische Merkmale in Bezug auf Gewicht und Kollisionsberechnung auf. Des weiteren können sie in beschriftbare Kisten wie im linken Teil der Abbildung thematisch einsortiert oder an die Wände gehängt werden.

Der für diese Arbeit wohl interessanteste 3D Desktop ist *BumpTop* [38], entwickelt von Anand Agarawala und Ravin Balakrishnan an der University of Toronto. Er wurde wie der *RealDesktop* 2006 entwickelt und ist diesem im Aussehen sehr ähnlich, legt aber noch viel mehr Wert auf das Stapeln der Dateien und das visuelle Sortieren derselben. Außerdem ist er nicht ausschließlich als Alternative für einen 2D Desktop, sondern als neue Interaktion für Computer mit Touchpen (in älteren Versionen) oder Multitouch, gedacht. Eine Maussteuerung wird auch unterstützt. Die Vorteile des Stapelns von virtuellen Daten behandeln die beiden in dem im selben Jahr herausgegebenen Paper [2], sowie Agarawala in seiner Masterarbeit [1]; dieses Thema wird später noch ausführlicher behandelt. *BumpTop* ist in meinen Augen noch ausgereifter als *RealDesktop*, vor allem Dank der sinnvollen Verwendung von Kontextmenüs und der intuitiven Steuerung, der eine ausführliche wissenschaftliche Recherche vorhergeht.

2.3 Grundlagen des Dateimanagements

Interessante Themen für diese Arbeit sind unter anderem die Vorteile eines realistischer gestalteten Arbeitsplatzes gegenüber den gebräuchlichen zweidimensionalen Desktops, vor allem mit Hinblick auf die Unterschiede von Piling und Filing. Weitere Themen von Interesse sind die Unterschiede von 2D und 3D in Bezug auf die Leistung, die

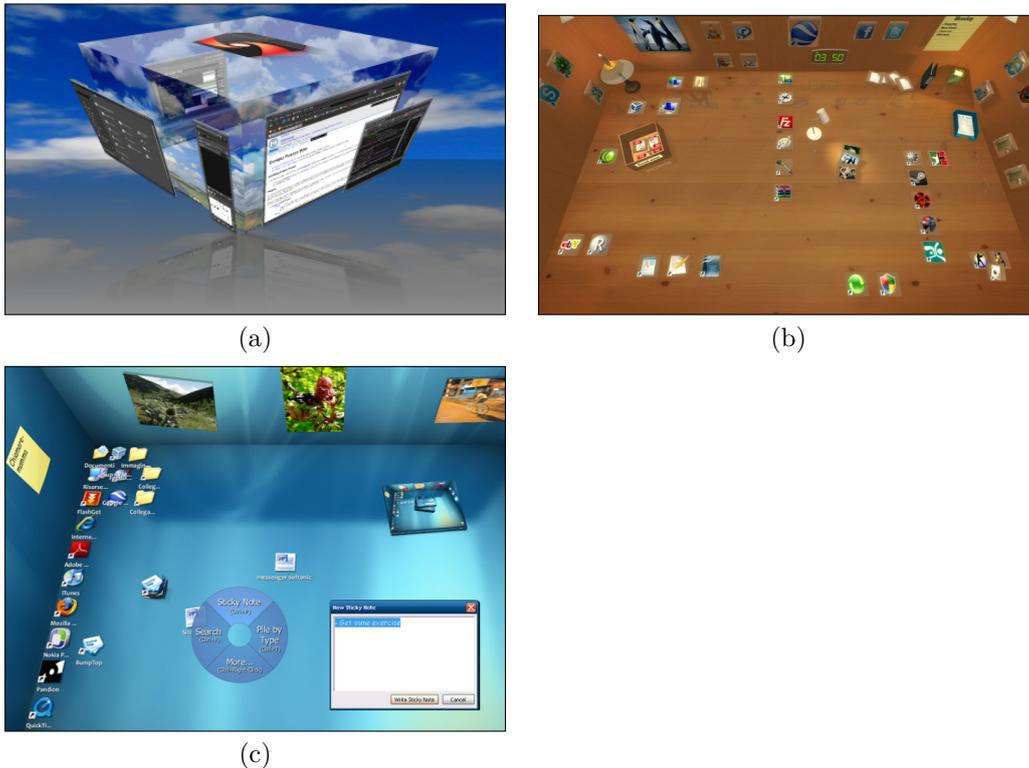


Abbildung 2.3: Beispiele für 3D Desktops. *Compiz Desktop Cube*, der mehrere Desktops auf einem Würfel abbildet (a), *RealDesktop*, ein 3D Desktop mit Physik (b), *BumpTop*, ein 3D Desktop mit Fokus auf Piling, mit geöffnetem Kontextmenü (c).

Organisation und vor allem das persönliche Empfinden der Benutzer; die Vorteile von 3D in Bezug auf die Merkfähigkeit sind ebenso ein wichtiger Punkt.

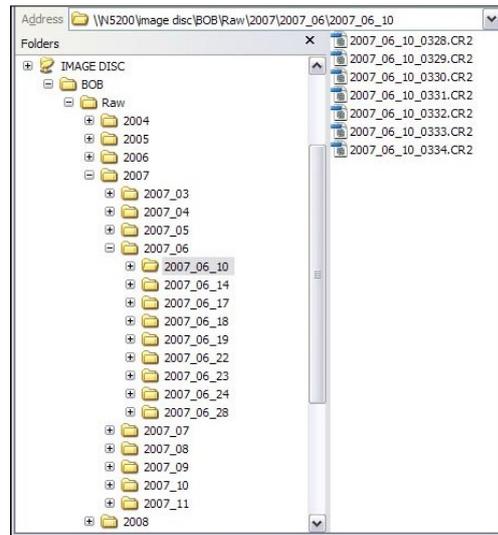
Die heute üblichen, meist riesigen Mengen an Daten, mit denen Personen privat und im Arbeitsumfeld täglich umgehen, bringen einen ebenso großen Organisationsaufwand mit sich. Auf privater Ebene bzw. im Heimgebrauch beschreibt die Dokumentenorganisation die Erstellung oder Beschaffung, Organisation und Pflege von privaten Dokumenten und Dateien, hauptsächlich durch den Besitzer derselben. Oft wird dieser Schritt zwischen den anderen Aufgaben im Lauf der täglichen Arbeit verrichtet und nicht erst als eigene Aufgabe angesehen. Da diese Art des Dokumentmanagements so häufig und in so vielen Situationen verwendet wird, können hier kleine Verbesserungen zu einer großen Effizienzsteigerung führen [7, 20].

2.3.1 Filing und Piling

In verschiedenen Studien [2, 20, 25] wurde versucht, die Organisationsgewohnheiten von Büroangestellten und Menschen, die häufig mit dem Computer arbeiten, zu analysieren und kategorisieren. Die zwei dominanten entdeckten Strategien sind „Piling“ (dt. stapeln) und „Filing“ (dt. einordnen), erstmals erwähnt in [25]. Beispiele dafür sind in Abb. 2.4 (a) und (c) zu sehen. Piling und Filing sind zwei sehr unterschiedliche Ansätze



(a)



(b)



(c)



(d)

Abbildung 2.4: Beispiele für Filing (oben) und Piling (unten), links in einem realen Umfeld, rechts im virtuellen. Bildquellen: (a) <http://www.amoseyinc.com/>, (b) <https://www.tricolorlab.com/basic-workflow>, (c) <http://www.pieragravenor.com/piles-on-my-desk/>, (d) <https://www.reddit.com/>.

der eigentlich analogen Informationsaufbewahrung, die aber auch in virtuellen Dateisystemen widergespiegelt werden. Andere Bezeichnungen dafür sind „neat filing“ und „messy filing“.

Thomas Malone führte 1983 eine Studie [25] durch, um die organisatorischen Gewohnheiten von Büroangestellten zu studieren und Grundlagen für computerbasierte Organisationssysteme zu bilden. Er fand in den verschiedenen Stapeln und Archiven subtile, jedoch sehr aufschlussreiche Hinweise auf die Art und die Frequenz der Bearbeitung, die seiner Meinung nach in digitalen Systemen verloren gehen würden. Deshalb ist es wichtig, solche Aspekte bei der Entwicklung eines Dateimanagers zu berücksichtigen.

Die Teilnehmer seiner Befragung reichten dabei von Forschern über Sekretäre zu Managern. Eine der wichtigsten Erkenntnisse war, dass Dinge nicht nur speziell platziert werden, um sie später wieder zu finden, sondern auch, um als Erinnerung für bestimmte Aufgaben zu dienen. Ein Studienteilnehmer merkte an, dass seine platzierten Stöße als visuelle Denkanstöße gelten, und je weiter sie von ihm entfernt sind, desto seltener be-

achtet er sie und desto unwichtiger sind sie für seine derzeitige Arbeit. Malone schätzte, dass ca. 67% der Stöße auf den Schreibtischen der Teilnehmer mit bevorstehender Arbeit zu tun hatten, also als eine Art Todo-Liste und Ablage in einem gesehen werden konnten. Er fügte hinzu, dass dadurch, dass Stöße kein System haben, die Position oft umso wichtiger ist, damit sie wieder gefunden und verwendet werden können.

Die andere identifizierte und dem Stapeln gegenüber stehende Art der Organisation ist das Einordnen oder Filing. Das Filing und Kategorisieren wurde dabei von vielen Teilnehmern als "kognitiv schwierig" eingestuft. Ein Teilnehmer meinte "Es ist interessant, wie schwer es ist, diese Dinge zu charakterisieren [...] Es ist fast so als sie wegzulassen, würde bedeuten, sie nicht charakterisieren zu müssen [...] sie wegzulassen bedeutet, die Entscheidung, wie man sie benutzt oder wo man sie hingibt für jetzt zu verschieben." Da die Studie ergab, dass der Großteil der untersuchten Arbeitsplätze unbenannte Stöße enthielt, empfahl Malone, dass auch virtuelle Desktops diese Funktion zur Verfügung stellen sollten und sowohl verzögerte Klassifizierung als auch betitelte, logisch angeordnete Dateien und Archive unterstützen sollten.

Die Verwendung von solchen Stapeln in einem Computersystem würde die Arbeit natürlicher gestalten und den Workflow erheblich erleichtern. Weitere Ansätze zur weiteren Verbesserung der Erinnerung in computergestützten Systemen umfassen die Wiedergabe der Wichtigkeit einer Datei durch Frequenz, Größe, Ort und Farbe bzw. Opazität. Je öfter eine Datei also aufscheint, je größer sie ist und abhängig von der Platzierung (an bestimmten Orten oder zentraler im Sichtfeld), desto wichtiger ist sie für den Benutzer. Außerdem kann die Wichtigkeit automatisch angepasst werden, so könnten zum Beispiel Dateien, die lange nicht verwendet wurden, über Zeit größer werden oder in den Vordergrund rutschen, um nicht vergessen zu werden. Andere, vergleichsweise unwichtige Dateien, könnten hingegen mit der Zeit kleiner werden und irgendwann ganz verschwinden.

2001 verglichen Steve Whittaker und Julia Hirschberg [32] ebenfalls die verschiedenen Arten der Organisation von Arbeitsunterlagen, wobei sie ebenfalls die Vorteile von Piling bestätigen konnten. Sie erhoben qualitative und quantitative Daten von Angestellten, die kurz vor einem Umzug in kleinere Büros standen und sich daher einen sehr genauen Überblick über ihre Unterlagen machen mussten. Ungefähr ein Drittel der Teilnehmer wurde als "Piler" identifiziert und hatte 40% seiner Akten in unsortierten Stapeln gelagert. Es ist aber anzumerken, dass alle Teilnehmer sowohl Piling als auch Filing verwendeten.

Die Vorteile des Pilings sind die Einfachheit der Wartung, der kleinere Overhead bei der Wiedergewinnung der Daten und der generell leichtere Zugang. Die in Malones Forschung [25] angesprochene Funktion der Stapel als visuelle Erinnerung wurde in [32] ebenfalls bestätigt. Piler haben einen schnelleren, einfacheren Zugang zu kürzlich verwendeten Informationen und greifen aufgrund des kleineren Aufwandes öfter auf ihre Stöße zu als Filer auf ihre Ordner und Akten. Information, die oft gebraucht wird, ist außerdem automatisch weiter oben im Stapel und wird deswegen leichter gefunden, Information, die weniger oft gebraucht wird, wandert immer weiter nach unten. Die Archive von Pилern waren auch um 60% kleiner, was daran liegt, dass selten gebrauchte Daten aus Stößen leichter verworfen werden als die aus Archiven.

Filing hingegen hat einen starken Overhead, zum einen bei der initialen Erstellung des Archivs, zum anderen bei der Wartung desselben. Einige Teilnehmer archivierten

nur, wenn sie dabei Unterstützung durch einen Sekretär hatten. Ein weiterer Nachteil ist, dass selten verwendete Informationen nur ungern weggeworfen werden, vermutlich wegen des dafür aufgebrauchten Aufwandes beim Archivieren. Nur 7% der Filer gaben an, es falle ihnen leicht, ihre Archive zu säubern, gegenüber 43% der Piler. Zusätzlich werden Dokumente oft voreilig archiviert, woraufhin sich später herausstellt, dass sie kaum oder nicht gebraucht werden. Es stellt sich auch als schwierig heraus, Daten nach mehreren, teilweise übergreifenden, Kategorien zu archivieren, oder es wird eventuell eine Kategorie vergessen und daraufhin eine duplizierte erstellt.

2.3.2 Filing und Piling am Computer

Sarah Henderson und Ananth Srinivasan [20] merken an, dass die gängige Methode des Filings in Computersystemen aus einer Zeit kommt, in der noch mit wenigen hundert Dateien gearbeitet wurde, und Computer nur von gut ausgebildeten Spezialisten verwendet wurden. Inzwischen werden Computer aber von Menschen mit verschiedensten Ausbildungshintergründen und teilweise nur minimalem Vorwissen verwendet, welche damit um einiges anspruchsvollere Aufgaben bestreiten; der Aufbau der User Interfaces hat sich in den letzten Jahrzehnten aber nicht großartig verändert. Henderson und Srinivasan gehen sogar noch härter mit Dateimanagern ins Gericht und schreiben

Despite [the widespread introduction of graphical user interfaces], the user interfaces of [Macintosh and Windows] were not designed for modern document management tasks.

Natürlich ist auch Piling nicht ohne Nachteile. Es ist zum Beispiel ab einem gewissen Punkt nicht mehr gut skalierbar, und die gesuchte Information ist mit einer wachsenden Anzahl an Stapeln und Daten schwieriger zu finden. Wenn die Anzahl an Stapeln zu sehr steigt, können sie außerdem jede freie Oberfläche eines Büros oder eines Desktops belegen. Während Piles für die gerade verwendeten Dateien von Vorteil sind, wird es mit zunehmender Anzahl von Dateien und Zeit schwieriger, einzelne Dateien in den Piles zu finden. Das gilt für Computer wie reale Büros gleichermaßen. Im Computersystem liegt das daran, dass die vom Betriebssystem zur Verfügung gestellten Speicherstrukturen rein für Filing gedacht sind, und Piling nicht unterstützt wird. In der Realität können solche Piling-Ordner oder ein voller Desktop wie in Abb. 2.4 (d) mit unsortierten und quasi wahllos gefüllten Aktenschränken verglichen werden. Trotz der aufgezählten Vorteile und dem Fakt, dass ein Drittel der Befragten hauptsächlich Piling verwendet, gibt es jedoch keine vergleichbare Technologie in den gängigen Betriebssystemen.

Warum dann überhaupt Filing verwenden, wenn Piling doch viel besser für einen dynamischen, schnellen Arbeitsprozess ist? Die Antwort ist, wenig überraschend, um Dateien und Dokumente auch nach längerer Zeit wiederzufinden [22], was durch verschiedene Arten der Sortierung (z.B. alphabetisch oder thematisch) erleichtert wird. Wenn nie archiviert wird, wächst der Berg an Piles immer weiter, bis Daten irgendwann nur noch mit einer ausführlichen und arbeitsintensiven Suche gefunden werden können. Es ist daher natürlich sinnvoll, vor allem wichtige Daten ordentlich in ein System, sei es nun reell oder digital, einzusortieren, trotzdem sollte auch in Computern ein System zum Stapeln von Daten zur Verfügung stehen. Ein temporärer Ordner, in den Dateien vor dem Einsortieren zwischengespeichert werden, erfüllt diesen Zweck nur teilweise, da einige Vorteile des Pilings nicht genutzt werden können.

2.4 Persönliche Organisationstypen

In ihrer Studie definieren Henderson und Srinivasan [20] außerdem drei verschiedene persönliche Organisationstypen, sortiert nach dem Level der Organisation „Piler“, „Filer“ und „Structurer“. Piling und Filing hat hier nichts mit den zuvor beschriebenen Methoden zu tun, sondern mit persönlichen Organisationsvorlieben im vorgegebenen Dateisystem, im Fall der Studie der Windows Explorer unter Windows XP. Alle genannten Organisationstypen verwenden also aufgrund mangelnder Alternativen die Methode des Filings, auch wenn Piling um einiges besser geeignet wäre. In der Realität würde aber eine Mischung aus Filing und Piling allen drei Typen Vorteile im Arbeitsprozess bringen.

Schon jetzt können bei der Arbeit mit digitalen Dokumenten also Piler identifiziert werden, das Dateisystem ist aber nicht darauf ausgelegt. Diese Zielgruppe würde stark von einem ins Betriebssystem integrierten Piling-Mechanismus profitieren. Piler lagern Dateien oft direkt auf dem Desktop oder in einem Ordner ab, ohne diese nach einer bestimmten Eigenschaft einzusortieren. Wenn die Menge an Daten zu groß und unübersichtlich wird, werden sie kurzerhand in einen neuen Ordner verschoben und das Piling beginnt von Neuem.

Falls die Daten bei der Suche sortiert werden, dann meistens nach Bearbeitungsdatum, da die wichtigen Dateien erst kürzlich gebraucht wurden. Hier spiegelt sich ein Aspekt des Pilings im echten Leben wider. In Bezug auf Ordnerstruktur und Aufbau haben Piler flache Hierarchien und oft keine tiefen Ordnerstrukturen. Sporadisch versuchen manche Piler, ihre Daten zu sortieren, nicht jedoch, weil sie großen Nutzen darin sehen, sondern weil sie das Gefühl haben, es müsse so sein. Das Ziel dieser Gruppe ist „Kleinstmöglicher Aufwand – Größtmögliche Zugänglichkeit“.

Die Filer charakterisieren sich dadurch, dass sie, sobald notwendig, Ordner für neue Dateien erstellen und diese thematisch passend in die vorhandene Baumstruktur einbauen. Die Struktur weist normalerweise eine mittlere Tiefe und immer noch relativ hohe Breite auf, sowie einige nicht klassifizierte Dateien auf oberster Ebene. Neue Ordner werden hinzugefügt, wenn die Menge an Dateien in einem alten Ordner zu groß wird, eine neue Kategorie auftritt oder während einer Säuberungsaktion (für alte, nicht mehr benutzte oder nicht klassifizierte Dateien). Die bevorzugte Suchmethode ist das manuelle Durchgehen der Struktur nach Gedächtnis. Filer würden vom Piling insofern profitieren, dass die in der obersten Ebene gesammelten Dateien separat gelagert werden, und dort nach den in Abschnitt 2.3.1 genannten Kriterien des Pilings automatisch sortiert werden. Wie im realen Umfeld würden auch weniger Dateien archiviert werden, was Arbeitsaufwand erspart.

Henderson und Srinivasan beschreiben zusätzlich noch eine dritte Kategorie, die „Structurer“. Der Unterschied zum Filing ist, dass die Ordnerstruktur nicht nur existiert und gewartet wird, um das Finden von Dateien zu erleichtern, sondern um einen generellen Überblick über die Struktur der eigenen Informationen zu geben. Aus diesem Grund werden Ordner und Ordnerstrukturen bereits im Vorhinein erstellt, bevor etwaige Inhalte dafür vorhanden sind, weil es hilfreich ist, die konzeptuellen Strukturen vor sich zu haben. Die Ordnerstrukturen sind hier normalerweise schmal und tief, und haben kaum unkategorisierte Dateien in den oberen Ebenen. Durch die sehr durchorganisierte Struktur greifen die diese Organisation verwendenden User am seltensten auf die

Suchfunktion zurück, da sie meistens genau wissen, wo ihre Dateien liegen, oder diese anhand der Namen und der Struktur der Ordner leicht zu finden sind. Außerdem reevaluieren sie ihre Struktur gerne im Lauf einer Suche. Durch die starke Untergliederung verlassen sich Structurer bei der Suchfunktion darauf, dass zusätzlich zum Dateinamen der Speicherort oder die Hierarchie angezeigt wird, da der Kontext ausschlaggebend für den Erfolg der Suche ist.

2.5 Räumliches Gedächtnis und Suchstrategien

1994 und 1995 hielten Deborah Barreau und Bonnie Nardi et al. Studien [3, 26] über die Organisations- und Suchgewohnheiten auf Desktops ab, Barreau hauptsächlich mit MS DOS und Nardi und ihre Kollegen hauptsächlich mit Mac OS. 1995 entschieden sie sich, ihre Ergebnisse in einem weiteren Paper [4] zu vergleichen. Wie schon in Malones [25] Studie auf realen Schreibtischen gab der Speicherort die meiste Information über die Wichtigkeit der Daten preis. Wichtige Daten werden zum Beispiel dort platziert, wo sie am ehesten gesehen werden, Daten, die bald zu löschen sind, neben dem Papierkorb.

Im Grunde wurden zwei verschiedene Strategien zum Finden von Dateien charakterisiert: logische Suche und Orts-basierte Suche. Bei der logischen Suche wird ein textueller Suchterm verwendet, um über die im Betriebssystem gelegene Dateisuche die gewünschte Datei zu finden. Bei der Orts-basierten Suche wird versucht, anhand der Ordnerstruktur zu erraten oder sich zu erinnern, wo die Datei liegen könnte. Dieser Vorgang kann und muss teilweise mit mehreren Teilen des hierarchischen Baumes wiederholt werden. In beiden Studien [4, 25] bevorzugten die Teilnehmer sehr stark die Orts-basierte Suche, die logische Suche wurde als langsam empfunden und wurde nur bevorzugt, wenn der Dateiname eindeutig bekannt war. Wenn die Teilnehmer sich nicht genau an den Namen erinnern konnten, zögerten sie von Beginn an, die logische Suche zu benutzen.

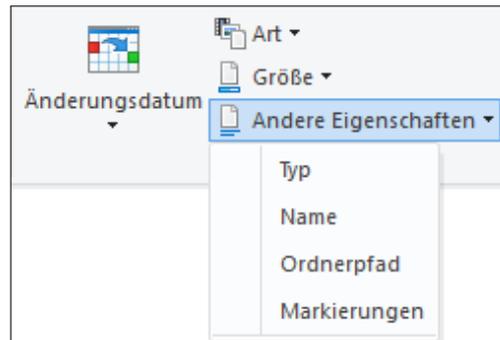
In der Studie wurde der richtige Ordner bei der Orts-basierten Suche in den meisten Fällen sofort gefunden, danach wurden die Dateien im Ordner nach bestimmten Kriterien wie alphabetisch oder nach Bearbeitungsdatum sortiert, um die Suche zu erleichtern. Auf die logische Suche wurde in diesem Fall erst zurückgegriffen, wenn der richtige Ordner bzw. die richtige Datei nicht gleich gefunden wurde.

Barreau und Nardi stellten in ihrem Studienvergleich [4] große Gemeinsamkeiten zwischen ihren Ergebnissen mit jeweils MS DOS und MAC OS Benutzern fest und hielten sie in vier Punkten fest:

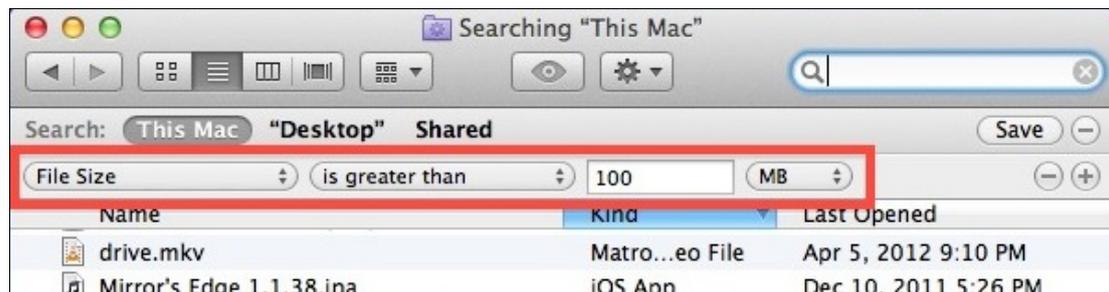
To our surprise, we found striking similarities across the two study populations. The similarities were:

- A preference for location-based search for finding file.
- The critical reminding function of file placement.
- The use of three types of information: ephemeral, working and archived.
- Lack of importance of archiving files.

Beide Studien lassen aus sich schließen, dass hauptsächlich die Art, wie die Daten später verwendet werden, ausschlaggebend dafür ist, ob und wie diese archiviert werden.



(a)



(b)

Abbildung 2.5: Die erweiterten Optionen der Windows Dateisuche unter Windows 10 (a) und Mac OS X (b). Das Menü 'Art' bei Windows enthält Kategorien wie 'Programm', 'Film' oder 'Dokument'. Bei Mac OS können mit dem „Plus“-Button (rechts von der roten Box) ebenfalls weitere Kategorien hinzugefügt werden. Bildquelle (b) [53]

Eine weitere Studie [27] von Ravasio et al. zeigte, dass auch versierte Benutzer nur auf die logische Suche zurückgreifen, wenn die Datei nicht mehr anders gefunden werden kann und praktisch als verloren angesehen wurden. Es sollte aber angemerkt werden, dass sich die Suchfunktionen seit der Durchführung dieser Studien erheblich verbesserten, zum Beispiel können bei der Suche jetzt mehrere Dateiattribute angegeben werden, zwei Beispiele dafür sind in Abb. 2.5 zu sehen; außerdem beschleunigten sie sich durch die verbesserte Prozessorleistung, und vor allem wegen der verbesserten Zugriffsgeschwindigkeit der Datenträger.

Es gibt keine neueren Studien zu dem Thema, aber die Befragung der Studienteilnehmer in Kapitel 6 ergab, dass nach wie vor ein Großteil der Benutzer die manuelle, Orts-bezogene Suche bevorzugt. Es gibt aber Services wie Cloud-Speicher, zum Beispiel Dropbox [39] oder Google Drive [44], welche schnelle Suchalgorithmen verwenden und beinahe sofort zutreffende Ergebnisse liefern. Sie verwenden außerdem ebenfalls weitere Suchkriterien wie Typ, Änderungsdatum oder enthaltene Wörter. Gerade bei großen Datenmengen, die von mehreren Geräten automatisch auf die Cloud übertragen werden, wird der Sortieraufwand schnell zur Last. Nicht nur Cloud-Speicher, auch der Windows Explorer stellt inzwischen einige weitere Suchkriterien zur Verfügung, zu sehen in Abb. 2.5 (a).

Trotz der verbesserten Suche nahmen die Verfasser der Studien an, dass Ortsbezogenen Suche eine tiefgreifendere Einbindung und Interaktion in die (meistens selbst erstellte) Dateistruktur darstellt, und deshalb ebenfalls bevorzugt wird. Eine weitere ihrer Thesen ist, dass es für Benutzer befriedigender ist, Nutzen vom eigenen räumlichen Gedächtnis zu machen, um Dateien und Dokumente zu finden.

Es zeigt sich hier sehr deutlich, dass die Frage „Wo habe ich die gesuchte Datei platziert?“ von fast allen Benutzern sehr viel leichter beantwortet werden kann als die Frage „Wie heißt die Datei, die ich suche?“. Hier könnte die Umsetzung in 3D einen Vorteil bringen, sofern sie in anderen Punkten ähnlich vorteilhaft ist als der 2D Counterpart.

2.5.1 Klassifizierung von Dateien nach Verwendungsart

Anzumerken ist auch noch, dass bei den genannten Studien [3, 4, 26] auf digitalen Desktops drei verschiedene Arten von Information identifiziert werden konnten:

- *Ephemeral* (dt. *flüchtig*). Diese Daten haben eine kurze Lebensdauer und beziehen sich auf Memos, Todo-Listen oder zwischengespeicherte Informationen. Sie werden meist auf dem Desktop abgelegt und nicht erst thematisch einsortiert.
- *Working* bzw. arbeitsbezogene Daten, die mit der derzeitigen Arbeit verknüpft sind und eine Lebensdauer von mehreren Wochen oder Monaten haben. Diese Daten werden häufig abgerufen, und ihre Position ist gut im räumlichen Gedächtnis des Nutzers verankert.
- *Archived* bzw. archivierte Daten, die viele Monate oder teils Jahre lang gespeichert werden und als Informationsrücklage für vergangene Projekte dienen. Die Daten, die dieser Kategorie zugeordnet werden können, sind überraschenderweise nicht sehr zahlreich.

Je nach Art der Datei machen zu gewissen Zeitpunkten der Verwendung verschiedene Lagerungsstrategien mehr oder weniger Sinn.

Ephemere Daten: Diese werden kaum jemals in ein Archiv aufgenommen werden, trotzdem werden sie z.B. in Form von Text-Dokumenten auf dem Desktop gespeichert, und sind damit Teil des File-Archivs. Eventuell werden sie sogar vergessen und unbeabsichtigt in ein Archiv verschoben, was ganz und gar nicht dem Zweck solcher Informationen dient.

Arbeitsbezogene Daten: Diese werden zum Teil in ein Archiv übernommen, doch der Großteil wird nach verrichteter Arbeit wieder verworfen. Diese Informationen würden in einer realen Arbeitssituation von den meisten Menschen in Form von Piles gelagert, die regelmäßig durchforstet werden.

Archivierte Daten: Diese entsprechen ganz dem ursprünglich gedachten Zweck eines Dateisystems, und sind dementsprechend gut für Computer geeignet. In der Realität arbeitet aber niemand mit ausschließlich archivierten Daten und sortiert diese sofort nach der Verwendung wieder in das Archiv ein.

2.6 2D vs. 3D

Im vorigen Abschnitt wurde festgestellt, dass die Einführung von Piles massive Vorteile bringen kann. Mit einem dreidimensionalen Umfeld könnte das visuell ansprechend umgesetzt und in die natürlich gehaltene Interaktion eingebunden werden. In diesem Abschnitt will ich mehr darauf eingehen, inwiefern sich die Erweiterung um eine Dimension im visuellen, grafischen Sinn auf das Benutzerverhalten auswirken kann.

Die Frage, worin die genauen Unterschiede zwischen 2D und 3D in Bezug auf Performanz und Benutzererfahrung liegen, wurde in mehreren Studien [11, 29, 34], unter anderem vom Militär und der Luftfahrtindustrie, behandelt. So zum Beispiel Wickens, Liang, Prevett und Olmos, die die Unterschiede von 2D- und 3D-Displays beim Landeanflug von Flugzeugen verglichen. Hier zeigten sich gemischte Ergebnisse. Bei einem dreidimensionalen Display etwa sei die laterale (links-rechts) Navigation leichter als bei einem zweidimensionalen, die vertikale (oben-unten) Navigation sei jedoch erschwert.

Zweidimensionale Displays verbessern wiederum das Geländebewusstsein, was jedoch den Erkenntnissen von John et al. [31] widerspricht. Diese geben an, das Geländebewusstsein sei bei dreidimensionalen Displays verbessert, sie unterstützen aber die Ergebnisse, die angeben, dass die vertikale Navigation auf zweidimensionalen Displays besser funktioniert als in 3D. Vor allem die Aufgabe, den höchsten Punkt auf einer Karte zu finden, sei vielen Probanden auf zweidimensionalen Displays leichter gefallen, was auch in [33] bestätigt wird.

Wenn man von der Luftfahrt absieht und diese Fragen auf den normalen Gebrauch von Computern anwendet, werden die Leistungs-Unterschiede wesentlich geringer. Cockburn und McKenzie führten mehrere Studien [10, 11, 13, 14] zu dem Thema durch, auch mit Hinblick auf das räumliche Erinnerungsvermögen, wobei sie verschiedene Programme verwendeten. So zum Beispiel [11] das Dokument-Sortierwerkzeug *DataMountain* [29], das Thumbnails von Dokumenten wie Websites in einer 2D oder 2,5D Umgebung darstellt, um die Geschwindigkeit beim Sortieren und beim Wiederfinden jener Dokumente zu messen und zu vergleichen.

Die Fragen, die sie damit beantworten wollten, waren "Gibt es Unterschiede in der Performanz?" und "Welche persönlichen Präferenzen haben die Benutzer?". In der Studie wurde darauf Wert gelegt, dass die Aufgaben genau gleich gestaltet wurden und der einzige Unterschied die zusätzliche Dimension ist, um möglichst aussagekräftige Daten zu erhalten. Um einem Lerneffekt vorzubeugen, wurden die Teilnehmer außerdem in zwei Gruppen geteilt, die jeweils nur eine Version des Programmes benutzten.

Die Ergebnisse gaben an, dass die Testpersonen mit dem 2D Interface jeweils schneller waren; wie auch beim Vergleich von Kommandozeile und der Fensterversion, bietet die einfachere Lösung das Potential zu höherer Effizienz (wenngleich die Lernkurve um ein Vielfaches steiler ist). Cockburn und McKenzie führen das vor allem auf das relativ kleine Fenster und die noch kleineren (max. 64x64 Pixel [29]), weiter entfernten Thumbnails in der 3D Version des Programmes zurück, das keine Veränderung des Sichtfeldes zulässt. Sieht man sich Abb. 2.6 an, wird schnell ersichtlich, dass Thumbnails im hinteren Teil der Anwendung schwer zu unterscheiden sind. Die Probanden konnten sich zwar schnell an die ungefähre Position des richtigen Thumbnails erinnern, brauchten aber länger als in 2D, um durch die eng angeordneten Dokumente zu navigieren. Außerdem sind wir hier grafisch betrachtet, bedingt vor allem durch die technischen Beschränkungen,



Abbildung 2.6: *DataMountain*, ein 2.5D Tool zum Sortieren von Dokumenten mit Hilfe von Miniaturansichten. Gut ersichtlich ist, wie klein und unübersichtlich die Thumbnails im Hintergrund werden. Bildquelle [11]

noch weit von einer befriedigenden dreidimensionalen Lösung entfernt.

Viel wichtiger für diese Arbeit ist jedoch die Erkenntnis, dass die dreidimensionale Variante deutlich ansprechender auf die Probanden wirkte. Diese Aussage trifft auch auf andere Anwendungsgebiete, wie zum Beispiel die Interpretation von zwei- und dreidimensionalen Graphen [24], zu.

Sehr wichtig ist es daher bei der Implementierung eines dreidimensionalen Dateimanagers, die Menge an Daten im richtigen Maß zu beschränken, da die Leistung bei einer zu großen Menge an Daten sowohl in der Realität als auch im virtuellen Raum bei einer zu großen Datenmenge nachlässt [14].

Kapitel 3

Anforderungen und Konzepte

Früher oder später werden zweidimensionale Monitore und Displays sehr wahrscheinlich von einer weiter entwickelten, vermutlich in irgendeiner Form dreidimensionalen Technologie abgelöst werden. Was zurzeit noch der Inhalt von futuristischen Science Fiction Filmen wie *Avatar* [36] (holographische Anzeige) oder *Ready Player One* [37] (fortgeschrittene Virtual Reality-Ausrüstung), und das Thema ambitionierter Forschungen ist, wird in der einen oder anderen Form auch in der Realität Einzug in den täglichen Gebrauch halten. Eine dieser Forschungen [30] hat es sich sogar zur Aufgabe gemacht, die Displays aus Science-Fiction-Filmen erfolgreich nachzubauen, wie aus einem Interview mit *ScienceDaily* [49] hervorgeht:

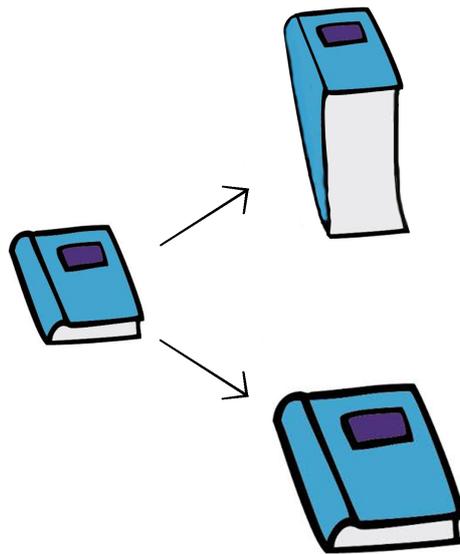
We refer to this colloquially as the Princess Leia project. Our group has a mission to take the 3D displays of science fiction and make them real. We have created a display that can do that.

Da in diesem Bereich eine große Menge an Zeit und Geld investiert wird, ist es nur eine Frage der Zeit, bis Displays bzw. Interaktionen dieser Art für die Verwendung in alltäglichen Geräten bereit sind. Aus diesem Grund sollte mit Prototypen und realistischen Simulationen bereits jetzt erforscht werden, worauf in Bezug auf Benutzerfreundlichkeit später Wert gelegt werden muss und welche Besonderheiten bei der Entwicklung dieser relativ unerforschten Interaktionsmethode mit Computern vorliegen. Wenn bereits vor der eigentlichen Entwicklung Hindernisse entdeckt und umgangen werden können, erleichtert das den Umstieg von 2D auf 3D enorm. Nicht umsonst haben sich Dateimanager in den letzten 20 bis 30 Jahren kaum verändert; wenn der Umstieg nicht optimal verläuft, könnte eine vielversprechende Technologie aufgrund von Anfangsfehlern abgelehnt werden.

Mein Ziel für diese Arbeit ist es, mit Hilfe einer ausführlichen Recherche und einer begleitenden Studie lehrreiche Informationen zu gewinnen, die bei der Entwicklung einer dreidimensionalen Desktop- oder Dateimanipulationssoftware helfen sollen. Diese Software könnte für „normale“ Monitore, Augmented oder Virtual Glasses oder jegliche zukünftig hergestellte Technologie entwickelt werden. Um die Forschung in Rahmen einer Masterarbeit zu halten, wird hier die Eignung des dreidimensionalen Ansatzes als Organisationstool getestet, und außerdem die bereits erwähnte Studie [11] von Cockburn und McKenzie in einem kleineren Maßstab, dafür aber in einem grafisch weitaus fortgeschrittenerem Umfeld, wiederholt.

Tabelle 3.1: Anforderungen an einen Dateimanager

<i>Faktor</i>	<i>Hauptanforderungen</i>
Dateimenge	In 3D ist es besonders wichtig, die angezeigte Dateimenge genug zu reduzieren bzw. zu simplifizieren, um die visuelle Unordnung gering zu halten. Dies ist einer der wichtigsten Punkte, da die Daten durch die dreidimensionale Darstellung bei gleicher Sichtbarkeit mehr Platz benötigen als Daten in 2D.
Dateityp	Da der Dateiname nicht unbedingt sichtbar ist, muss der Designer entscheiden, wie er Informationen über den Dateityp visualisiert. Zu wenige Informationen könnten den Benutzer in seiner Arbeit behindern, bei zu vielen verschiedenen verwendeten 3D-Modellen besteht wiederum die Gefahr einer zu steilen Lernkurve, bis der Benutzer diese mit dem richtigen Dateityp assoziiert.
Feedback über Meta-informationen	Ein weiterer Aspekt der dreidimensionalen Dateimanager ist die Möglichkeit, Metainformationen auf sehr unterschiedliche und subtile Weise in das Erscheinungsbild von Dateien einzubauen.

**Abbildung 3.1:** Beispiel für unterschiedliche Darstellungsmöglichkeiten der Dateigröße. Die relative Höhe oder die Gesamtgröße kann erhöht werden, oder auch das Gewicht.

3.1 Faktoren

Es gibt unzählige Faktoren, die bei der Implementierung eines Dateimanagers beachtet werden müssen, und die Liste der möglichen Verbesserungen der Benutzerfreundlichkeit und -hilfen ist quasi endlos. Trotzdem will ich die meiner Meinung nach wichtigsten hier auflisten. Die jeweilige Kurzfassung ist in Tabelle 3.1 zu sehen, und in den folgenden Abschnitten wird auf jede noch näher eingegangen.

3.1.1 Dateimenge

Da die Menge an (auch privaten) Dateien in einem Ordner, und vor allem im gesamten Dateisystem, schnell eine überschaubare Anzahl übersteigt, müssen diese zusammengefasst oder der angezeigte Informationsinhalt reduziert werden. Andernfalls können sich die angezeigten Dateien zu ungeordneten und unübersichtlichen Stapeln oder Haufen formen, die keine Information über die enthaltenen Daten preisgeben. Im Gegenteil, durch die Unordnung und Überlappung werden Dateien vor dem Benutzer verborgen, und müssen, wie in der Realität, zeitaufwändig gesucht werden.

Hier gilt es, sinnvolle und, wenn möglich, gleichzeitig ästhetisch ansprechende Methoden zu finden, die die Menge an Daten reduzieren oder diese automatisch sortieren. Ein Beispiel wären Ordner darstellende Schachteln, die mehr Objekte beinhalten können, als ihre Größe physikalisch zulässt. Oder die Komprimierung einer größeren Menge an Daten in einen kleinen Ball oder Orbit. Solche Überschneidungspunkte aus Realität und Virtualität zeigen das große Potential der dreidimensionalen Dateimanager.

Sie beinhalten natürlich auch die Gefahr, aus Faulheit oder Zeitdruck zu anfänglich bequemen Aufschubhilfen zu werden, welche im Nachhinein wiederum mehr Aufwand zum Sortieren benötigen. Das Besondere an einem realistisch gehaltenen Dateimanager ist aber, dass die Realität abgebildet und mit verschiedensten Hilfen verbessert werden kann. So kann das aufgeschobene Sortieren durch die Anwendung erleichtert oder ganz übernommen werden.

Wenn große Mengen an Daten aber wieder entpackt werden, braucht es Wege, diese sinnvoll und übersichtlich darzustellen. Einige solcher Möglichkeiten sind verschiedene Sortier- und Auffächerungsmethoden, die einem unübersichtlichen Haufen oder einem vollen Ordner Struktur und Ordnung verleihen. Agarawala hat dies in seinem Thesisprojekt [1] so umgesetzt, dass die auf dem Desktop befindlichen *Piles*, bei denen zuerst nur die oberste Datei zu sehen ist, auf verschiedene Arten aufgefähert und nach bestimmten Kriterien sortiert werden können. Seine Ansätze sind:

- Rasteransicht – Sehr ähnlich wie in einem Ordner in einem zweidimensionalen Dateisystem werden die Dateien nach einem Raster ausgerichtet.
- Blätteransicht – Wie Seiten in einem Buch werden die Dateien dicht aneinander gereiht und können durchgeblättert werden.
- Fischaugenansicht – Alle ausgewählten Dateien werden in einer Kette aufgelegt, nur das Icon des derzeit ausgewählten ist voll sichtbar. Die anderen überlappen sich und werden zum Rand hin kleiner.
- Kompressionsansicht – Ähnlich wie in der Blätteransicht geht man der Reihe nach durch die Dateien, anstatt der Umblätteranimation werden nicht angesehene Dateien aber auf eine Linie komprimiert.
- Fächeransicht – Die Dateien werden wie ein Kartendeck nach den Richtungsangaben des Benutzers aufgefähert.
- *Pile*-Ansicht – Hier kann stufenlos zwischen einem geordneten Stapel und einem ungeordneten *Pile* hin- und hergewechselt werden.

In einem frei beweglichen 3D-Dateimanager kommt der Vorteil hinzu, dass die dritte Dimension noch ausgiebiger verwendet werden kann. Abb. 3.2 zeigt, wie die Fächeransicht beispielsweise umgesetzt werden kann. Wie in Abb. 3.2 (a) zu sehen ist, bringt die

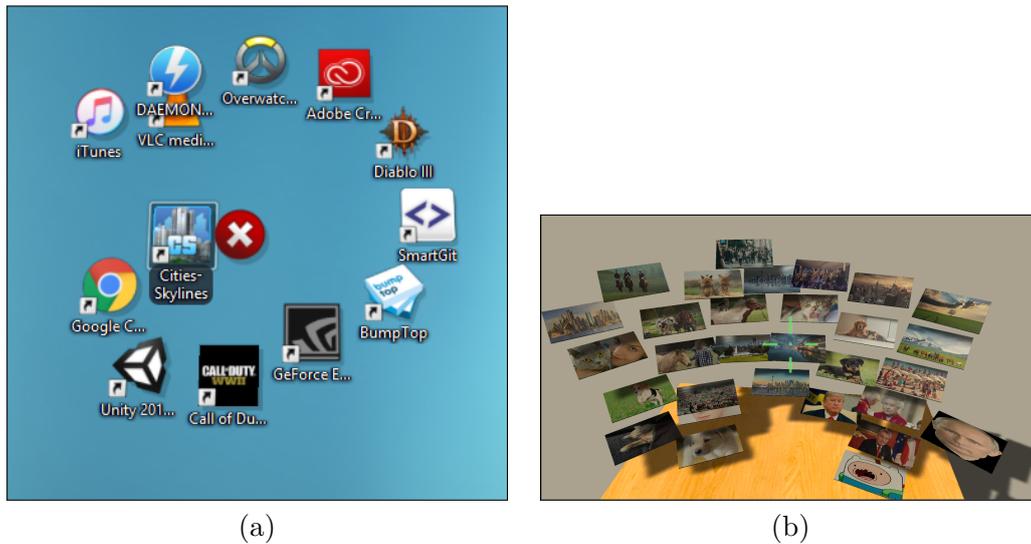


Abbildung 3.2: Beispiele für die Auffächerung von Dateien. In *BumpTop* (a), mit einem geordneten *Pile* und einem aufgefacherten daneben, und in meinem Prototypen (b).

Auffächerung ohne Dateinamen allerdings nicht viel Nutzen, da nur die jeweiligen Icons angezeigt werden. In einer älteren Version von *BumpTop* wurden diese beispielsweise nicht angezeigt, was den Benutzer dazu nötigte, seine Dateien auswendig zu kennen. Doch selbst das bringt nichts, wenn mehrere Dateien mit der selben Endung nahe beieinander liegen oder die Dateien umsortiert werden.

In Abb. 3.2 (b) ist eine mögliche Umsetzung in meinem *First-Person* Prototypen zu sehen, ebenfalls ohne implementierte Anzeige der Dateinamen. Während die ausgeblendeten Dateinamen gerade bei Bildern noch nebensächlich sein können, sollten sie für andere Dateien definitiv eingeführt werden. Durch die Nutzung des dreidimensionalen Raumes und der kreisförmigen Auffächerung kann ein Benutzer beispielsweise nur mit der Bewegung einer Virtual Reality-Brille seine Dateien erkunden.

Analog zu den von Agarawala umgesetzten automatischen Methoden muss es dem Benutzer möglich sein, seine Daten selbst zu ordnen und zu sortieren, zum Beispiel mit Hilfe der zuvor erwähnten Schachteln. Je mehr Funktionen dabei von den bekannten 2D-Dateisystemen oder Metaphern aus dem realen Leben übernommen werden, desto leichter fällt den Benutzern der Umstieg und die Eingewöhnung. Schachteln, die eindeutig Unterordner repräsentieren, sollten benennbar und einfärbbar sein, und bereits auf den ersten Blick zumindest teilweise Informationen über den Inhalt preisgeben. Die Möglichkeiten sind hier sehr vielseitig und sollten Elemente aus dem realen Leben mit neuen, surrealen Methoden verbinden. Beispiele dafür sind die Manipulation der Gravitation und anderer physikalischer Eigenschaften, Röntgensicht, die Einblendung von Verhältnissen, die Änderung der eigenen Größe, etc.

3.1.2 Dateityp

Inwieweit der genaue Dateityp besonders bei persönlichen Dateien wichtig ist, sei dahingestellt. Dennoch sollte man sich zumindest darüber Gedanken machen, wie die

Dateikategorie (z.B. Bild, Video, Dokument) dargestellt werden soll, um Informationen subtil weiterzugeben. Durch die dreidimensionale Natur stehen hier mehrere visuell ansprechende Möglichkeiten zur Verfügung, etwa indem der tatsächliche Inhalt oder ein Teil davon direkt am 3D-Modell als Textur dargestellt wird. Besonders Bild- und Textdateien eignen sich dafür, da sie selbst in eine Textur umgewandelt werden können und somit keine weiteren Informationen für die Darstellung nötig sind.

BumpTop [38] und *RealDesktop* [46] benutzen zur Repräsentation der Dateien flache Kacheln mit dem jeweiligen Icon als Textur (mit Ausnahme von Bilddateien, die das enthaltene Bild anzeigen), meiner Meinung nach schöpft das die vielfältigen Möglichkeiten aber nicht genug aus. Während Icons und Miniaturansichten in 2D Sinn ergeben, können in 3D weitaus mehr und bessere Metaphern und Realitätsabbildungen umgesetzt werden.

Ein weiterer Faktor ist, dass möglicherweise nicht alle Dateien einzeln angezeigt werden sollten. Der Inhalt von Programmordnern, die eine Menge (für Normalverbraucher) nichtssagende binär-Dateien und Programmbibliotheken enthalten, sollte zum besseren Verständnis zusammengefasst und somit verborgen werden. Eine Idee dafür wäre eine kleine Box mit einem Knopf, der die enthaltene *executable*-Datei bzw. deren Verknüpfung ausführt, während die zugrundeliegenden Dateien verborgen bleiben.

Die Liste der oft verwendeten Dateiendungen ist lang und wächst stetig, darum werden hier die wichtigsten aus der privaten Nutzung aufgezählt [55]:

- Mediendateien (Fotos, Videos, Musik/Audio),
- komprimierte Archive,
- ausführbare Dateien,
- Dokumente (Text, Tabellen, Präsentationen),
- Systemdateien.

In einem professionellen Umfeld kommen noch weitere oft verwendete Endungen hinzu. Hier ist es außerdem wichtiger jederzeit den Dateinamen sowie die Dateiendung zu sehen, um sinnvoll mit den Dateien arbeiten zu können. Ob hier überhaupt dreidimensionale Dateimanager verwendet werden und sich schließlich durchsetzen, wird sich erst zeigen. In vielen Bereichen, wie zum Beispiel der Programmierung, kommt die Kommandozeile nach wie vor regelmäßig zum Einsatz.

3.1.3 Visuelles Feedback über Metainformationen

Wie Agarawala und Balakrishnan es bereits in ihrem Paper [2] ansprechen, gibt die Position in einem *Pile* bereits subtile Informationen über die Benutzung einer Datei preis. Ebenso kann es sich mit anderen Metainformationen verhalten, wenn diese in 3D auf die Darstellung der Dateien angewandt werden. Die Dateigröße etwa kann, wie in dem Beispiel in Abb. 3.1, durch die physikalische Größe, die Dicke oder durch das Gewicht aussagekräftig wiedergegeben werden. Die Häufigkeit der Benutzung kann durch unterschiedliche Opazität dargestellt werden, oder durch die Verwendung anderer, „gealterter“ 3D-Modelle. Während bis zu einem gewissen Grad auch das Aussehen der Dateien im zweidimensionalen Counterpart verändert werden kann, stechen Veränderung bei dreidimensionalen Modellen jedoch mehr ins Auge.

3.2 Realitätsnähe und Virtual Reality

Wie bereits in diesem und auch in Kapitel 2 beschrieben, birgt die dreidimensionale Nähe zur Realität sowohl Vorteile für den Benutzer bezüglich Merkfähigkeit und Akzeptanz [14] als auch Potential für die Verbesserung der allgemeinen Benutzererfahrung. Demnach sollte das Hauptaugenmerk darauf liegen, den Benutzer so gut wie möglich in die virtuelle Welt eintauchen zu lassen.

Das wird zum Beispiel durch die Einbindung von Virtual Reality und Handheld-Controllern wie denen der HTC Vive [15] ermöglicht. Die Controller der Vive haben genaue Bewegungssensoren für die Drehung und den Winkel der Hand, außerdem wird ihre absolute Position zusammen mit der des Benutzers (= des Headsets) im Raum verfolgt. Die Vorteile des Positionstrackings sind enorm, die Immersion und das Raumgefühl sind bereits sehr nahe an der Realität. Technologien wie diese gewährleisten einen sehr natürlich erscheinenden Umgang mit virtuellen Objekten, und eine optimale Bewegungsfreiheit und -steuerung im dreidimensionalen Raum. Bei der Entwicklung der begleitenden Projekte war eines der Hindernisse, beim Bewegen und Drehen von Objekten die zweiachsige Bewegung der Maus auf die drei Achsen des virtuellen Raumes zu übertragen, was zu einer unbefriedigenden Manipulation der Objekte führte. In einer Virtual Reality-Anwendung würde die umständliche Handhabung der Objekte wegfallen, und sofern sich alles Nötige in Reichweite befindet, auch die Bewegung durch den Raum.

3.2.1 Virtual Reality-Handschuhe

Wenn man sich neueste Forschungsprojekte wie *Haptx*¹ oder *VRGlu*² ansieht, wird klar, dass die Immersion in die virtuelle Welt und der Umgang mit virtuellen Objekten noch sehr viel realistischer werden wird. Der in Abb. 3.3 zu sehende Handschuh etwa kann punktgenau virtuelle Berührungen auf die Hand übertragen. Dadurch wird das menschliche Gehirn eher davon überzeugt, die in der virtuellen Realität gesehenen und berührten Objekte seien echt. Umgesetzt wird das haptische Feedback mit Hilfe von 120 über den Handschuh verteilten Gel-polstern, deren Druck auf die Hand stufenlos verstellbar ist. Zusätzlich können die einzelnen Finger in ihrer Bewegung eingeschränkt und zurückgehalten werden, sodass ein realistisches Gefühl beim Hochheben und Zusammendrücken von Objekten entsteht. Durch den zusätzlich angeregten Sinn „akzeptiert“ das menschliche Gehirn die virtuelle Realität eher als real [8] und taucht tiefer in das Programm ein.

Noch sind Handschuhe dieser Art sehr klobig, mit dicken Kabelsträngen mit dem Computer verbunden, und aufgrund des Prototyp-Status teuer in der Produktion; sie weisen den Blick aber in eine vielversprechende Zukunft. Sobald die Technologie ausgereifter, die Handschuhe (oder Ganzkörperanzüge) leichter und billiger werden, werden sie eine größere Verbreitung finden und in mehr und mehr Bereiche der Interaktion mit Computern vordringen. Sie kann auch für Simulationen für noch weiter entwickelte Technik, wie zum Beispiel das am Beginn der Kapitels erwähnte Interface von *Avatar*, dienen. Ähnlich wie die Simulation von Auto-Crashtests würde es den Entwicklern

¹<https://haptx.com/>

²<https://vrglu.com/>



Abbildung 3.3: Der Virtual Reality-Handschuh der Firma *Haptx*, der haptisches Feedback mit 120 aufblasbaren Gel-Polstern auf die Hand überträgt. Zusätzlich können die einzelnen Finger in ihrer Bewegung eingeschränkt werden. Bildquelle <https://www.engadget.com>

solcher Technologien teure Testdurchläufe mit echten Prototypen ersparen.

Ein großer Vorteil der Verwendung von Virtual Reality als Simulation für neue Interfaces ist die Minimierung des Verlustes bei gleich bleibendem Risiko bezüglich der Durchsetzung von verschiedenen Technologien. Niemand weiß im Vorhinein, welcher Ansatz sich im Endeffekt durchsetzen wird, als Vergleich dient etwa der Kampf der *Disc-Formate Blu-Ray* gegen *HD DVD*. Trotz relativ gleicher Voraussetzungen ging *HD DVD* schließlich aufgrund von Geschäftsverbindungen und der *Blu-Ray*-Unterstützung der *Playstation 3* unter. Man sieht, oft kann durch unvorhersehbare und unzusammenhängende Gründe ein Produkt vom Markt verschwinden oder aber diesen dominieren.

3.2.2 Bewegungssteuerung

Da das Ziel der Applikation die Bearbeitung und Manipulation der eigenen Dateien ist, kann eine realistische Bewegungssteuerung außen vorgelassen werden. Die eigentliche Frage ist, ob und wie die Bewegung umgesetzt werden soll. Da freie Bewegung in Virtual Reality leicht zu Übelkeit führen kann [21, 23], sollte darauf vielleicht ganz verzichtet werden. Eine in VR-Spielen oft verwendete Umsetzung ist die Teleportation zu fixen Punkten oder zu einem selbst mit dem Controller projizierten Punkt. Die Verwendung von echten *Walking Simulators*³ schießt hier wohl über das Ziel hinaus, zumal in der angesprochenen Anwendung die räumliche Bewegung gegenüber den Hand- und Armbewegungen eindeutig zweitrangig ist.

3.2.3 Metaphern

Zusätzlich zu moderner Technik ist es hilfreich, realitätsnahe Metaphern für die verwendeten Objekte einzubauen, wie zum Beispiel die korrekte Berechnung der Biegung von Papier (in diesem Fall Dokumente oder Notizzettel im 3D-Dateimanager). Bereits 2001 erforschte Beaudouin-Lafon [5] die Vorteile von Papier-ähnlichen Fenstern gegenüber den starren überlappenden Fenstern, die bis heute verwendet werden. Er ermöglichte es in seinem Programm die Ordnerfenster bis zu einem gewissen Grad zu drehen und nach dem Prinzip der *Pile*-Metapher scheinbar ungeordnet übereinanderzulegen. Zu-

³Zum Beispiel *Infinadeck*; <http://www.infinadeck.com/>



Abbildung 3.4: An die Realität angelehnte Mechaniken wie diese erschaffen für Benutzer eine immersive Benutzererfahrung. Bildquelle <http://gnupart.tistory.com/entry/Unity3D-Simple-Page-Curl-Effect>

sätzlich implementierte er die papierähnliche Physik für die offenen Fenster, ähnlich wie in Abb. 3.4. So können diese stufenlos verbogen und verdreht werden. Der Desktop blieb dabei trotzdem 2D, die Fenster sahen nur so aus, als wären sie in 3D gefaltet worden. Trotzdem kam die Illusion einer realitätsnäheren Arbeitsumgebung sehr gut bei den Studienteilnehmern an.

Die zugrundeliegenden Ordner, die zurzeit durch Möbelstücke dargestellt werden, können ebenfalls durch verschiedene andere Metaphern dargestellt werden. Laut [22] werden Archive (im Fall der Studie von Forschern und Universitätsprofessoren) nicht nur verwendet, um Dinge wiederzufinden; so werden auch Archive erhalten, deren Hauptzweck deren Sichtbarkeit für die Nachwelt ist, als eine Art Erbe. Andere Archive werden zum Teilen von Ressourcen erstellt. Wieder andere präsentieren die wichtigsten Errungenschaften ihrer Karriere gut sichtbar für Besucher. Wenngleich sich nicht alle dieser Anwendungsfälle auf digitale Dateisysteme übertragen lassen, sollten die verfügbaren Möbelstücke und Archivierungsoptionen für verschiedene Arten der Archivierung gedacht und optimiert werden.

3.3 Augmented und Mixed Reality

Eine weitere vielversprechende Technologie für einen anderen Umgang mit Daten ist Augmented Reality. Anstelle von den zwei Bildschirmen in einer Virtual Reality-Brille, die dem Benutzer eine komplett neue, virtuelle Welt vorgaukeln, werden virtuelle Elemente auf die durchsichtige Scheibe der Brille projiziert. Dadurch lassen sich Objekte auf realen Gegenständen wie zum Beispiel der eigenen Schreibtischplatte, den umgebenden Wänden oder dem Boden anzeigen.

Das Potential liegt hier in der Erweiterung des eigenen Schreibtisches um virtuelle Elemente, vor allem Piles. So wie ein Teil des Schreibtisches meistens mit unsortierten Dokumenten, Notizzetteln und Arbeitsutensilien bedeckt ist, kann ein Teil der virtuellen Arbeitslast „aus dem Bildschirm heraus“ verlagert werden. Microsoft's *HoloLens* bietet bereits die Funktion, virtuelle Notizzettel und Anwendungsfenster auf verschiedene Oberflächen zu projizieren, Dateimanipulation bleibt aber noch außen vor.

Ein Konzept für eine in meinen Augen sinnvolle Beispielanwendung ist in Abb. 3.5 zu sehen. Die rechts im Bild in blau eingblendeten virtuellen Ordner dienen als Erweiterung des vorhandenen Bildschirms um ein weiteres (Ordner-)Fenster. Das am Rand des Tisches abgelegte virtuelle Dokument kann, wie in [25] und [32] beschrieben,

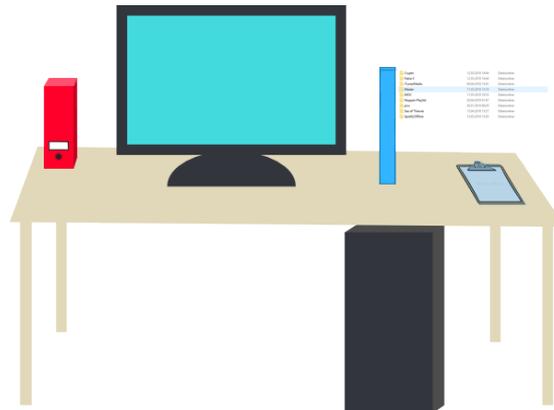


Abbildung 3.5: Ein mögliches Anwendungsgebiet für Augmented Reality zur Dateibearbeitung am Computer. Links in rot ein realer Ordner mit Dokumenten, rechts in blau ein auf die Tischplatte projizierter Ordner und ein virtuelles Dokument.

als visuelle Erinnerung für spätere Arbeit dienen. Die Tischplatte selbst könnte einen Ordner im Dateisystem darstellen, oder einfach Verknüpfungen zu anderen Dokumenten aufweisen.

Ganz gleich auf welche Weise sie verwendet wird, Augmented Reality Technologie bietet eine wertvolle Erweiterung für einen beispielsweise von Programmierern wenig genutzten Raum, sei es zu Hause oder im Büro. Anstatt sich in vielen geöffneten Fenstern am PC zu verirren, können Dateien nun auch in die reale Welt ausgelagert werden und profitieren dabei von den selben Sortier- und Ordnungsmethoden, die in Abschnitt 3.1.2 bereits aufgezählt sind.

Ein Konzept von Mixed Reality ist die Verwendung von Wänden als Programmfenster. Dieses Konzept kann meiner Meinung nach auch in Virtual Reality bzw. den dreidimensionalen Ansatz einer Desktops übertragen werden. So können Flächen im Programm gezielt für Aufgaben und andere Programmfenster verwendet werden, was wiederum das räumliche Gedächtnis beansprucht und die virtuelle Welt geographisch und thematisch unterteilt.

Kapitel 4

Designziele

Teil dieser Masterarbeit ist die Implementierung eines Projektes; das Ziel dieses Projektes wiederum ist ein Prototyp eines Dateimanagers, eine Art Sandbox zum Testen und Evaluieren verschiedener Mechanismen der Datenmanipulation. Einige der in Kapitel 3 beschriebenen Anforderungen wurden dabei mit eingebaut, jedoch nicht alle. Und selbst die gesamten beschriebenen Anforderungen sind nur ein sehr kleiner Teil dessen, was für einen 3D-Dateimanager möglich wäre. Eines der Hauptziele der entwickelten Sandbox ist eine Grundlage für eine Benutzerstudie, um öffentlichere Meinung und Informationen aus anderen Blickwinkeln zu erhalten.

4.1 Grundlagen der Applikation

Die Grundlagen der Applikation können von praktisch jeder beliebigen Spiel-Engine übernommen werden, die über *Utility*-Klassen verfügt, die die Manipulation und Erstellung von Dateien ermöglichen. Die selbstständige Entwicklung einer Engine würde das Vielfache an Zeit in Anspruch nehmen und den eigentlichen Fokus der Arbeit – die Organisation von Daten in 3D – auf die reine Programmierarbeit verschieben. Die Engine übernimmt außerdem die unbedingt notwendigen, jedoch nicht primär wichtigen Teile des Programmes wie:

- Die Berechnung der Physik und Kollisionen,
- die korrekte Kameraführung / Steuerung des „Spielers“,
- die Belichtung,
- etc.

Während diese Bereiche unumgänglich für die vollständige Funktion einer derartigen Applikation sind, nimmt die Entwicklung derselben mehr Zeit in Anspruch als die Entwicklung der korrekten Manipulation der Dateien selbst. Da es sich bei diesem Projekt um einen Prototypen handelt, ist die Verwendung einer Spiele-Engine definitiv gerechtfertigt. Für die Entwicklung des eigentlichen Programmes sollte jedoch auch die Engine selbst entwickelt werden und tiefer im Betriebssystem selbst verwurzelt sein. Dadurch sind eine höhere Performanz und weniger Probleme mit etwaigen Berechtigungen des Betriebssystems gewährleistet. Sollte das Dateisystem vollständig auf diesen Ansatz umgestellt werden, kann eventuell die Repräsentation von Ordnern ganz weggelassen

werden, wodurch die Dateien praktisch nur noch mit ihren Positionsdaten gespeichert werden und keine überliegende Hierarchie vorhanden ist.

4.2 Funktionen zur Dateimanipulation

Die wichtigsten grundlegenden Funktionen zur Dateimanipulation sind das

- Verschieben,
- Kopieren,
- und Löschen

von Dateien und sind demnach unabdingbar bei der Planung eines Dateimanagers. Folgend wird auf diese Funktionen und deren Umsetzung in 3D näher eingegangen.

4.2.1 Verschieben von Daten

Während der Prozess des Verschiebens in einer Kommandozeile nicht wirklich ersichtlich ist, kann er in den navigatorischen und dreidimensionalen Dateimanagern gleich umgesetzt werden. Die einfachste Lösung ist das Verschieben der Datei per Drag&Drop, also das Aufheben der Datei mit einem gedrückt gelassenen Linksklick und das Ablegen der Datei am gewünschten anderen Ort mit dem Loslassen der Maustaste. Im Hintergrund findet dabei der Verschiebeprozess statt.

Es muss aber darauf Rücksicht genommen werden, wo im Dateisystem sich die Datei befindet, während sie zwischen Ursprung und Ziel bewegt wird. Es könnte beispielsweise der freie Raum zwischen diesen beiden Orten bereits durch einen Ordner repräsentiert werden. Ich habe mich jedoch dafür entschieden, die Datei im Dateisystem im Ursprungsordner zu belassen, bis diese in einem neuen Ordner angekommen ist. So kann die Datei leichter wiedergefunden werden, sollte sie verloren gehen oder zwischen Herkunftsort und Ziel liegen gelassen werden.

4.2.2 Kopieren von Daten

Zusätzlich zu der Möglichkeit eines Kontextmenüs oder dem Ctrl+C, Ctrl+V Befehl denke ich, eine aktivere Option kann hier von Nutzen sein. Die Datei kann etwa in eine speziell markierte Box, auf ein Fließband, das durch eine „Black Box“ läuft, oder auf das Modell eines Kopierers gelegt werden, um den Kopiervorgang zu beginnen. Wenngleich es mehr Zeit benötigt, gibt es dem Benutzer ein realistisches Gefühl beim Arbeiten mit seinen Daten, und der zusätzliche Aufwand könnte zum Nachdenken über die Sinnhaftigkeit des Vorgangs und die Organisation anregen.

4.2.3 Löschen von Daten

Hier gilt das selbe Prinzip wie beim Kopieren, zusätzlich oder anstatt des Tastenanschlagens kann die Datei in einen „echten“ Mülleimer geworfen oder einfach zusammengeknüllt werden wie ein Blatt Papier. Hier kann der Fantasie freier Lauf gelassen werden und viele verschiedene Arten der Zerstörung von Dateien angewendet werden, sei es nun z.B. Feuer oder eine Schere. Der spielerische Ansatz macht die Benutzererfahrung angenehmer und kann dem Benutzer Freude beim Arbeiten bereiten [16].

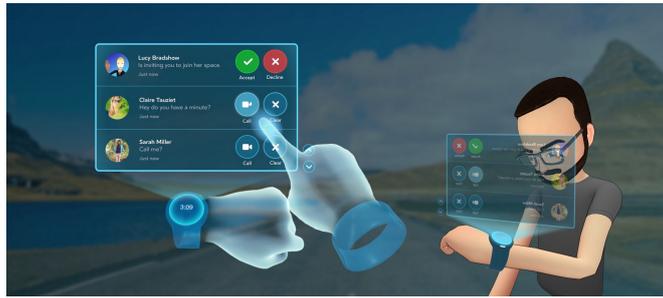


Abbildung 4.1: Konzept für ein tragbares Virtual Reality Kontextmenü. In der angegebenen Quelle sind weitere Konzepte zu finden. In einer VR-Anwendung können mit Konzepten wie diesem textbasierte Menüs auf praktische, einfache und zugleich ästhetische Weise umgesetzt werden. Bildquelle <https://medium.com/>

4.2.4 Weitere grundlegende Funktionen

Da sich das Thema dieser Arbeit um die Organisation dreht, werden nicht alle der Grundfunktionen umgesetzt. Der Vollständigkeit halber werden sie hier aber noch aufgelistet:

- Das Erstellen von Dateien,
- das Umbenennen von Dateien,
- die Dateisuche,
- die Manipulation von Dateiattributen, -eigenschaften und -berechtigungen.

Diese Funktionen sind unerlässlich für einen Dateimanager, werden aber aufgrund der geringen Auswirkung auf die Organisation außen vor gelassen. Die Dateisuche wird ebenfalls nicht implementiert, da für die Studie dem räumlichen Erinnerungsvermögen hier der Vorrang bei der Suche gegeben wird. In einer vollständigen Applikation könnte diese Funktion umgesetzt werden, indem die Datei (oder ein Abbild davon mit einem Indikator der echten Position) zum Benutzer fliegt, oder die Datei mit einem besonderem Shader auch durch andere Gegenstände sichtbar ist.

Während diese Funktionen zurzeit nicht Teil der Designziele für den Prototypen sind, werden sie in Zukunft für die Weiterführung des Projektes wieder interessant. Erwähnenswert ist auch, dass im Lauf der weiterführenden Implementierung Funktionen „erfunden“ werden können, die aufgrund der Neuartigkeit der Dateimanipulation noch nie verwendet wurden.

4.2.5 Optionale Funktionen

Aus Sicht der Aufgabe des Computers nicht wichtig, für den Benutzer aber beinahe unerlässlich, sind Hilfsmethoden, die die Orientierung und Manipulation der Dateien erleichtern. Gemeint sind Funktionen wie das Kontextmenü, die für ein schnelles Arbeiten nicht unbedingt notwendig sind, dem Benutzer aber eine Menge Arbeit ersparen. Alle Funktionen des Kontextmenüs sind zum Beispiel über die Kommandozeile oder das jeweilige Programm ebenfalls verfügbar, es entsteht aber ein erheblicher zusätzlicher Aufwand beim Suchen oder Ausführen des Befehls.

Es gibt hier bereits verschiedene Konzepte, besonders die für Virtual Reality gedachten sind sehr unterschiedlich und interessant. Ein vielversprechendes Beispiel dafür sind virtuelle *Wearables*, zu sehen in Abb. 4.1. Zu sehen ist ein *Avatar*, der ein Menüfenster auf seinem Handgelenk projizieren kann. Durch die in Kapitel 3 vorgestellten Handheld Controller wird dadurch eine komfortable Steuerung gewährleistet. Je nachdem, ob der Dateimanager für einen bestimmten Zweck gedacht ist und nicht alle Funktionen besitzen soll, kann ein Kontextmenü auch vernachlässigt werden. Für einen „echten“ Dateimanager sind das Menü und weitere mit *Wearables* umsetzbare Funktionen auf jeden Fall notwendig.

Kapitel 5

Implementierung

Dieses Kapitel geht auf die tatsächlich umgesetzten Ziele aus Kapitel 4 ein, sowie auf die Art und Durchführung der jeweiligen Implementierung. Im Laufe des Projektes entstand ein Prototyp für einen *First-Person* Dateimanager, der separat zum Betriebssystem und dessen Dateimanager handelt und nicht direkt darin eingebunden ist. Dieser wird wie ein normales Programm gestartet und manipuliert im Hintergrund die Dateien im System. Dieser Vorgang kann zum Beispiel mit einem separat geöffneten Ordnerfenster verfolgt werden.

5.1 Grundlagen

Zur Umsetzung des Grundsteines des Projektes wurde die bewährte Spiel-Engine *Unity* [48] verwendet. Diese stellt alle nötigen Werkzeuge zur Erstellung einer *First-Person* Applikation bereit. Zur visuellen Umsetzung wurden frei verfügbare 3D-Modelle aus dem *Unity Asset Store* verwendet und in das Projekt eingebunden. Der gesamte selbst verfasste Code des Projektes (*Unity* erstellt und verwaltet z.B. *Shader*-Dateien selbst) ist in *C#* umgesetzt.

Unity wurde hauptsächlich wegen der leichten Zugänglichkeit, der umfassenden Online-Tutorials und der großzügigen Auswahl an frei zur Verfügung stehenden Modellen ausgewählt. Die Engine arbeitet mit Objekten, denen eine beliebige Anzahl an Komponenten, wie zum Beispiel Skripte und Physik-Attribute, hinzugefügt werden können. Außerdem können Skripte während der Ausführung des Programmes verändert werden, was eine schnelle Arbeitsweise beim Erstellen eines Prototypen ermöglicht. Eine Alternative dazu wäre die *Unreal*-Engine gewesen. Wie im vorigen Kapitel bereits erwähnt, war die Implementierung einer eigenen Engine aufgrund des hohen Zeitaufwandes keine Option.

Die Steuerung des Programmes ist simpel gehalten und funktioniert wie in gängigen *First-Person* Spielen, mit den Tasten *W*, *A*, *S* und *D* zur horizontalen Bewegung und der Maus zum Rotieren der Kamera. Ein Fadenkreuz zeigt zur Erleichterung des Aufhebens von Objekten den Mittelpunkt des Sichtfeldes und somit den Punkt, an dem mit der Umwelt interagiert werden kann. Objekte werden mit einem gedrückt gehaltenen Linksklick aufgenommen und können frei und ohne Einschränkung in Bezug auf das Gewicht bewegt werden. Als direktes Vorbild wurde dabei das Spiel *Amnesia: The Dark*

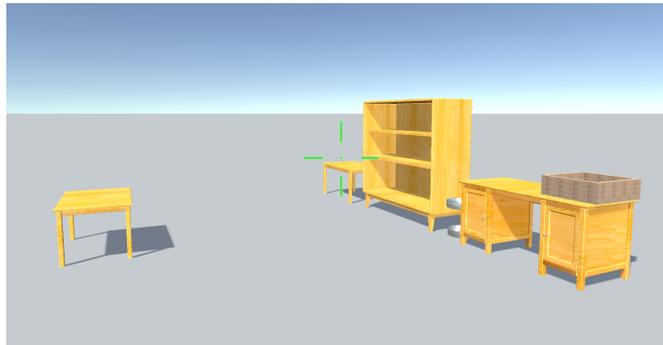


Abbildung 5.1: Weitwinkelaufnahme des *Thinkspaces*, mit den statischen Möbelstücken im Vordergrund. Im Hintergrund ist die endlose Weite zu sehen.

Descent [41] verwendet, da ich die Interaktion mit Objekten in diesem Spiel persönlich als sehr angenehm empfunden habe.

Ein *Rigidbody* sorgt bei den aufnehmbaren und statischen Objekten jeweils für das korrekte physikalische Verhalten. Ein weiterer Vorteil von *Unity* ist hierbei die Verwendung von *Prefabs*, die ähnlich wie vorgefertigte Schablonen als Grundmodell für instanziierte Objekte verwendet werden. Die im System gefundenen Dateien werden auf diese Weise in der dreidimensionalen Welt erstellt, das *Prefab* basiert dabei auf der jeweiligen Dateiendung.

5.2 Umgebung

Die Umgebung ist im Programm sehr schlicht gehalten, der Benutzer und alle Objekte befinden sich auf einer unendlich großen grauen Ebene, mit einem ebenso schlichten blauen Himmel darüber. Dadurch soll im Prototypen und auch bei der Benutzerstudie die Aufmerksamkeit ausschließlich auf die Interaktion mit den vorhandenen Objekten gelenkt werden, ohne den Benutzer mit der Erkundung des Gebietes abzulenken. Zu Beginn der Recherche war geplant, die Umgebung wie ein gemütliches Wohnzimmer oder ein Büro einzurichten, diese Idee wurde aber wieder fallen gelassen. Die ursprüngliche Idee dahinter war, dass der Benutzer sich in seiner Umgebung wohl und zu Hause fühlen sollte.

Stattdessen wurde schließlich die endlose Ebene umgesetzt, die zu Beginn der Entwicklung „*Thinkspace*“ getauft wurde. Sie soll einen ablenkungsfreien Raum darstellen, in dem mit voller Konzentration mit den eigenen Daten gearbeitet werden kann. Es soll einen Ort darstellen, an den man alleine und ungestört geht, um Ideen zu finden und umzusetzen.

Zu sehen ist der *Thinkspace* in Abb. 5.1 hinter den Möbelstücken. Besonders gut an dieser Umsetzung gefällt mir die Metapher der nicht vorhandenen Grenzen der eigenen Vorstellungskraft. Mit einem entsprechenden Editor, mit dem weitere Möbelstücke oder eventuelle andere Objekte beliebig hinzugefügt werden können, kann sich ein Benutzer endlos weit entfalten. So kann er beispielsweise projekt- oder themenbezogene Orte oder ganze „Städte“ anlegen, in denen die relevanten Dateien gelagert sind. Dabei kommt die



Abbildung 5.2: Nahaufnahme des virtuellen Arbeitsplatzes in Form eines Schreibtisches, hier mit sichtbarem *Collider* zum Auslösen des Verschiebevorgangs (in dunklem Grau). Links der Papierkorb zum Löschen, rechts auf dem Schreibtisch die Box zum Kopieren von Dateien.

schon in der Antike verwendete Methode des Gedächtnispalastes [19] in den Sinn, durch den man sich besser an Verschiedenes erinnern können soll. Ähnliche Versuche als Software [18][6] wurden bereits implementiert, jedoch mit reinem Bild- und Textmaterial und nicht mit der Repräsentation von Dateien und Projekten.

5.3 Statische Objekte

Die statischen Möbelstücke, welche die Ordner im Dateisystem repräsentieren, sind in den Abbildungen 5.2 und 5.1 zu sehen. Der Schrank wurde als Sinnbild für ein längerfristiges Archiv gewählt, der Tisch als ungeordnete, schnelle Ablegefläche und der Schreibtisch als Ablegefläche für Dateien, die gerade in Verwendung sind¹. Dem Schreibtisch könnte noch ein virtueller Monitor hinzugefügt werden, in dem Programmfenster abgebildet werden. Um die Möglichkeiten von 3D und vor allem Virtual Reality noch mehr auszuschöpfen, könnten die Programmfenster aber auch ähnlich wie beim Mixed Reality-Ansatz frei im Raum verteilt oder an Oberflächen gebunden werden.

Jedes dieser statischen Objekte verfügt über mindestens einen unsichtbaren *Collider*², die als „Container“ gekennzeichnet sind und Informationen über den jeweiligen gekoppelten Ordner beinhalten. Dieser ist in Abb. 5.2 als grauer Quader auf dem Schreibtisch dargestellt. In der eigentlichen Applikation ist dieser nicht zu sehen. Die zwei Fächer im unteren Teil des Schreibtisches, die einzelnen Regale des Schrankes sowie der Bereich auf dem Schrank und dem Tisch verfügen ebenfalls über solche Container.

5.4 Dynamische Objekte

Als dynamische Objekte werden alle Objekte gesehen, die vom Benutzer aufgehoben und nach Belieben bewegt werden können. Im Prototypen sind das alle Dateien, die beim Start des Programmes gefunden werden. Später in der Entwicklung werden noch

¹Vergleiche dazu die Typen von Daten in Kapitel 2: archiviert, arbeitsbezogen und ephemeral.

²*Collider* sind Elemente, die in *Unity* für die Kollisionsberechnung verwendet werden.

weitere Objekte hinzugefügt werden, wie zum Beispiel Schachteln zum leichteren Lagern von Dateien, oder diverse Zier- und Hilfsobjekte.

Wie bereits zu Beginn des Kapitels beschrieben, werden *Prefabs* verwendet, um das Aussehen und die Eigenschaften der Dateien zu bestimmen. Dazu wird im *Unity* Editor ein Objekt erstellt, mit den jeweiligen Skripts und Eigenschaften versehen und gespeichert. Diese *Prefabs* werden schließlich im Code des Programmes aus der *Asset Database* geladen und an gewünschtem Ort mit gewünschter Rotation instanziiert:

```
1 var object = (GameObject)AssetDatabase.LoadAssetAtPath("Assets/Prefabs/Image.prefab",
  ", typeof(GameObject));
2 // Add or alter information here
3 Instantiate(object, location, rotation);
```

Auf diese simple Weise können beliebig viele (bzw. bis die Grenzen des Rechners erreicht sind) Instanzen solcher Objekte erzeugt werden, was großen Datenmengen theoretisch entgegen kommt. Im Fall der Instanzierung eines Bildes wie im gezeigten Code wird in Zeile 2 noch eine Methode zum Ändern der Textur des Objektes aufgerufen. Dabei wird der mitgelieferte Dateipfad verwendet, um die Bilddatei als Textur zu laden, die dann auf das Prefab angewendet wird.

In der fortführenden Implementierung können noch weitere dynamische Objekte hinzugefügt werden, die keine Dateien repräsentieren. Ein Beispiel wären weitere Container bzw. dynamische Unterordner, Ziergegenstände oder Minigames. Diese haben zwar keine direkten Vorteile in Bezug auf den Dateimanager, bringen aber einen ästhetischen "Wohlfühlfaktor". Ähnliche Elemente wurden auch schon in Windows Vista in Form von Widgets eingeführt, und sind beispielsweise in *BumpTop* (in Form von eingerahmten Fotos an den Wänden) ebenfalls verfügbar. Solche rein kosmetischen Objekte würden in der Implementierung als einfache *GameObjects* dargestellt, ohne beigefügtes Skript.

5.5 Instanziierung und Programmstart

Über ein zu Beginn erscheinendes Menü kann ein Pfad aus dem Dateisystem ausgewählt werden. Dieser Schritt wurde eingeführt, um nicht etwa ein ganzes Laufwerk oder alle privaten Dateien auf einmal zu laden. Außerdem werden nur Dateien, die sich in der obersten Ebene oder in den bereits existierenden Containern befinden, verwendet. Beginnend beim angegebenen Pfad wird die Ordnerhierarchie nach Ordnern mit den gleichen Namen wie die vorhandenen Container durchsucht, und, wie zuvor bereits beschrieben, für jede gefundene Datei ein Objekt aus einem *Prefab* instanziiert. Das Instanzieren geschieht über das zentrale Skript der Applikation, den *FileManager*. Der *FileManager* trägt zusätzlich die Aufgabe, alle Dateien und Dateibewegungen zu verfolgen und zugänglich zu machen. Beim Start lädt dieser über den separaten *FileReader* die Ordnerstruktur und die enthaltenen Dateien und legt eine Liste derselben an. Zum Auslesen der Dateien wird die *Unity*-eigene *FileUtility* Klasse verwendet.

Je nachdem, ob bereits gespeicherte Positionsdaten aus einer vorigen Sitzung vorhanden sind, werden die Dateien mittig im entsprechenden Container oder mit der gespeicherten Position erstellt. Nachdem die Dateien als *GameObject* instanziiert wurden, werden auch diese im *FileManager* in einer Liste gespeichert, um die Positionen verfolgen und später wiederum speichern zu können.

5.5.1 Ordnerstruktur und -initiierung

Bevor die Dateien instanziiert werden, wird überprüft, ob die Anzahl und Namen der gefundenen Ordner mit den existierenden Containern übereinstimmt. Falls nicht, werden die entsprechenden Ordner erstellt und der daraus resultierende absolute Dateipfad³ in den Container-Objekten vermerkt. Dateien, die keinem existierenden Container zugeordnet werden können, also entweder manuell im Dateisystem hinzugefügt oder verschoben wurden, werden an einem zuvor definierten Standardplatz instanziiert. Im Fall des Prototypen ist das der Tisch, da die Dateien dort leicht aufgefächert werden können und weiter verteilt werden können.

Da der Prototyp separat zum existierenden Dateimanager handelt und die Dateien auch ohne das Programm wieder auffindbar sein sollen, wird die verwendete Container-Struktur auch im Dateisystem widergespiegelt. Da es praktisch unmöglich ist, die dreidimensionale Struktur mitsamt Koordinaten und Eigenschaften in einer für Menschen verständlichen Form im Dateisystem abzubilden, werden Attribute wie die Position im 3D-Raum außen vor gelassen. Sollte der Dateimanager als Hauptverwaltungssystem des Betriebssystems verwendet werden, fällt dieser Schritt glücklicherweise weg, und das Speichersystem kann auf die neue Speicherart optimiert werden.

Die statischen Objekte und Container werden zurzeit noch in eine Baumstruktur umgewandelt, mit den Namen der die Container enthaltenden Möbelstücke selbst als Elemente der ersten Ebene des Baumes. Die zweite Ebene spiegelt den jeweiligen Container wider. Als weitere Ebene könnten noch die in Kapitel 3 erwähnten Schachteln verwendet werden, und deren Beschriftung als Ordnername. In meinen Augen ist das aber optional, da die Einteilung in die *Collider* durchaus reichen sollte, solange diese nicht zu groß sind. Der Baum ist offensichtlich nicht sehr tief, was an der simplen Speicherstruktur des Prototypen liegt. Eine zu tiefe Struktur wäre auch kontraproduktiv beim Wiederfinden der Dateien im 2D-Dateimanager, da sie nicht mit diesem platziert wurden und somit das räumliche Gedächtnis nicht genutzt wird. Es entsteht somit eine logische Suche mit räumlichen Kriterien. In einer weiterführenden Implementierung können noch weitere Ebenen hinzugefügt werden, wie zum Beispiel Räume eines Hauses, Viertel einer Stadt, Bundesländer eines Staates, je nachdem, welche Metapher für die Unterteilung verwendet wird.

Die in Abb. 5.1 zu sehenden Container ergeben mit der jetzigen Implementierung die in Abb. 5.3 zu sehende Ordnerhierarchie. Sollten weitere Möbel hinzugefügt werden, werden der Ordner- und der Objektname mit einer inkrementierten Zahl versehen.

5.6 Funktionen des Dateimanagers

Folgend wird näher auf die Implementierung der in Kapitel 4 aufgelisteten unbedingt benötigten Funktionen des Dateimanagers eingegangen. Der benötigte Code ist in Skriptform an die jeweiligen Objekte angehängt und wird beim Eintreten eines als "File" gekennzeichneten Objektes in den entsprechenden "triggernden" *Collider* ausgeführt. *Collider*, die als "Trigger" gekennzeichnet sind, kollidieren in *Unity* nicht auf physische Weise, sondern sind standardmäßig unsichtbar und lösen in der vererbenden Klasse das *OnTriggerEnter*-Event aus. Die vererbende Klasse ist dabei die am meisten in *Unity*

³Der gesamte Dateipfad inklusive Laufwerksbuchstaben; nicht relativ zum Root-Verzeichnis.

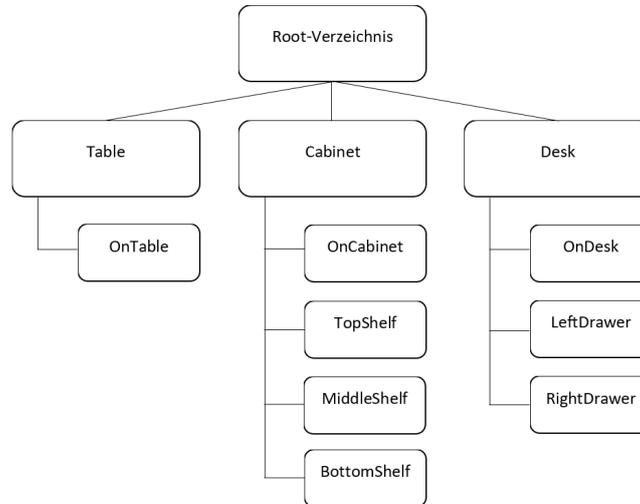


Abbildung 5.3: Die resultierende Ordnerstruktur nach dem Start der in Abb. 5.1 abgebildeten *Container*-Konstellation.

verwendete, die *MonoBehaviour* Klasse. Diese steuert zusätzlich den Update-Zyklus, und unter anderem den Zugriff auf die dem selben Objekt zugefügten Komponenten.

5.6.1 Verschieben und Umbenennen

Eines der Skripten, das *MoveScript*, wird zum Verschieben von Daten verwendet. Jedes Objekt, das eine Datei repräsentiert, hat die Informationen über den tatsächlichen Dateipfad und den Container, der mit diesem Dateipfad verknüpft ist. Wird eine Datei aufgehoben und aus dem Container entfernt, geschieht zunächst nichts; erst wenn das Objekt im nächsten Container den *Collider* berührt, wird die Datei im Hintergrund verschoben, und die Information in der Liste der *GameObjects* angepasst. Der Gedankengang dahinter ist, dass Daten nicht versehentlich von ihrem ursprünglichen Ort entfernt werden und folglich auch im Dateisystem des Betriebssystems nicht mehr an ihrem vermuteten Platz auftauchen. Bedingt durch die physikalisch realistisch gehaltene Art des Prototypen können Dateien beispielsweise hinter oder unter Möbelstücke rutschen und sind daraufhin schwer auffindbar.

Zum Verschieben werden in *Unity* der vollständige Herkunftspfad sowie der Zielpfad, beide inklusive richtigem Dateinamen, benötigt. Da der Prozess vom Container aus gesteuert wird, muss das Skript erst den Dateipfad aus dem File-Objekt auslesen. Durch das Komponentensystem von *Unity* kann über das durch den Trigger ausgelöste Kollisionsevent sehr einfach auf das Objekt und somit den Pfad zugegriffen werden. Der Verschiebebefehl der *FileUtil*-Klasse kann auch zum Umbenennen oder Ändern der Dateieindung verwendet werden, indem beim „Verschieben“ der Ordnerpfad zwar gleich bleibt, der Dateiname jedoch geändert wurde.

5.6.2 Kopieren

Der Kopiervorgang wird auf die selbe Art ausgelöst wie das Verschieben, mit der Ausnahme, dass der *Collider* keinen Ordnerpfad gespeichert hat. Beide Dateien befinden sich demnach in dem Container, in dem sich auch der Kopier-*Collider* befindet. Falls dieser sich in keinem Container befindet, sind die Dateien in dem Ordner, dessen Container der Ursprungsdatei zuletzt zugeteilt war. Im Prototypen ist der Kopier-*Collider* in der in Abb. 5.2 zu sehenden Box rechts auf dem Schreibtisch. Sobald eine Datei dort hineingelegt wird, wird

1. das *GameObject* kopiert und mit den Dateiinformationen versehen,
2. die Liste von allen Dateien und *GameObjects* aktualisiert und
3. die Datei im Dateisystem kopiert.

Im Fall der Abbildung befinden sich dann beide Dateien innerhalb des Ordners und des gleichnamigen *Colliders OnTable*.

Da sich im selben Ordner nicht zweimal die selbe Datei mit selbem Namen befinden darf, wird der Dateiname wie im normalen Dateisystem um eine Zahl erweitert, die mit jedem Kopiervorgang inkrementiert wird. Will man eine Datei mehrmals kopieren, muss diese zuerst aus dem *Collider* entfernt werden, um beim nächsten Wiedereintritt erneut das Skript auszuführen.

5.6.3 Löschen

Der Löschvorgang läuft sehr ähnlich ab wie der Kopiervorgang. Im Papierkorb befindet sich ebenfalls ein *Trigger-Collider*, nur dass diesem ein Skript zum Löschen von *GameObjects* und den zugehörigen Dateien angehängt ist. Zusätzlich wird das entsprechende *GameObject* aus der Liste entfernt.

Noch gibt es keine Funktion wie beim normalen Papierkorb die die gelöschten Dateien erst komprimiert und archiviert, bevor sie endgültig aus dem System gelöscht werden. Der Prototyp handelt wie die Kommandozeile und löscht Dateien sofort, was mit der implementierten Physik durchaus zu Problemen führen kann. Sollte etwa ein Stapel Dokumente umkippen und einige Dateien versehentlich in den Papierkorb gelangen, könnten wichtige Dokumente verloren gehen. Eine Funktion zum rückgängig Machen der letzten Aktion ist hier das Mindeste, was zum Schutz der Daten implementiert werden muss.

Kapitel 6

Benutzerstudie

Im Zuge des begleitenden Projektes wurde eine Benutzerstudie durchgeführt, um das entwickelte Programm auf Stärken und Schwächen zu testen, und um eine allgemeine Meinung zu dem Thema zu erfahren. Der Ablauf der Studie ist dabei grob an die 2001 von Andy Cockburn und Bruce McKenzie veröffentlichte [11] Forschung und begleitende Benutzerstudie angelehnt, jedoch ohne zweidimensionalen Vergleich. Eine Aufgabe der Studie war das Testen des Prototyps, um eventuelle Schwächen zu finden und bearbeiten zu können; die andere Aufgabe war herauszufinden, inwiefern der verwendete, physikalisch realistisch gehaltene Ansatz als Werkzeug zum Sortieren von Dateien geeignet ist.

6.1 Aufgabe und Vorbereitung

Die Teilnehmer wurden gebeten, in einer dreidimensionalen Umgebung 30 Fotos in einen Schrank mit drei Regalen einzusortieren. Die Fotos liegen zu Beginn auf einem Tisch bereit, direkt gegenüber des Schrankes. Beim Starten des Durchlaufes steht der Benutzer rechts der Möbel, mit Blick auf die Objekte gerichtet. So wird die Aufmerksamkeit sofort auf diese gelenkt. Vor der Studie standen zwei Schränke und zwei Tische zum Einsortieren zu Verfügung, allerdings wurden diese aufgrund des daraus resultierenden zu hohen Platzangebotes bei der gegebenen Menge an Fotos auf einen Schrank reduziert. Das geringe Platzangebot zwingt den Benutzer dazu, die vorhandenen Regale in Subregionen zu unterteilen und die bereits einsortierten Fotos zu reorganisieren, was einem echten Archivierungsverlauf näher kommt.

Die im Programm dargestellten Fotos entsprechen in ihrer Größe und ihrem Aussehen einem größeren Standard Fotoformat mit Glanzdruck, es wurde jedoch kein physikalisch korrektes Verhalten des Materials implementiert, d.h. die Fotos lassen sich nicht biegen. Die Fotos sind starre Kacheln, auch das Gewicht ist größer als in der Realität. Es werden wegen des geringen Mehrwerts auch keine Luftströme berechnet; wenn ein Foto losgelassen wird, fällt es also wie in einem Vakuum gerade nach unten. Zur leichteren Handhabung ist das Bild auf beiden Seiten der Kachel zu sehen. Dem Fortlauf der Studie steht die minimalistisch implementierte Physik aber nicht im Weg, da das wichtigste das Bild selbst und dessen Position sind. In Abb. 6.1 ist das Programm während eines Durchlaufes zu sehen, mit mehreren bereits einsortierten Fotos und einem hochgehaltenen im Vordergrund.



Abbildung 6.1: Ein Screenshot der für die Studie verwendeten Umgebung. Im Hintergrund der Schrank mit bereits kategorisch einsortierten, im Vordergrund ein hochgehaltenes Foto hinter dem zur Steuerungshilfe angezeigten Fadenkreuz.

Es wurden speziell Fotos für diese Studie gewählt, weil die wichtigste Information für ein sinnvolles Kategorisieren und Einsortieren das Bildmaterial selbst ist und keine weiteren (Meta-)Informationen angezeigt und gelernt werden müssen. Des Weiteren sind die Kategorien, nach denen sortiert wird, für den Verfasser der Studie bei der Analyse leicht zu erkennen. Das Problem mit Dokumenten und anderen Daten ist, dass diese den Teilnehmern entweder bereits bekannt sein müssen oder sie sich die Informationen erst anlernen müssen. Der erste Fall ist für eine Studie aufgrund der schlechten Vergleichbarkeit, der zweite aufgrund der langen Dauer und damit sinkenden Motivation der Teilnehmer nicht geeignet.

6.2 Ablauf

Zu Beginn wurde jedem Teilnehmer die Steuerung, wie beschrieben in Kapitel 5, erklärt. Nach einer kurzen Eingewöhnungsphase, in der sich die Teilnehmer frei bewegen durften, wurde ihre Aufmerksamkeit auf den Stapel Fotos gelenkt und die eigentliche Aufgabe erklärt und begonnen. Den Probanden wurde bereits zu Beginn erklärt, dass sie die Fotos nach dem Einsortieren wiederfinden sollten. Alle Teilnehmer wurden während des Bewältigens der Aufgabe genau beobachtet und Auffälligkeiten in Bezug auf Umgang mit dem Programm und in Bezug auf das Organisationsverhalten notiert. Es stand jedem Teilnehmer jederzeit frei, Fragen zu stellen und die Fotos beliebig oft neu zu sortieren. Die einzige Einschränkung war, dass die Fotos nicht auf dem Tisch vorsortiert werden durften.

Nach dem erfolgreichen Einsortieren der 30 Fotos wurde allen Teilnehmern noch einmal die Chance gegeben, die Sortierung zu ändern, was von einigen in Anspruch genommen wurde. Danach wurden den Probanden nacheinander fünf zufällig ausgewählte Fotos gezeigt, und sie wurden gebeten, diese in den Regalen zu finden und deren Position zu zeigen. Die Fotos wurden im Regelfall schnell gefunden, wobei die größten Hindernisse der sehr flache Blickwinkel auf die im obersten Regal liegenden Fotos sowie der starke Schattenwurf des von oben kommenden Lichtes darstellten.

Nach diesem letzten praktischen Schritt wurden alle Teilnehmer gebeten, einen Fragebogen auszufüllen. Neben dem Liefern demographischer Informationen sollte dieser dazu dienen, das zweidimensionale Organisationsverhalten der Teilnehmer zu bestimmen und diese zumindest grob in eine der in Abschnitt 2.4 genannten Kategorien „Piler“, „Filer“ und „Structurer“ einzuteilen. Der gesamte Fragebogen ist in Anhang A.3 zu finden. Eine bessere Einschätzung wäre mit Hilfe eines Snapshots des Dateisystems möglich, um die genaue Anzahl der Ordner und der enthaltenen Dateien, sowie der Ordnertiefe und -breite, festzulegen.

6.3 Analyse der gewonnenen Daten

6.3.1 Demografische Daten

Die Studie wurde mit zwölf Personen durchgeführt, sieben weiblich, fünf männlich, und dauerte jeweils zwischen zehn und zwanzig Minuten. Die Dauer hing hauptsächlich davon ab, ob die Teilnehmer mit der Steuerung vertraut waren, bzw. wie lange es dauerte, sich mit der Steuerung vertraut zu machen. Sieben der Teilnehmer sind Studenten, wovon sechs ein Studium mit Themengebiet Informatik absolvieren und viel Zeit am Computer verbringen. Einer der nicht studierenden Teilnehmer ist als Quality Engineer erwerbstätig, und somit auch versiert um Umgang mit Daten. Acht Teilnehmer sind zwischen 21 und 26 Jahre alt, zwei davon unter 20, und zwei über 40. Wie im Rest der Masterthesis werden in der Auswertung Teilnehmer jedes Geschlechts teilweise geschlechtsneutral in der männlichen Form angesprochen.

6.3.2 Auswertung des Fragebogens

Die für die Organisationstypen relevanten Antworten sind in Abb. 6.2 zu sehen. Einige der Befragten hatten Schwierigkeiten damit, sich selbst, besonders in Bezug auf die ersten beiden Fragen, objektiv einzuschätzen. Wie an den Antworten auf Frage 2 zu sehen ist, war die persönliche Einschätzung sehr unterschiedlich, wenngleich sich diese Unterschiede in der späteren Organisation der Fotos nicht so deutlich widerspiegeln. Dadurch stellt sich die Frage, ob sich die Organisation von Fotos überhaupt mit der Organisation von Dateien gleichstellen lässt, oder ob hier wirklich nur die persönliche Einschätzung so stark differiert. Es müsste hier wiederum ein *Snapshot* des privaten Dateisystems erstellt und analysiert oder eine Vergleichsstudie gestartet werden, um diese Frage beantworten zu können.

Die Ergebnisse der Befragung deuten darauf hin, dass die Hälfte der Teilnehmer Piler, die andere Filer sind. Neue Dateien werden dennoch von den meisten zeitnah in die bestehende Ordnerstruktur einsortiert, und die Ordnerstruktur ist nach Selbsteinschätzung nicht sehr breit und hat viele Unterordner. Mehr als die Hälfte der Teilnehmer hat aber eine Menge Dateien auf dem Desktop, was ein Indiz für Piling sein kann. Ein *Snapshot* des Systems würde hier Klarheit verschaffen. Keiner der Teilnehmer lässt sich in die Kategorie Structurer einordnen, da niemand angab, leere Ordner bereits im Vorhinein anzulegen. Nur vier der Teilnehmer lagern viele Dateien auf dem Desktop ab, zwei davon besitzen themenübergreifende „Dumping“-Ordner. Diese lassen sich demnach in die Piler-Kategorie einordnen. Einer der Teilnehmer, der zu dieser Kategorie gezählt werden

kann, war auch der Einzige, der bevorzugt die textuelle Suchfunktion zum Finden von Dateien verwendet. Damit bestätigt sich die in [4] und [25] beschriebene Erkenntnis, dass der Großteil der Benutzer von Computern und realen Archiven die ortsbezogene Suche bevorzugt. Dieser Teilnehmer brauchte mitunter auch am längsten, um die Fotos im zweiten Teil des praktischen Teils wiederzufinden. Am schnellsten beim Wiederfinden waren wenig überraschend jene Teilnehmer, die sich nach dem Einsortieren noch die Zeit nahmen, die Fotos umzusortieren.

In Bezug auf die Alltagstauglichkeit und die zukünftige Entwicklung von 3D-Dateimanagern waren die Meinungen sehr unterschiedlich. Bis auf eine Teilnehmerin gaben alle an, Spaß an den Aufgaben, und mit dem Programm im Allgemeinen, gehabt zu haben. Trotzdem sortieren die meisten in der gewohnten zweidimensionalen Umgebung besser und denken auch nicht, dass in Zukunft hauptsächlich 3D-Dateimanager verwendet werden. Es besteht hier noch ein großer Bedarf an weiterer Forschung und Entwicklung, um das volle Potential der Materie auszuschöpfen. Es ist auch unwahrscheinlich, dass dreidimensionale Dateimanager in Zukunft überall verwendet werden (zumindest anfänglich), stattdessen muss die Verwendung sinnvoll auf gewisse Teilbereiche zugeschnitten werden.

6.3.3 Auswertung des praktischen Teils

Die Durchführung des praktischen Teiles verlief ohne Probleme, obwohl keiner der Studienteilnehmer zuvor einen dreidimensionalen Dateimanager oder Desktop verwendet hatte, und nur die Hälfte der Teilnehmer mit der *WASD*- und Maussteuerung vertraut waren. Diejenigen, die die Steuerung noch nicht verwendet hatten, machten jedoch schnell Fortschritte und wurden innerhalb der ersten 10 Minuten spürbar schneller in ihren Bewegungen. Die zwei Teilnehmer über 40 hatten die größten Einstiegsprobleme und waren ungeduldig mit der Steuerung, doch auch sie gewöhnten sich im Lauf der Studie daran.

Im Lauf der Studie wurde aufgrund des Feedbacks der ersten Teilnehmer noch eine Zoomfunktion eingebaut, da die Fotos teilweise eine zu hohe Auflösung für die gegebene Fläche aufwiesen und das Motiv durch den zu großen Detailreichtum nicht klar erkennbar war. Außerdem wurde die Laufgeschwindigkeit mehrmals verringert, da die genaue Platzierung eines Objektes sich sonst als schwierig erwies. Bei der ersten Teilnehmerin standen noch zwei Schränke zum Einsortieren zur Verfügung, was den zweiten Teil der Studie, in dem die Fotos gefunden werden müssen, in eine Art Memory-Suchspiel verwandelte. Die Fotos waren nicht in Kategorien zusammengepackt, sondern nebeneinander aufgereiht. Als Ergebnis mussten die Fotos nicht kategorisiert oder nach einer anderen Strategie sortiert werden, da das simple Durchgehen der Fotoreihen sich als ebenso erfolgreich erwies.

Die Motive der Fotos selbst lassen sich grob in fünf Kategorien mit teilweise vorhandenen Unterkategorien einteilen:

- Tiere – Hunde, Katzen und Nutztiere,
- Nahaufnahmen von Personen – Unbekannte Personen und bekannte Politiker,
- Personengruppen – Teilweise mit Tieren und Landschaften,
- Panoramafotos – Landschaftsfotos und Skylines,
- Anderes – Ein einzelnes Foto, das keiner anderen Kategorie zuzuweisen ist.

ORGANISATION:

1. Ich bin generell eine organisierte Person.



2. Meine persönlichen Daten sind gut strukturiert.



3. Ich habe Ordner, in die ich alle möglichen Dateien ohne gemeinsames Thema verschiebe.



4. Ich erzeuge Ordner für bestimmte Dateikategorien bereits bevor ich Dateien besitze, die zu dieser Kategorie gehören.



5. Wenn ich neue Dateien erstelle oder erhalte, sortiere ich sie sofort in meine Ordnerstruktur ein.



6. Ich habe nicht viele Unterordner in meiner Ordnerstruktur.



7. Auf meinem Desktop liegen viele Dateien.



8. Wenn ich eine Datei suche, finde ich sie normalerweise sehr schnell.



3D DATEIMANAGER

9. Ich denke, die Zukunft von Dateimanagern und Desktops liegt in 3D.



STUDIE FEEDBACK

10. Es hat mir Spaß gemacht, Dateien in 3D zu sortieren.



11. In 3D würde ich meine Dateien besser sortieren als in 2D.



12. Ich sortiere Dateien lieber in 3D als in 2D.



13. Es war leicht, mich an den Ort der gesuchten Fotos zu erinnern.



■ Lehne voll ab ■ Lehne ab ■ Neutral/Keine Angabe ■ Stimme zu ■ Stimme voll zu

Abbildung 6.2: Die für die Analyse relevanten Fragen der Benutzerbefragung. Der Befragungsmodus war eine fünfteilige Likert-Skala. Die restlichen Fragen sind im Appendix ersichtlich und stellen Feedback für die Implementierung des Testprogrammes dar.

Eine Besonderheit der gewählten Fotos ist, dass sich die Kategorien teilweise vermischen. So gibt es mehrere Personen, die mit Tieren oder mit Landschaften im Hintergrund abgebildet sind. Der Zweck dahinter war festzustellen, ob die Studienteilnehmer Teile des Motivs gegenüber anderen priorisieren, oder neue (Sub-) Kategorien oder Ablegeplätze kreieren. Außerdem sollten sie angeregt werden, ihre Fotos beim Auftauchen einer neuen (Unter-)Kategorie umzusortieren. Eine der Teilnehmerinnen durchschaute diese Taktik und ordnete die Fotos beinahe exakt nach den oben aufgelisteten Kategorien an. So legte sie etwa Fotos von Personen links ins Regal, Fotos von Tieren rechts, und Fotos von Personen mit Tieren in die Mitte.

Die anderen Teilnehmer, die nach Kategorien ordneten, priorisierten bei den Bildern mit Tieren und Personen häufiger die Tiere, und bei den Bildern mit Personen und Landschaften meistens die Personen. Da die Anzahl an Katzen- und Hundefotos die der Nutztiere überstieg, reorganisierten einige ihre Fotos während und nach dem initialen Sortieren und legten neue Kategorien für diese zwei Rassen an. Eine unerwartete Besonderheit ist außerdem, dass in diesem Fall bei Fotos mit Katzen und Hunden immer die Katze als Kategorie gewählt wurde, und auch bei Personen mit Katzen oft die Katze priorisiert wurde.

Bis auf diese Besonderheit der Priorisierung verlief die Sortierung sehr ähnlich wie in der Studie von Cockburn und McKenzie, bei der die Thumbnails von Webseiten ebenfalls in thematisch zusammengehörige Cluster gelegt wurden. Da Cockburn und McKenzie aber bis zu 99 Thumbnails sortieren ließen, wurden diese öfter umsortiert und waren aufgrund der geringen verfügbaren Fläche schwer zu unterscheiden.

Das einzelne Foto der Kategorie „Anderes“ – ein Portrait einer Cartoon-Figur – wurde nicht, wie erwartet, bei den anderen Personen platziert, sondern initial fast immer als neue Kategorie angelegt. Nachdem die Teilnehmer erkannten, dass keine weiteren Fotos dieser Kategorie zuordenbar sind, wurde dieses Bild teilweise an den Rand verschoben, um Platz für größere Kategorien zu schaffen. Eine Teilnehmerin erschuf für dieses und ein Foto eines Menschen mit Pferdemaske die Kategorie „Blödsinn“. Am Ende des Sortierens wurde die Kategorie aufgrund der geringen Größe wieder aufgelöst, das Pferd zu den Tieren und die Comicfigur kategorielos am Rand des Regals platziert.

Fünf der Piler aus der Befragung ließen sich auch eindeutig in ihrer dreidimensionalen Organisation erkennen. Anstelle von Kategorien legten diese alle Fotos gut sichtbar, aber unsortiert, in einer Reihe in den Regalen auf. Nachdem der Platz dafür zu eng wurde, wurde hinter der relativ weit vorn platzierten Reihe eine weitere, immer noch relativ gut sichtbare Reihe begonnen. Eine andere Teilnehmerin häufte die Fotos wahllos in den Regalen auf, obwohl sie im Fragebogen angegeben hatte, dass ihre Dateien gut organisiert seien und auch ihr Desktop aufgeräumt sei. Zwei dieser Teilnehmer brauchten zum Finden der Fotos mitunter am längsten. Andere Piler wiederum legten Bestzeiten hin; Diese legten Wert darauf, die Fotos unsortiert, aber gut sichtbar, in den Regalen aufzulegen. Beim Wiederfinden verließen sie sich ausschließlich auf das räumliche Gedächtnis, und fanden das richtige Foto meist sofort. Dadurch, dass sie nicht erst die richtige Kategorie, und danach das richtige Foto suchten, sparten sie Zeit. Falls das richtige Foto allerdings nicht sofort gefunden wurde, brauchten sie erheblich länger, da nun alle Fotos durchgesehen werden mussten. Einer der Teilnehmer, die laut Fragebogen zu den Pйлern gehören, sortierte die Fotos wie die restlichen Teilnehmer nach Kategorie und teilte diese auch wieder in Subkategorien, um genau zu sein in die Kategorien „Katzen“

Tabelle 6.1: Die gemessenen Zeiten im zweiten Schritt der Studie.

<i>Messreihe</i>	<i>Durchschnittliche Zeit</i>	<i>Max. Zeit</i>	<i>Min. Zeit</i>
Gesamt	6,67	37	0,85
Filer (Fragebogen)	7,7	31	0,85
Filer (Praktisch)	5,9	15	0,85
Piler (Fragebogen)	5,6	37	1,2
Piler (Praktisch)	7,5	37	1,2

und „Andere Tiere“. Obwohl dieser angab, „so gut wie alle seiner Dateien“ auf dem Desktop zu lagern, bewies er sich im praktischen Teil als organisiert und befand sich beim Wiederfinden im Durchschnitt der gemessenen Zeiten.

6.3.4 Zeitmessungen

Während des zweiten Schrittes der praktischen Studie wurde die Zeit gemessen, die die Benutzer brauchten, um ein zufällig ausgewähltes Bild zu finden. Der Timer wurde dabei gestartet, sobald der Teilnehmer das Bild gesehen und sich wieder auf den Bildschirm konzentriert hatte. In Tabelle 6.1 sind jeweils die durchschnittliche, längste und kürzeste Zeit für alle Kandidaten und jeweils die selbst eingeschätzten und praktisch erkannten Filer und Piler gelistet.

Es springt sofort ins Auge, dass die (praktischen) Filer im Durchschnitt um einiges schneller waren als die Piler. Die Piler mussten das gesamte Set an Bildern durchgehen und teilweise auch überlappende Bilder verschieben, um das gesuchte Foto zu finden. Es macht also auch Sinn, in einer dreidimensionalen Umgebung, die Piling ermöglicht, Daten ordentlich zu sortieren und dafür Kategorien anzulegen. Wie wir aus Kapitel 2 wissen, sind Piles auch hauptsächlich zum schnellen Arbeiten, nicht zum Lagern und Wiederfinden von Dateien gedacht.

Einige der Studienteilnehmer merkten an, sie hätten gerne eine Funktion zum Beschriften oder anderweitigen Kennzeichnen der selbst eingeteilten Regionen in den Regalen. Dadurch wären sie laut eigener Einschätzung bei der Suche schneller gewesen, und es kann auch nach längerer Zeit die angelegte Kategorie leicht ins Gedächtnis gerufen werden.

6.4 Zusammenfassung

Aus diesen Daten kann man herauslesen, dass Filer und Piler auch in einer dreidimensionalen Umgebung ihr Verhaltensmuster beibehalten. Demnach sollte man bei der Entwicklung eines solchen Systems an beide Gruppen denken und ihnen entsprechend angepasste Hilfswerkzeuge zur Verfügung stellen. Leider konnte in der Umfrage kein Structurer gefunden werden, diese Gruppe war auch in den initialen Studien [4, 25] die kleinste Gruppe. Für diese Gruppe wäre ein Editor, mit dem beliebig viele Schränke und andere Aufbewahrungsgegenstände hinzugefügt werden können, von größtem Nut-

zen. Doch auch für die anderen beiden Gruppen ist diese Funktion bei der Archivierung größerer Datenmengen unabdingbar.

Die erfassten Daten stimmen größtenteils mit den Erwartungen überein. Wie schon in den vorhergehenden Studien war die Gruppe der Filer die größte, und diese teilten die Dateien im Lauf des Einsortierens in laufend erschaffene Kategorien ein. Diese Vorgehensweise ist sehr ähnlich zu der Art, wie die meisten [11] Menschen ihre Dateien im Dateimanager und auch im echten Leben sortieren.

Auch wenn die Piler im Durchschnitt merklich langsamer waren, bedeutet das nicht, dass Piling schlechter ist als Filing. Da die Aufgabenstellung keine umfassendere Interaktion mit den Dateien verlangte, sondern nur das für Piler eher seltene Einsortieren, ist es nicht verwunderlich, dass die Piler schlechter abschnitten. In einer echten Arbeitsumgebung, in der Dateien häufig aufgenommen und wieder abgelegt werden, bietet Piling eindeutige Vorteile. Es ist meiner Meinung nach wichtig, dass auch dieses Szenario noch ausgiebig getestet wird, um für alle potentiellen Benutzer eine befriedigende Benutzererfahrung zu schaffen.

6.5 Resultat

Die gewonnenen Daten zeigen, dass sich bei einem Sprung von 2D auf 3D in Bezug auf die Organisationsgewohnheiten nicht viel verändern würde. Die Eignung von dreidimensionalen Dateimanagern als Ganzes für den Alltagsgebrauch bestätigt sich dadurch aber noch nicht. Zumindest kann aber behauptet werden, dass dieser Teilaspekt dem Umstieg nicht im Wege stehen wird. Da das Thema ein sehr umfassendes ist, besteht hier noch enormer Forschungsbedarf; zusätzlich kommt ein gewisser Glücksfaktor hinzu, auf welche Technologie letztendlich gesetzt wird. Auf die weitere Eignung und das Gesamtergebnis wird im folgenden Kapitel näher eingegangen.

Kapitel 7

Diskussion

In diesem Kapitel will ich die vorigen Kapitel und die Erkenntnisse der Benutzerstudie aus einem subjektiveren Blickwinkel zusammenfassen und diskutieren. Außerdem wird der Wert des Themas für die Zukunft, und vor allem die Vision dahinter, behandelt.

Es ist schwer, den Erfolg oder Misserfolg des im Lauf der Arbeit implementierten Projektes zu quantifizieren und evaluieren. Berücksichtigt man den im Vergleich zum Endprodukt sehr frühen Prototyp-Status und die Größe, die ein solches Projekt annehmen kann, ist es vielleicht sogar noch zu früh, um überhaupt darüber urteilen zu können. Trotzdem werde ich versuchen, ein paar Lehren aus der Thematik und dem Vorgang zu ziehen.

7.1 Benutzerstudie

Jedes Projekt kann auf verschiedene Arten evaluiert werden, etwa in Bezug auf Effizienz, Benutzerzufriedenheit oder Funktionsvielfalt. In Bezug auf die Benutzerstudie sind hauptsächlich die Effizienz und die Organisation von Interesse. Die Effizienz ist hauptsächlich deswegen interessant, weil sie einer der wohl wichtigsten Faktoren für den Erfolg oder Misserfolg eines Werkzeugs für derartige Aufgaben ist. Obwohl keine Vergleichsstudie in 2D abgehalten wurde, gehe ich davon aus, dass die Teilnehmer wie in [11] schneller gewesen wären, teilweise bedingt durch das gewohnte Umfeld, teilweise durch die schnellere Navigation im durch die zwei Dimensionen beschränkten Raum. Da seit ca. 30 Jahren praktisch die selbe Technologie verwendet wird, würde ein Umstieg von 2D auf 3D vor allem denen, die schon lange mit Computern arbeiten, sehr schwer fallen.

Was bei der referenzierten gleich wie bei der selbst durchgeführten Benutzerstudie auffällt, ist, dass alle Teilnehmer jeweils Spaß an der Durchführung der Aufgaben hatten, und die dreidimensionale Variante persönlich als besser bewertet wurde. Sobald also das „Gesamtpaket“ ausgereift genug ist, sollte sich 3D gegenüber 2D durchsetzen können.

Ein weiterer auffälliger Punkt ist, dass die meisten glauben, sie würden ihre Dateien in 3D schlechter sortieren; auch sind die Meinungen bezüglich der Frage, ob 3D-Dateimanager die 2D-Dateimanager ablösen werden, eher neutral bis negativ. Es ist verständlicherweise schwer, diese Frage zu beantworten, ohne sich mit dem Thema auseinandergesetzt zu haben. Auch hat keiner der Teilnehmer zuvor einen 3D-Dateimanager

oder Desktop verwendet. Meine Meinung bezüglich des Themas ist, dass sich 3D zumindest in einigen, hauptsächlich privaten, Bereichen durchsetzen kann. Die weitaus interaktiveren und immersiveren Gestaltungsmöglichkeiten bieten großes Potential bei der Implementierung einer übergreifenden Lösung, die sich nicht nur auf Dateien konzentriert. So wie Dateimanagement nur Teil des Betriebssystems, und die Organisation von Dateien nur ein Teil des dreidimensionalen Ansatzes ist, so sollte diese Funktion auch als Teil eines größeren Ganzen sein.

7.1.1 Effizienz

Bezogen auf die Effizienz wird ein dreidimensionaler Dateimanager möglicherweise nie, und wenn, dann erst nach einer längeren Eingewöhnungsphase, die Effizienz eines zweidimensionalen Dateimanagers erreichen. Doch auch ein Normalverbraucher mit einem zweidimensionalen Dateimanager wird nie so effizient sein wie ein Experte auf der Kommandozeile. Trotzdem hat sich die Fenstervariante aus verständlichen Gründen gegenüber der Kommandozeile durchgesetzt. Sie ist verständlicher, spricht wie die dreidimensionale Variante das räumliche Erinnerungsvermögen an und ist durch die Verwendung von Metaphern zugänglicher.

Der größte Nachteil in Bezug auf die Effizienz ist in der dreidimensionalen Variante das Bedürfnis nach Fortbewegung auf drei Achsen. Während der Standard-Desktop mit der Maus im Bruchteil einer Sekunde überquert werden kann, dauert es in einer *First-Person* Applikation wesentlich länger, von einem Archiv zum nächsten zu gelangen. Zum einen ist in einer dreidimensionalen Umgebung die begehbare Fläche größer, zum anderen die Bewegungsgeschwindigkeit zum Eindämmen von Schwindel und Orientierungslosigkeit langsamer. Der Schwindeffekt kann in Virtual Reality durch eine unbalancierte Steuerung noch verstärkt werden.

Dazu kommt, dass Funktionen wie das Auswählen, Zusammenfassen oder gemeinsame Bearbeiten in 2D mit den üblichen Werkzeugen sehr viel schneller gehen. Während die Merkfähigkeit zwar besser eingesetzt werden kann [6], sind auch logische, namens- und eigenschaftsspezifische Suchfunktionen effizienter und liefern gleichzeitig bessere Ergebnisse.

7.1.2 Metaphern

Warum also überhaupt einen dreidimensionalen Dateimanager verwenden, wo er doch mit den heutigen Interaktionsmethoden nie die Effizienz eines zweidimensionalen Gegenstücks erreichen kann? Eine der Antworten ist meiner Meinung nach die leichtere Zugänglichkeit für Personen, die Probleme mit der abstrakten Struktur eines Dateisystems und der abstrakten Funktionsweise eines Computers an sich haben. Schon seit den 1980er Jahren werden GUIs (Graphical User Interfaces) mit visuellen Hilfsmitteln wie z.B. repräsentativen Icons verwendet, um die Arbeit mit Computern einstiegfreundlicher

zu gestalten [51], man vergleiche dazu Abschnitt 3.2.3 und die Auflistung der angegebenen Quelle.

Ein weiterer großer Vorteil einer dreidimensionalen Applikation ist, dass ein Teil der Umgewöhnung von der Arbeit mit realen Gegenständen zu der Arbeit mit virtuellen Objekten wegfällt beziehungsweise erleichtert wird. Bei der Verwendung von Computern besteht zu Beginn ein gewisser Erklärungs- und Einarbeitungsbedarf, um sich mit den Konzepten und Vorgängen des User Interfaces vertraut zu machen.

Es sollte dagegen leichter fallen, ein Objekt, das real aussieht und die Datei zusätzlich durch eine logische Metapher darstellt zu verstehen und zuordnen zu können. In zweidimensionalen Dateimanagern werden diese zwar auch verwendet, der Effekt wird aber durch ein dreidimensionales Modell, das eventuell noch Auskünfte über den Inhalt der Datei gibt, noch verstärkt.

7.1.3 Orientierung

Nach wie vor haben viele Personen auch Probleme mit der Navigation und Orientierung im Dateisystem. Wer einen Computer nur sporadisch verwendet, befasst sich weniger damit, ein System der Ordnung in seine Dateien zu bringen, und hat weniger Übung darin, sich im System zurechtzufinden. Ein dreidimensionaler Dateimanager mit starkem Realitätsbezug und sinnvollen, leicht verständlichen Metaphern könnte diesen Personen unter die Arme greifen. Dadurch, dass Computer für diese Zielgruppe zugänglicher wären, wäre auch die Überwindung geringer, sich noch mit dem Thema zu befassen und davon zu profitieren. Wie bereits in Kapitel 2 beschrieben, verbindet das menschliche Gehirn Erinnerungen gerne mit Orten. Das ist in einem zweidimensionalen Browser auch gegeben, der Effekt kann aber mit der dritten Dimension und visuellen Landmarken wie den im Prototypen verwendeten Möbeln und rein kosmetischen, optionalen Gegenständen, noch verstärkt werden.

Kapitel 8

Schlussbemerkungen

Diese Masterthesis soll dazu anregen, über die Verwendung von 3D bei Dateimanagern nachzudenken. Wichtig ist dabei nicht ausschließlich die Einbringung der dritten Achse an sich, wie es bei den bisherigen Versuchen oft der Fall war; hingegen sollen die Möglichkeiten, die sich in Bezug auf Darstellung der Dateien an sich, sowie die vielseitigen neuen Interaktionsmöglichkeiten mit dem Computer und der virtuellen Umgebung, genau evaluiert werden. Ich denke, dass in den bisherigen Versuchen oft zu wenig Gedanken darin investiert wurden, welche völlig neuen Konzepte durch eine weitere Dimension ermöglicht werden; stattdessen wurde viel Wert darauf gelegt, das Element „3D“ so effektiv wie möglich in Szene zu setzen. Ich denke dieser Schritt sollte erst mit einer stark weiter entwickelten Technologie, zum Beispiel „echtem“ 3D, gemacht werden. Gemeint sind damit Bildschirme oder andere Werkzeuge, die keine Hilfsmittel wie Brillen brauchen, um dreidimensionale Inhalte anzuzeigen.

In Rückblick auf das Projekt war ich überrascht, wie viel Arbeit und vor allem vorangehende Forschung in einen Dateimanager gesteckt werden muss. Während diese Thesis nur einen kleinen Teil des großen Ganzen behandelt, und auch auf diesen bezogen noch weitere Studien fällig sind, gibt es ein großes Spektrum an weiten Bereichen, die bedacht werden müssen. Ich denke aber, es ist möglich, mit einem größeren Team einen dreidimensionalen Dateimanager zu kreieren, der tatsächlich von der Allgemeinheit akzeptiert und verwendet werden wird. Ich persönlich blicke gespannt in die Zukunft, und hoffe, dass dieses Thema in Bezug auf Hardware wie Software wieder etwas mehr in den Fokus rückt. Gerade die intensive Forschung in Bezug auf verschiedene Anzeigemöglichkeiten wirkt auf mich vielversprechend und könnte früher oder später zu einem Umschwung führen.

8.1 Aus der Sicht des Entwicklers

Abgesehen von der teils minimalistischen Dokumentation von *Unity* ging die Arbeit mit der Engine ohne große Probleme voran. Da ich selbst nur sehr wenig Erfahrung mit *Unity* hatte, waren die umfassenden Beginner-Tutorials sehr von Vorteil. Das größte Problem waren überraschenderweise die Elemente, die *Unity* eigentlich ohne Probleme selbst übernehmen sollte, so zum Beispiel die *Collider* der verschiedenen Objekte. Bis zum Schluss fielen etwa Dateiobjekte durch die Containerobjekte, manchmal sofort,

manchmal erst nach einiger Zeit. Oft blieben diese dann auch in den anderen Collidern stecken oder kollidierten mit anderen Dateiobjekten, sodass eines der beiden weg geschleudert wurde.

Trotzdem würde ich auch im Nachhinein wieder *Unity* verwenden, um einen derartigen Prototypen zu entwickeln. Wenngleich eine Lernkurve vorhanden ist und viele Fehler vermutlich nicht ohne Supervision eines versierteren Programmierers behoben werden können, ist es doch gut geeignet für diesen frühen Schritt der Entwicklung.

Anhang A

Inhalt der CD-ROM/DVD

A.1 Generell

Pfad: /

_DaBa.pdf Masterthesis
questionnaire_ger.pdf . Fragebogen der Benutzerstudie
ergebnisse.xlsx Ergebnisse der Benutzerstudie

A.2 Kopien der Onlinequellen

Pfad: /references/

computerhope.pdf . . . What are the most common file types and file extensions? [55].
emac.pdf Dired File Type Faces [56].
gnomefiles.pdf Nautilus in action [54].
guivsccli.pdf Command Line vs. GUI [51].
OSXDaily.pdf Find large files in Mac OS with search [53].
sciencedaily.pdf Better than a hologram: Research produces 3-D images floating in 'thin air' [49]
softpanorama.pdf . . . Orthodox file managers (OFM) Standards [50].
xdotorg.pdf XDotOrg Foundation [52].

A.3 Bilder

Pfad: /graphics/

*.jpg, *.png Verwendete Abbildungen

Teilnehmer # _____

General

1. Ich bin... Männlich Weiblich Anderes / Keine Angabe

2. Alter: _____

3. Beruf / Studiengang: _____

	5	4	3	2	1
	Lehne voll ab	Lehne eher ab	Neutral/Keine Antwort	Stimme zu	Stimme voll zu

Organisation:

1. Ich bin generell eine organisierte Person.

2. Meine persönlichen Dateien sind gut strukturiert.

3. Ich habe Ordner, in die ich alle möglichen Dateien ohne gemeinsames Thema verschiebe.

4. Ich erzeuge Ordner für bestimmte Dateikategorien bereits bevor ich Dateien besitze, die zu dieser Kategorie gehören.

5. Wenn ich neue Dateien erstelle oder erhalte, sortiere ich sie sofort in meine Ordnerstruktur ein.

6. Ich habe nicht viele Unterordner in meiner Ordnerstruktur.

Teilnehmer # _____

	5	4	3	2	1
	Lehne voll ab	Lehne ab	Neutral/Keine Antwort	Stimme zu	Stimme voll zu

7. Auf meinem Desktop liegen viele Dateien.

8. Um eine Datei zu finden, benutze ich meistens...

... den Speicherort. ... die Suchfunktion. _____

9. Wenn ich eine Datei suche, finde ich sie normalerweise sehr schnell.

3D Dateimanager:

1. Haben Sie bereits einen 3D Dateimanager oder 3D Desktop verwendet?

JA NEIN

1.5 Wenn ja, welche(n)?

Antwort: _____

2. Werden Sie in Zukunft einen 3D Dateimanager oder 3D Desktop verwenden?

JA NEIN

3. Ich denke, die Zukunft von Dateimanagern und Desktops liegt in 3D.

Testprogramm:

1. Es hat mir Spaß gemacht, Dateien in 3D zu sortieren.

Teilnehmer # _____	5	4	3	2	1
	Lehne voll ab	Lehne ab	Neutral/Keine Antwort	Stimme zu	Stimme voll zu
2. In 3D würde ich meine Dateien besser sortieren als in 2D.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3. Ich sortiere Dateien lieber in 3D als in 2D.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4. Es war leicht, mich an den Ort der gesuchten Fotos zu erinnern.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5. Ich war vor der Studie mit der <i>WASD</i> -Steuerung vertraut.	<input type="radio"/>	JA	<input type="radio"/>	NEIN	
<hr/>					
Feedback:					
1. Es war leicht, mich herumzubewegen.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2. Die Bedienung war verständlich und leicht zu benutzen.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3. Es war leicht, Dateien so zu bewegen und zu sortieren, wie ich wollte.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4. Die Aufgaben waren klar und verständlich.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5. Ich hatte Spaß an den Aufgaben.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Quellenverzeichnis

Literatur

- [1] Anand Agarawala. „Enriching the Desktop Metaphor with Physics, Piles and the Pen“. Magisterarb. Toronto: University of Toronto, Graduate Department of Computer Science, 2006 (siehe S. 9, 22).
- [2] Anand Agarawala und Ravin Balakrishnan. „Keepin’ It Real: Pushing the Desktop Metaphor with Physics, Piles and the Pen“. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’06. Montréal: ACM, 2006, S. 1283–1292 (siehe S. 1, 9, 10, 24).
- [3] Deborah Barreau. „Context As a Factor in Personal Information Management Systems“. *Journal of the Association for Information Science and Technology* 46.5 (Juni 1995), S. 327–339 (siehe S. 1, 15, 17).
- [4] Deborah Barreau und Bonnie A. Nardi. „Finding and Reminding: File Organization from the Desktop“. *SIGCHI Bulletin* 27.3 (Juli 1995), S. 39–43 (siehe S. 1, 15, 17, 43, 46).
- [5] Michel Beaudouin-Lafon. „Novel Interaction Techniques for Overlapping Windows“. In: *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*. UIST ’01. Orlando: ACM, 2001, S. 153–154 (siehe S. 26).
- [6] Katy Börner. „iScape: A collaborative memory palace for digital library search results“. In: *Proceedings of the 2001 International Conference on Human-Computer Interaction*. New Orleans: Lawrence Erlbaum Associates, Inc, Aug. 2001, S. 1160–1164 (siehe S. 35, 49).
- [7] Stuart K. Card, George G. Robertson und William York. „The WebBook and the Web Forager: An Information Workspace for the World-Wide Web“. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’96. Vancouver: ACM, 1996, S. 111–117 (siehe S. 10).
- [8] Cristiano Carvalheiro u. a. „User Redirection and Direct Haptics in Virtual Environments“. In: *Proceedings of the 2016 ACM on Multimedia Conference*. MM ’16. Amsterdam: ACM, 2016, S. 1146–1155 (siehe S. 25).
- [9] Olivier Chapuis und Nicolas Roussel. „Metisse is Not a 3D Desktop!“ In: *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*. UIST ’05. Seattle: ACM, 2005, S. 13–22 (siehe S. 9).

- [10] Andy Cockburn. „Revisiting 2D vs 3D Implications on Spatial Memory“. In: *Proceedings of the Fifth Conference on Australasian User Interface*. Bd. 28. AUIC '04. Dunedin: Australian Computer Society, Inc., 2004, S. 25–31 (siehe S. 18).
- [11] Andy Cockburn und Bruce McKenzie. „3D or Not 3D?: Evaluating the Effect of the Third Dimension in a Document Management System“. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '01. Seattle: ACM, 2001, S. 434–441 (siehe S. 1, 2, 18–20, 40, 47, 48).
- [12] Andy Cockburn und Bruce McKenzie. „An Evaluation of ConeTrees“. In: *People and Computers XIV – Usability or Else! Proceedings of HCI 2000*. Hrsg. von Sharon McDonald, Yvonne Waern und Gilbert Cockton. Sunderland: Springer London, 2000, S. 425–434 (siehe S. 7).
- [13] Andy Cockburn und Bruce McKenzie. „Evaluating Spatial Memory in Two and Three Dimensions“. *International Journal of Human-Computer Studies* 61.3 (Sep. 2004), S. 359–373 (siehe S. 18).
- [14] Andy Cockburn und Bruce McKenzie. „Evaluating the Effectiveness of Spatial Memory in 2D and 3D Physical and Virtual Environments“. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '02. Minneapolis: ACM, 2002, S. 203–210 (siehe S. 18, 19, 25).
- [15] HTC Corp. „Head mounted display“. Pat. US29537091 (USA). Feb. 2015 (siehe S. 25).
- [16] Sebastian Deterding u. a. „Gamification. Using Game-design Elements in Non-gaming Contexts“. In: *CHI '11 Extended Abstracts on Human Factors in Computing Systems*. CHI EA '11. Vancouver: ACM, 2011, S. 2425–2428 (siehe S. 30).
- [17] Kim M. Fairchild, Steven E. Poltrock und George W. Furnas. „SemNet: Three-dimensional Graphic Representations of Large Knowledge Bases“. In: *Readings in Information Visualization*. Hrsg. von Stuart K. Card, Jock D. Mackinlay und Ben Shneiderman. Morgan Kaufmann Publishers Inc., 1999. Kap. 5, S. 190–206 (siehe S. 7).
- [18] Eric Fassbender und Wolfgang Heiden. „The virtual memory palace“. *Journal of Computational Information Systems* 2.1 (März 2006), S. 457–464 (siehe S. 35).
- [19] Joshua Foer. *Moonwalking with Einstein. The art and science of remembering everything*. Englisch. New York: Penguin Press, 2011 (siehe S. 35).
- [20] Sarah Henderson und Ananth Srinivasan. „Filing, Piling & Structuring: Strategies for Personal Document Management“. In: *Proceedings of the 2011 44th Hawaii International Conference on System Sciences*. Kuauai: IEEE, Jan. 2011, S. 1–10 (siehe S. 10, 13, 14).
- [21] Lawrence J. Hettinger und Gary E. Riccio. „Visually Induced Motion Sickness in Virtual Environments“. *Presence: Teleoperators and Virtual Environments* 1.3 (Juli 1992), S. 306–310 (siehe S. 26).
- [22] Joseph 'Jofish' Kaye u. a. „To Have and to Hold: Exploring the Personal Archive“. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '06. Montréal: ACM, 2006, S. 275–284 (siehe S. 13, 27).

- [23] Jiwon Lee, Mingyu Kim und Kim Jinmo. „A Study on Immersion and VR Sickness in Walking Interaction for Immersive Virtual Reality Applications“. *Symmetry* 9 (Mai 2017), S. 78 (siehe S. 26).
- [24] Ellen Levy u. a. „Gratuitous Graphics? Putting Preferences in Perspective“. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '96. Vancouver: ACM, 1996, S. 42–49 (siehe S. 19).
- [25] Thomas W. Malone. „How Do People Organize Their Desks?: Implications for the Design of Office Information Systems“. *ACM Transactions on Information Systems* 1.1 (Jan. 1983), S. 99–112 (siehe S. 1, 10–12, 15, 27, 43, 46).
- [26] Bonne Nardi, Ken Anderson und Thomas Erickson. „Filing and Finding Computer Files“. In: *Proceedings of the East-West HCI*. Moskau, Juli 1995, S. 162–179 (siehe S. 1, 15, 17).
- [27] Pamela Ravasio, Sissel Guttormsen Schär und Helmut Krueger. „In Pursuit of Desktop Evolution: User Problems and Practices with Modern Desktop Systems“. *ACM Transactions on Computer-Human Interaction* 11.2 (Juni 2004), S. 156–180 (siehe S. 16).
- [28] George G. Robertson, Jock D. Mackinlay und Stuart K. Card. „Cone Trees: Animated 3D Visualizations of Hierarchical Information“. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '91. New Orleans: ACM, 1991, S. 189–194 (siehe S. 7, 8).
- [29] George Robertson u. a. „Data Mountain: Using Spatial Memory for Document Management“. In: *Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology*. UIST '98. San Francisco: ACM, 1998, S. 153–162 (siehe S. 18).
- [30] Daniel Smalley u. a. „A photophoretic-trap volumetric display“. *Nature* 553 (Jan. 2018), S. 486–495 (siehe S. 20).
- [31] Mark St. John u. a. *Navigating Two-Dimensional and Perspective Views of Terrain*. Techn. Ber. 1827. San Diego, CA: SPAWAR Systems Center San Diego, Juli 2000. URL: <http://www.dtic.mil/dtic/tr/fulltext/u2/a381612.pdf> (siehe S. 18).
- [32] Steve Whittaker und Julia Hirschberg. „The Character, Value, and Management of Personal Paper Archives“. *ACM Transactions on Computer-Human Interaction* 8.2 (Juni 2001), S. 150–170 (siehe S. 1, 12, 27).
- [33] Christopher D. Wickens, Sonia Miller und Mingpo Tham. „The Implications of Data-Link for Representing Pilot Request Information on 2D and 3D Air Traffic Control Displays“. In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. Bd. 38. 1. Philadelphia: HFES, 1994, S. 61–65 (siehe S. 18).
- [34] Christopher D. Wickens u. a. „Egocentric and Exocentric Displays for Terminal Area Navigation“. In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. Bd. 38. 1. Philadelphia: HFES, 1994, S. 16–20 (siehe S. 1, 18).
- [35] So-Yeon Yoon, James Laffey und Hyunjoo Oh. „Understanding Usability and User Experience of Web-Based 3D Graphics Technology“. *International Journal of Human-Computer Interaction* 24.3 (2008), S. 288–306 (siehe S. 6).

Audiovisuelle Medien

- [36] *Avatar*. Film. Drehbuch/Regie: James Cameron. Mit Sam Worthington, Zoe Saldana, Stephen Lang, Michelle Rodriguez, Sigourney Weaver. 2009 (siehe S. 20).
- [37] *Ready Player One*. Film. Drehbuch: Zak Penn und Ernest Cline. Regie: Steven Spielberg. Mit Tye Sheridan, Olivia Cooke, Ben Mendelsohn, T.J. Miller, Simon Pegg, Mark Rylance. 2018 (siehe S. 20).

Software

- [38] Anand Agarawala. *BumpTop*. 2012. URL: <https://bumpTop.github.io/> (siehe S. 9, 24).
- [39] Inc. Dropbox. *Dropbox*. 2007. URL: <https://www.dropbox.com/> (siehe S. 16).
- [40] Daniel Richard G. *fsv - File System Visualizer*. 1999. URL: <http://fsv.sourceforge.net/> (siehe S. 7).
- [41] Friction Games. *Amnesia: The dark descent*. 2010. URL: <https://www.amnesia-game.com/> (siehe S. 34).
- [42] Silicon Graphics. *fsv - File System Navigator*. URL: <https://www.linux.com/news/my-workstation-os-irix> (siehe S. 7, 8).
- [43] Stanley G. Kugell. *DIRECT*. URL: <http://www.saildart.org/DIRECT.SGK%5BUP%2CDOC%5D1> (siehe S. 4).
- [44] Google LLC. *Google Drive*. 2012. URL: <https://drive.google.com/> (siehe S. 16).
- [45] Fernando Santesteban. *Cubix*. URL: <http://www.cubixproject.com/> (siehe S. 7, 8).
- [46] Marc Schiller. *RealDesktop*. URL: <http://www.real-desktop.de/> (siehe S. 9, 24).
- [47] Open Source. *XWindowing*. URL: <https://www.freedesktop.org/wiki/Desktops/> (siehe S. 5).
- [48] Unity Technologies. *Unity*. Version 2018.1. 2018. URL: <https://unity3d.com/> (siehe S. 33).

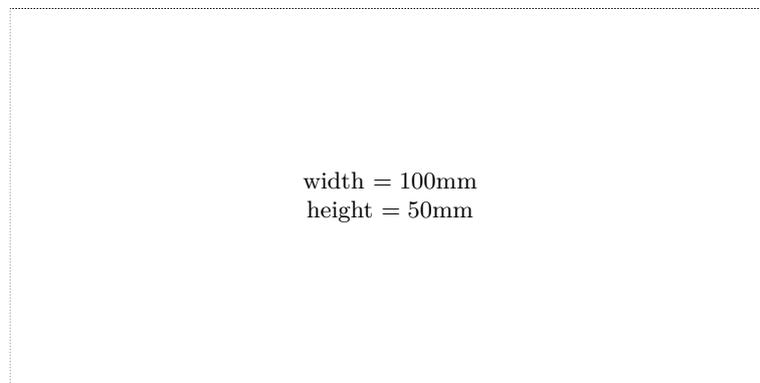
Online-Quellen

- [49] *Better than a hologram: Research produces 3-D images floating in 'thin air'*. URL: <https://www.sciencedaily.com/releases/2018/01/180124151154.htm> (besucht am 05.05.2018) (siehe S. 20, 53).
- [50] Nikolai Bezroukov. *Orthodox file managers (OFM) standards*. URL: <http://www.softpanorama.org/OFM/Standards/index.shtml> (besucht am 06.01.2018) (siehe S. 5, 6, 53).
- [51] *Command line vs. GUI*. Dez. 2017. URL: <https://www.computerhope.com/issues/c000619.htm> (besucht am 27.05.2018) (siehe S. 50, 53).

- [52] X.Org Foundation. *X.Org*. URL: <https://www.x.org/wiki/> (besucht am 06. 03. 2018) (siehe S. 9, 53).
- [53] Paul Horowitz. URL: <http://osxdaily.com/2012/04/17/find-large-files-in-mac-os-x-search/> (besucht am 11. 03. 2018) (siehe S. 16, 53).
- [54] *Nautilus in action*. URL: <https://projects.gnome.org/nautilus/screenshots.html> (besucht am 01. 19. 2018) (siehe S. 5, 6, 53).
- [55] *What are the most common file types and file extensions?* Jan. 2018. URL: <https://www.computerhope.com/issues/ch001789.htm> (besucht am 17. 04. 2018) (siehe S. 24, 53).
- [56] Ji Xiufeng. *Dired File Type Faces*. URL: <https://www.emacswiki.org/emacs/DiredFileTypeFaces> (besucht am 06. 01. 2018) (siehe S. 6, 53).

Messbox zur Druckkontrolle

— Druckgröße kontrollieren! —



— Diese Seite nach dem Druck entfernen! —