



Fortgeschrittene Konzepte der Prozessmodellierung durch den Einsatz von Multi-Touch-Gesten

Bachelorarbeit an der Universität Ulm

Vorgelegt von:

Benjamin Rudner
benjamin.rudner@uni-ulm.de

Gutachter:

Prof. Dr. Manfred Reichert

Betreuer:

Jens Kolb

2011

Fassung 31. Oktober 2011

© 2011 Benjamin Rudner

This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/de/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.

Satz: PDF- \LaTeX 2_ε

Inhaltsverzeichnis

1	Einleitung	1
2	Grundlagen	3
2.1	Benutzertypen und Anwendungsszenarien	3
2.1.1	Benutzertypen und ihre Anforderungen	3
2.1.2	Anwendungsszenarien und Begleiterscheinungen von Multi-Touch- Anwendungen	5
	Verdeckung der Oberfläche und Präzision beim Tippen	7
	Ermüdung der Gliedmaßen	8
	Anzahl der benutzten Finger	9
	Beidhändige Bedienung	10
	Zusammenarbeit mehrerer Benutzer	11
	Besonderheit Multi-Touch-Wand	12
2.2	Zeichensysteme und Gesten	12
2.2.1	Gebärdensprache als Vorbild	13
2.2.2	Symbole im Alltag	14
2.2.3	Apples Multi-Touch Gesture Dictionary	15
2.3	Zusammenfassung der Grundlagen	17
3	Konzepte zur Prozessmodellierung auf Multi-Touch-Geräten	19
3.1	Entwicklung einer Gesten-basierten Modellierung	19
3.1.1	Gesten-Set für die Modellierungsfunktionen	20
	F1: Aktivität einfügen	20
	F2: Element benennen	22
	F3: Verzweigungsblock einfügen	23
	F4: Verzweigung einfügen	25
	F5: Datenelement einfügen	26
	F6: Lese- und Schreib-Kanten setzen	27

Inhaltsverzeichnis

F7: Subprozess bilden und auflösen	27
F8: Element Löschen	28
3.1.2 Gesten-basierte Hilfsfunktionen	30
H1: Markieren	30
H2: Rückgängig und Wiederholen	30
H3: Verschieben und Kopieren	31
H4: Hilfe aufrufen	32
3.1.3 Erlernen der Gesten-basierten Prozessmodellierung	33
3.1.4 Bewertung des Gesten-basierten Ansatzes	33
3.2 Entwicklung einer menügeführten Modellierung	34
3.2.1 Drag & Drop-Menüleiste	34
3.2.2 Slider-Menüleiste	36
3.2.3 Bewertung des menügeführten Ansatzes	38
3.3 Hybridlösung basierend auf Gesten und Menüs	39
3.3.1 Hybrid-Set zur Prozessmodellierung	39
F1: Aktivität einfügen	39
F2: Element umbenennen	40
F3: Verzweigungsblock einfügen	40
F4: Verzweigung einfügen	41
F5: Datenelement einfügen	41
F6: Lese- und Schreib-Kanten setzen	42
F7: Subprozess bilden und auflösen	42
F8: Element Löschen	43
3.3.2 Hilfsfunktionen	43
H1: Markieren	43
Element-bezogen: H3: Verschieben und Kopieren, H4: Hilfe aufrufen .	44
Nicht Element-bezogen: H2: Rückgängig und Wiederholen, H4: Hilfe aufrufen	44
3.3.3 Bewertung der Hybridlösung	44
3.4 Fazit der Konzeption	45
4 Experimentelle Untersuchung	47
4.1 Vorbereitung und Durchführung der experimentellen Untersuchung	48
4.2 Auswertung der experimentellen Untersuchung	50

4.2.1	Detaillierte Auswertung der Aufgaben	50
	1. Aufgabe - F1: Aktivität erstellen	50
	2. Aufgabe - F2: Element benennen	52
	3. Aufgabe - F6: Lese-Kante setzen	53
	4. Aufgabe - F4: Verzweigung einfügen	55
	5. Aufgabe - F3: Verzweigungsblock einfügen	55
	6. Aufgabe - F5: Datenelement einfügen	58
	7. Aufgabe - F7: Subprozess bilden	60
	8. Aufgabe - F8: Element löschen	61
4.2.2	Resultierendes Gesten-Set aus der experimentellen Untersuchung . .	63
4.2.3	Abschließende Bemerkungen	65
5	Zusammenfassung	67
A	Anhang	69
	A.1 Ergebnisse der experimentellen Untersuchung im Überblick	69
	Literaturverzeichnis	77

1 Einleitung

Das Stichwort "Multi-Touch" erlangte im Jahr 2007 durch Microsoft Surface und vor allem durch Apples iPhone große Popularität. Diese Geräten lassen sich mit mehreren Fingern gleichzeitig (daher das Wort "Multi") direkt über ihren Bildschirm bedienen. Dies geschieht über sogenannte Gesten, also vom Hersteller festgelegte Fingerbewegungen, welche das Programm algorithmisch auswertet, sobald die Finger die (Bildschirm-)Oberfläche berühren (englisch: "Touch"). Die zugrunde liegenden Technologien dafür gibt es aber bereits seit 1982. Alan Kay arbeitete am Xerox Parc im Jahr 1968, also noch vor dem ersten PC, an einem flachen Pad mit Tastatur, dem DynaBook [19]. Doch erst sinkende Herstellungskosten und speziell zugeschnittene Software konnten den heutigen kommerziellen Erfolg ermöglichen. Für Aufsehen sorgte im Jahr 2006 ein TED-Vortrag von Jefferson Han über eine kostengünstige Multi-Touch-Lösung basierend auf Frustrated Total Internal Reflection (FTIR) [17, 4]. Seine Multi-Touch Collaboration Wall fand ebenso medienwirksam während der US-Präsidentenwahl 2008 beim Nachrichtensender CNN seinen Einsatz [14].

All diese Beispiele zeigen, dass Multi-Touch-Technologien einen langen Vorlauf brauchten bis diese günstig einer breiten Öffentlichkeit zur Verfügung standen. So ermöglicht sie speziell bei mobilen Geräten eine Vergrößerung des Bildschirms bei gleichbleibender Gerätegröße, da die Tastatur zur Eingabe wegfallen kann. Der Vorteil dieser Technologie ist, dass Eingabe und Ausgabe an derselben Stelle ein Gefühl der direkten Manipulation vermitteln, welche speziell für ältere Menschen sehr intuitiv sein kann [9]. Das alles macht Multi-Touch-Technologie interessant für Einsatzzwecke, die bisher über den Computer umständlich oder einschüchternd gewirkt haben können oder per se besser in einer Gruppenarbeit erfolgen. Darunter fallen vor allem Anwendungen wie Graphenbearbeitungen, wo es sinnvoll ist die Elemente anfassen zu können. Eine besondere Anwendung dieser Gattung, die in den letzten Jahren eine immer stärkere Entwicklung erfährt, ist die Geschäftsprozessmodellierung. Mit ihr ist es möglich Abläufe in einem Unternehmen zu modellieren, analysieren und zu implementieren, um durch so optimierte Prozesse wettbewerbsfähiger zu werden. In der Regel sind diese Prozesse so umfangreich, dass sie nicht vollständig

1 Einleitung

auf eine Multi-Touch-Oberfläche passen und gleichzeitig mit den Fingern bedient werden können. Es müssen also Wege gefunden werden, diesen Konflikt aufzulösen.

In dieser Arbeit wird nun versucht die Handhabung der Prozessmodellierung konzeptionell auf Multi-Touch-Geräten umzusetzen, um die Vorteile dieser Technik zu nutzen und den Einsatzbereich der Prozessmodellierung zu erweitern. Dazu werden grundlegende Gesten für die Modellierung von Prozessmodellen, basierend auf der Business Process Modeling Notation (BPMN), entwickelt und experimentell validiert.

Diese Arbeit gliedert sich wie folgt: in Kapitel 2 wird der aktuelle Stand der Literatur zum Thema Multi-Touch vorgestellt, die sich im Wesentlichen auf Ergebnisse von Benutzerstudien, Designstudien und biologische Faktoren bezieht. Die Ergebnisse dieser Recherche dienen als Grundlage für die Ausarbeitung zweier konträrer Ansätze zur Modellierung von Prozessen in Kapitel 3. Zum einen ist das ein Gesten-basierter Ansatz, d.h. die gesamte Steuerung des Programms basiert nur auf Gesten, und zum anderen ist das eine vollständig menügeführte Modellierung. Anschließend wird aus diesen beiden konträren Ansätzen ein hybrider Ansatz entwickelt. In Kapitel 4 wird ein Experiment zur Multi-Touch-Modellierung vorgestellt und die Ergebnisse werden diskutiert. Abschließend findet sich in Kapitel 5 eine Zusammenfassung der Arbeit. Der gesamte Ablauf ist noch einmal in Abbildung 1.1 visualisiert.

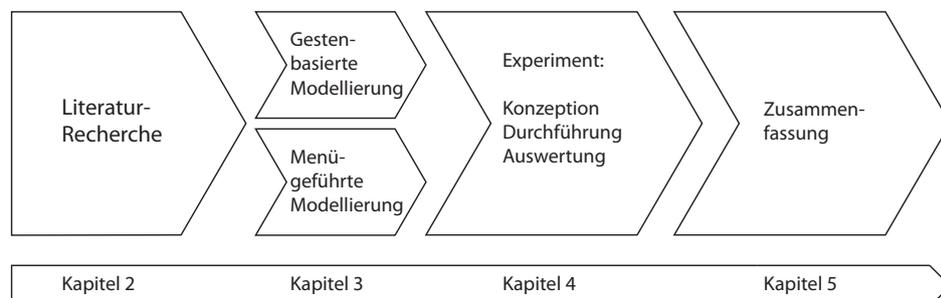


Abbildung 1.1: Gliederung der Bachelorarbeit

2 Grundlagen

Für das weitere Verständnis müssen zuerst einige Grundlagen eingeführt werden. Diese haben großen Einfluss auf spätere Entscheidungen bei der Konzeption und Untersuchung der entwickelten Konzepte. Dazu werden wir zuerst in Kapitel 2.1 die verschiedenen Benutzertypen einführen und welche Anwendungsszenarien es gibt bzw. welche Besonderheiten mit dem Einsatz von Multi-Touch-Technologien verbunden sind. Anschließend werden wir in Kapitel 2.2 auf die Rolle von Zeichen und Symbolen für Multi-Touch-Gesten eingehen.

2.1 Benutzertypen und Anwendungsszenarien

Ein Benutzer wird im Folgenden genannt, wer ein Programm bedient. Entwickelt man ein Softwareprodukt müssen immer die verschiedenen Benutzertypen, also die Zielgruppen, und Anwendungsszenarien betrachtet werden. Der Benutzer soll sein Problem effektiv, effizient und zufrieden in einem bestimmten Nutzungskontext lösen können. Dieser Nutzungskontext wird durch die Umgebung bestimmt, in der ein Benutzer arbeitet und spielt vor allem dann eine Rolle, wenn man den herkömmlichen Computer im Büro verlässt. Offensichtlich stellen sich andere Herausforderungen an denselben Benutzer und somit auch an die entsprechende Soft- und Hardware, wenn er unterwegs, alleine oder mit anderen zusammen am Whiteboard arbeitet.

2.1.1 Benutzertypen und ihre Anforderungen

Bei jedem neuen Programm gibt es zu Beginn nur Anfänger und erst nach und nach bilden sich erfahrene Benutzer (sogenannte Experten) heraus. Es ist dabei ein Unterschied, ob der Benutzer generell Erfahrung mit Computern besitzt oder ob die Technologie neu für ihn ist. So wird ein Benutzer, der bereits Erfahrung mit ähnlichen Technologien gesammelt

2 Grundlagen

hat, einen leichteren Einstieg in das neue Programm finden als einer, der noch keine Erfahrung sammeln konnte. Neben den EDV-Kenntnissen an sich spielt aber auch die Häufigkeit der Benutzung eine entscheidende Rolle. Jemand, der täglich mit einem bestimmten Programm arbeitet, wird sich darin besser zurechtfinden und größere Fortschritte machen, als jemand der nur sporadisch damit arbeitet. Kombiniert man all diese Eigenschaften ergibt sich die Matrix in Tabelle 2.1 [23]. An dieser Stelle sei angemerkt, dass das Aufgaben-Jobalter angibt, wie lange jemand bereits mit der Aufgabe vertraut ist.

Tabelle 2.1: Benutzertypen

Fachliche Kenntnisse	EDV-Kenntnisse	Nutzung von EDV	Benutzertypen	Beispiele
Aufgaben-Jobalter ≥ 3 Monate und Ausbildung min. mit genügend beurteilt	EDV-Jobalter ≥ 1 Monat und	durchschnittl. Nutzung von 2-3 Stunden/Tag	“geübter und regelmäßiger” Benutzer	Benutzerbetreuer einer Fachabteilung
	EDV-Ausbildung min. mit gut beurteilt	durchschnittl. Nutzung von $< 2-3$ Stunden/Tag	“geübter und sporadischer” Benutzer	Wissensverarbeiter (Mitarbeiter einer Stabsabteilung)
	EDV-Jobalter < 1 Monat oder	durchschnittl. Nutzung von 2-3 Stunden/Tag	“ungeübter und regelmäßiger” Benutzer	Assistenz- und Schreibkräfte, Sachbearbeiter in Banken, Versicherungen
	EDV-Ausbildung weniger als gut beurteilt	durchschnittl. Nutzung von $< 2-3$ Stunden/Tag	“ungeübter und sporadischer” Benutzer	Manager, Sachbearbeiter

Jeder Benutzertyp legt dabei auf verschiedene Aspekte besonderen Wert. Wirft man einen Blick auf die Tabelle 2.2 sieht man, dass es unabhängig vom Erfahrungsgrad, Eigenschaften eines konventionellen Programms gibt, die von allen Benutzern als wichtig angesehen werden [23]. Darunter fallen “Kooperations- und Kommunikationsförderlichkeit”, “Datenschutz/Datensicherheit”, “Nützlichkeit”, “Komfort” und “Erwartungskonformität”. Es gibt jedoch auch Kriterien, die von unterschiedlichen Benutzertypen unterschiedlich stark gewichtet werden. Vergleicht man sporadische Benutzer mit den regelmäßigen Benutzern

2.1 Benutzertypen und Anwendungsszenarien

unterscheiden sie sich in der Bewertung von "Verfügbarkeit", "Erlernbarkeit" und "Selbstbeschreibungsfähigkeit". So sind regelmäßige Benutzer darauf angewiesen, dass ihr Programm stabil läuft und ein angemessenes Antwortzeitverhalten zeigt. Sporadischen Benutzern hingegen ist dies nicht sonderlich wichtig, da sie sich nicht so oft damit beschäftigen müssen. Dafür ist es für sie um so wichtiger, aufgrund mangelnder Erfahrung schnell ein neues Programm zu erlernen, was durch eine schnelle Erlernbarkeit sowie eine selbst-erklärende Oberfläche begünstigt wird. Hilfreich sind dabei Konzepte wie Lernen durch Nachahmung, Learning-by-Doing, also Beispiele und Anleitungen, aber auch verständliche Rückmeldungen vom Programm. Betrachtet man "geübte und regelmäßige Benutzer", stellt man fest, dass sie großen Wert auf "Steuerbarkeit" und "Individualisierbarkeit" legen im Gegensatz zu den anderen Benutzertypen, die diese als unwichtig einstufen. Sie kennen sich mit dem Programm aus und wollen es auf ihre Fähigkeiten und Vorlieben zuschneiden, um das Arbeiten noch angenehmer zu gestalten. Dazu gehört auch die Interaktionsrichtung und -geschwindigkeit selbst bestimmen zu können. Der Benutzertyp "ungeübte und sporadische Benutzer" hingegen ist mehr als die anderen auf "Übersichtlichkeit" und "Fehlerrobustheit" angewiesen. Denn er kennt sich mit dem Programm nicht aus und viele neue Eindrücke sind zu verarbeiten. Es ist daher wichtig, durch eine übersichtliche Oberfläche seine Wahrnehmung zu schärfen und das Arbeitsgedächtnis zu entlasten. Programm- oder Eingabefehler sollten nicht vorkommen und sollten entweder direkt automatisch korrigiert werden, oder seitens des Benutzers durch aussagekräftige Meldungen behoben werden können.

2.1.2 Anwendungsszenarien und Begleiterscheinungen von Multi-Touch-Anwendungen

Es gibt verschiedene Szenarien unter denen ein Arbeiten mit Computern möglich ist und für jedes gibt es spezielle Hardware- und Software-Lösungen. Beginnend z.B. bei den älteren PDAs (Personal Digital Assistant). Das sind kleine, tragbare Computer, die hauptsächlich für Termin- und Adressverwaltung eingesetzt werden und mit einem speziellen Bedienstift benutzt werden. Daneben gibt es die moderneren Smartphones, wie das Apple iPhone, die neben der Computerfunktionalität auch als Mobiltelefon verwendet werden können. Beide Geräte passen in die Hosentasche und sind deshalb für mobile Programme geeignet. Schreibpapier große Tablets wie das Apple iPad und Laptops, die über eine Multi-Touch-Oberfläche verfügen, sind ebenfalls sehr handlich und für mobile Zwecke geeignet. Die

2 Grundlagen

Tabelle 2.2: Geübter/ungeübter und regelmäßiger/sporadischer Benutzer

Kriterium:	Geübter	regelmäßiger B.		sporadischer B.	
		Rang:	Gewicht:	Rang:	Gewicht:
Kooperations-, Komm.-förderlichkeit	1.	hoch	1.	hoch	
Datenschutz/Datensicherheit	2.	hoch	2.	hoch	
Verfügbarkeit	3.	hoch	12.	gering	
Nützlichkeit	4.	hoch	3.	hoch	
Komfort	5.	hoch	4.	hoch	
Steuerbarkeit	6.	hoch	10.	gering	
Individualisierbarkeit	7.	hoch	11.	gering	
Erwartungskonformität	8.	hoch	5.	hoch	
Selbstbeschreibungsfähigkeit	9.	mittel	6.	hoch	
Übersichtlichkeit	10.	mittel	9.	mittel	
Fehlerrobustheit	11.	mittel	8.	mittel	
Erlernbarkeit	12.	gering	7.	hoch	

	Ungeübter	regelmäßiger B.		sporadischer B.	
		Rang:	Gewicht:	Rang:	Gewicht:
Kooperations-, Komm.-förderlichkeit	1.	hoch	1.	hoch	
Datenschutz/Datensicherheit	2.	hoch	2.	hoch	
Verfügbarkeit	3.	hoch	12.	gering	
Nützlichkeit	4.	hoch	3.	hoch	
Komfort	5.	hoch	4.	hoch	
Steuerbarkeit	12.	gering	11.	gering	
Individualisierbarkeit	11.	gering	10.	gering	
Erwartungskonformität	6.	hoch	8.	hoch	
Selbstbeschreibungsfähigkeit	9.	mittel	7.	hoch	
Übersichtlichkeit	10.	mittel	9.	hoch	
Fehlerrobustheit	8.	mittel	6.	hoch	
Erlernbarkeit	7.	mittel	5.	hoch	

nächstgrößeren Geräte sind Multi-Touch-Tische wie der Microsoft Surface und elektronische Tafeln, auch digitale Whiteboards oder Smartboards genannt. Diese stehen an einem festen Ort und eignen sich für Präsentationen, Gruppenarbeiten und ähnlich komplexe Aufgaben.

Jede Größe dient also verschiedenen Einsatzzwecken und besitzt unterschiedliche Eigenschaften, die Einfluss auf die Entwicklung der dazugehörigen Software haben. Darunter fallen die Verdeckung der Oberfläche, die Präzision beim Tippen, die Ermüdung der Gliedmaßen, die Anzahl der benutzten Finger, beidhändige Bedienung, die Zusammenarbeit mehrerer Benutzer und die Besonderheit einer Multi-Touch-Wand. Diese Aspekte wollen wir im Folgenden genauer betrachten.

Verdeckung der Oberfläche und Präzision beim Tippen

Auf einer größeren Multi-Touch-Oberfläche lassen sich mehr Informationen unterbringen als auf einer kleinen. Für das Auge mag das weniger eine Rolle spielen, solange die Anzahl der Pixel pro Maßeinheit, also die Auflösung, ein bestimmtes Maß nicht unterschreitet. Im Gegensatz zu einem Mauszeiger allerdings skaliert der Finger mit der Größe der Oberfläche nicht mit. Dies hat zur Folge, dass Teile der Oberfläche beim Tippen verdeckt werden und die Treffsicherheit abnimmt, vor allem wenn die Elemente kleiner als der eigene Finger sind. So wurde in einer Studie aus dem Jahr 1988 festgestellt, dass Flächen mit einer Seitenlänge von 10 mm mit dem Zeigefinger in 66,7% aller Fälle getroffen werden. Die höchste Genauigkeit erzielt man bei einer Seitenlänge der Fläche von 26 mm. Hier liegt die Wahrscheinlichkeit bei 99,2% [16]. Vergrößert man unnötigerweise die Flächen für tippbare Flächen, verliert man auf Oberflächen kostbaren Platz für das eigentliche Programm. Geht man beispielsweise von einer Bildschirmgröße von 5 cm auf 7 cm aus, würde ein Button mit Seitenlänge 26 mm bereits 20% der Fläche bedecken. In einer aktuellen Studie konnte man nachweisen, dass Flächen eine Seitenlänge von mindestens 11,52 mm haben sollten, um eine 95%-ige Trefferwahrscheinlichkeit sicherzustellen [30]. Diese beachtlichen Abweichungen sind auf Unterschiede in den Geräten und Messauflösungen zurückzuführen. Je nach technischem Stand des Geräts müssen diese Unterschiede also bei der Konzeption eines Programms beachtet werden. Ein weiterer Aspekt, den es zu beachten gilt ist, dass sich der Mauszeiger/Cursor während der Berührung der Oberfläche leicht verschiebt, da die Berührung kontinuierlich verläuft. Der Finger trifft also woanders auf als er abhebt, was den Cursor entsprechend verschiebt.

Stellen diese Flächengrößen eine zu hohe Einschränkung für das Programm dar, muss entweder ein Trade-Off, also eine Balance, zwischen nutzbarer Fläche und Trefferquote gefunden werden, die Oberfläche an sich verändert werden oder es müssen andere Methoden bei der Eingabe herangezogen werden. Nachfolgend werden verschiedene Eingabemethoden beschrieben.

Neben den Fingern können auch Bedienstifte zur Eingabe verwendet werden. In einer Studie aus dem Jahre 2007 wurde der Umgang mit PDAs und dazugehörigem Bedienstift während dem Sitzen und Laufen untersucht [22]. Dabei fand man heraus, dass für eine 90%-ige Genauigkeit mit dem Bedienstift ein Durchmesser von 6,4 mm des Ziels schon ausreicht. Dies entspricht ungefähr einer Seitenlänge von 2,5 mm bei einem Quadrat. Gleichzeitig verringerten die Probanden jedoch ihre Laufgeschwindigkeit um 36%. Denn je nach Um-

2 Grundlagen

gebungssituation muss die Aufmerksamkeit mit anderen Ereignissen geteilt werden – die kognitive Belastung erhöht sich.

Für die Gestaltung von Multi-Touch-Oberflächen kommt es also darauf an, ob ein Bedienstift verwendet wird oder nicht. Wird ein Bedienstift verwendet, können die Elemente auf dem Bildschirm kleiner gehalten werden, als wenn der Benutzer seine Finger zur Eingabe einsetzt.

Eine weitere Methode die Präzision zu erhöhen, besteht in der sogenannten Take-Off-Technik. Dabei wird nicht die Flächenmitte des abgebildeten Fingerabdrucks als Cursor interpretiert, sondern dieser wird für den Benutzer sichtbar knapp oberhalb des Fingers abgebildet, sobald dieser die Oberfläche berührt. Ein Ereignis wird dabei erst ausgelöst, wenn der Finger wieder angehoben wird. Dies hat den Vorteil, dass der Benutzer eine genaue Rückmeldung darüber erhält, welchen Punkt er auswählt, und somit seine Fehlstellung korrigieren kann, bevor er die Funktion auslöst. Gleichzeitig wird so eine Verdeckung des Elements durch den Finger vermieden [25].

Eine andere Technik namens "Shift" begegnet diesem Problem, indem die Fläche unter dem Finger mit einer kleinen Verschiebung als Bildschirmlupe neben dem Finger noch einmal angezeigt wird [28].

Beide Techniken mögen eine Verbesserung hinsichtlich der Treffgenauigkeit von Elementen sein und somit eine Einsparung von Fläche ermöglichen, doch verändern sie die Herangehensweise an Multi-Touch-Oberflächen. Die Vorstellung, dass man von oben kommt und etwas durch Berühren auslöst, wird erst einmal umgekehrt und anstatt auf das gewünschte Objekt zu tippen, wie man es bereits von der Benutzung einer Maus gewohnt ist, muss der Benutzer daneben bzw. darunter tippen.

Möchte man also Multi-Touch-Technologie in seiner Reinform nutzen, ohne über Zwischeneingabegeräte wie Bedienstift oder Cursor von der gewonnenen Nähe zur Oberfläche wieder abzurücken, muss man entweder unbeabsichtigte Tipps in Kauf nehmen oder die Menü- und Befehlsstruktur der Anwendung überdenken. Hier setzen nicht zuletzt Gesten an, aber etwa auch Sprachbefehle.

Ermüdung der Gliedmaßen

Ein weiteres wichtiges Problem sind die Ermüdungserscheinungen in Hand, Arm und Gelenke, die bei längerer Bedienung von Multi-Touch-Geräten auftreten. Während bei her-

2.1 Benutzertypen und Anwendungsszenarien

kömmlichen Computern die Maus mit geringen Bewegungen der Hand gesteuert werden muss, um alle Bereiche der Oberfläche zu erreichen, muss bei Multi-Touch-Geräten die Hand bzw. der ganze Arm selbst an die geforderten Stellen der Oberfläche bewegt werden. Je größer und häufiger die Strecken zurückgelegt werden, desto eher tritt eine Ermüdung der Gliedmaßen ein. Ein Problem welches vor allem auf größeren Multi-Touch-Oberflächen zum Tragen kommt. Ein weiterer Faktor ist das ständige Tippen mit der Hand auf die Oberfläche des Multi-Touch-Gerätes.

Beide Probleme müssen durch eine intelligente Steuerung minimiert werden. Ein Vorteil von mobilen Geräten ist sicherlich, dass man eine bequeme Arbeitshaltung einnehmen kann, wenn man mit ihnen arbeitet. Findet man allerdings keine geeignete Ablage für mobile Geräte, müssen sie in der Hand gehalten werden, was wiederum eine schnelle Ermüdung fördert. Bei größeren Geräten hingegen muss die Arbeitshaltung an diese Geräte angepasst werden, sofern sie nicht höhenverstellbar, dreh- oder neigbar sind.

Anzahl der benutzten Finger

Während man auf einer menügeführten Oberfläche durch Schaffung einer Hierarchie scheinbar beliebig viele Funktionen unterbringen kann, muss die gleiche Funktionalität auf Multi-Touch-Oberflächen auf Gesten abgebildet werden. Bei diesen Gesten muss eine Unterscheidung zwischen einem und mehreren Fingern bei der Bedienung gemacht werden, um den gleichen Funktionsumfang wie bei menügeführten Programmen zu gewährleisten.

Usability-Designer versuchen dazu Gesten einer logischen Ordnung folgend zu entwickeln und bauen ein Gesten-System auf, welches Ein-, Zwei-, Drei- und Fünf-Finger-Gesten unterschiedliche Funktionsebenen zuschreibt. So werden beim Apple iPad mit einem Finger einzelne Elemente in einem Programm gesteuert, mit zwei Fingern generelle Befehle innerhalb des Programms ausgeführt, wie zum Beispiel das Zoomen über die Pinch-Geste (d.h. die entgegengesetzte Bewegung zweier Finger [24]) und mit mehreren Fingern zwischen den Programmen durch Wischen hin und her gewechselt, als wären die einzelnen Programme nebeneinander angeordnet [7]. Im Kontrast dazu zeigen Studien, die vor allem die Benutzer in die Entwicklung von Gesten miteinbezogen haben, dass ein einfacher Tipp nicht nur mit einem Finger, sondern auch mit zwei oder sogar mit drei Fingern ausgeführt wird [15, 31]. Gründe hierfür waren unter Anderem, dass die Benutzer dies bei größeren

2 Grundlagen

Objekten oder Aufgaben für nötig gehalten haben, so als ob sie mehr Kraft dafür benötigten oder dass es semantisch mit der Aufgabe in Beziehung stand, (z.B. ein Objekt zu vergrößern). Ein weiterer Grund war sicherzugehen, dass das Programm die Geste tatsächlich wahrnimmt, was aber auch zum Teil an der fehlenden Rückmeldung des Programms in der Studie gelegen haben könnte. Diese Erkenntnis macht es erforderlich Wirkungsweisen visuell zu verdeutlichen und ausreichend Rückmeldung über den Zustand des Programms zu geben, legt aber auch nahe auf eine feingliedrige Unterscheidung zwischen der Anzahl der Finger zu verzichten, wenn man ein möglichst natürliches Gesten-Set anbieten möchte.

Beidhändige Bedienung

Das Arbeiten mit beiden Händen ist vor allem dann eine Option, wenn das Gerät nicht mit einer Hand gehalten werden muss. Wobei selbst das noch möglich sein könnte, wenn der Daumen der haltenden Hand zu Hilfe genommen wird. Eine beidhändige Bedienung hat jedoch den Vorteil, dass man zwei Zeigefinger zur Verfügung hat, welche beim Einsatz von Multi-Touch-Technologie sehr beliebt sind. Es erhöht den Komfort für Gesten, die mit einer Hand zu umständlich wären. So wird ein Zoomen auf dem Apple iPad zwar mit der Pinch-Geste aus dem Zusammenspiel von Daumen und Zeigefinger erzielt. Den gleichen Effekt erhält man aber auch, wenn man dafür beide Hände zu Hilfe nimmt, da nur zwei Punkte aufeinander zu oder voneinander weg bewegt werden müssen.

Wird eine Geste auf zwei Hände aufgeteilt, kann dies auch die Geschwindigkeit erhöhen, mit der eine Aufgabe gelöst wird. Zusätzlich wird auch das Arbeitsgedächtnis entlastet, da der Einsatz der zweiten Hand als Stütze dient und die Aufgabe nun nicht mehr allein mental abgebildet werden muss [21].

Trotz der Vorteile wurde in anderen Studien festgestellt, dass Benutzer eher auf einhändige Gesten zurückgreifen [15, 31]. Wenn Benutzer auf beide Hände zurückgreifen, führen sie meistens symmetrische Bewegungen aus, anstatt sich zu überlegen, wie ihre nicht dominante Hand die dominante unterstützen könnte. Offenbar bevorzugen Benutzer einfache Gesten, die mit einer Hand ausgeführt werden können. Dies hat auch den Vorteil, dass weniger Oberfläche durch den eigenen Körper verdeckt wird.

Zusammenarbeit mehrerer Benutzer

Bei fest installierten Multi-Touch-Geräten muss nicht mehr auf Platzeinsparung und Akkuleistung hin optimiert werden und die Vorteile einer größeren Oberfläche können ausgenutzt werden. Dies vereinfacht nicht nur den Einsatz beider Hände, sondern ermöglicht auch ein gleichzeitiges Zusammenarbeiten von mehreren Benutzern am selben Gerät.

Denn trotz der Möglichkeiten, von z.B. Videokonferenzen, ist es für Benutzer wichtig von Angesicht zu Angesicht miteinander zu arbeiten [27]. Multi-Touch-Oberflächen bieten hierzu einen guten Ansatz dafür, da keine externen Eingabegeräte, wie Maus und Tastatur geteilt werden müssen. Des Weiteren besitzt die Multi-Touch-Technologie eine geringere Hemmschwelle für Menschen, die sonst wenig mit Computern arbeiten. Das Bedienkonzept kann durch Nachahmen der Handbewegungen erlernt werden, so dass auch Benutzer, die geringe Computer-Affinität besitzen, leicht in den Arbeitsprozess auf der Multi-Touch-Oberfläche mit eingebunden werden.

Dabei ergeben sich aber auch einige Besonderheiten, verglichen mit einem Einzelarbeitsplatz. So verschärft sich das Verdeckungsproblem, wenn mehrere Benutzer mit der Oberfläche arbeiten [29]. Vor allem, wenn es die Arme der anderen sind, über die man kein eigenes Körperbewusstsein besitzt. Zudem verringert sich mit der Anzahl der beteiligten Personen der eigene Anteil an der Oberfläche, sofern jeder gleichberechtigt ist. Arbeitet man gemeinsam oder in Konkurrenz zu einander, bietet es sich an, zwischen öffentlichem und privatem Bereich zu unterscheiden. So kann jeder an seiner Lösung arbeiten und diese dann in der Mitte, im öffentlichen Bereich, der Allgemeinheit zur Verfügung stellen. Hier ergibt sich ein weiteres Problem, wenn sich Personen gegenüberstehen, nämlich das der Ausrichtung und der damit verbundenen Lesbarkeit von Ergebnissen. Mögliche Lösungen wären, diese für jeden nach Bedarf drehbar zu machen, sie (auf Kommando) zu spiegeln, oder Elemente auf der Oberfläche zu mehrmals zu kopieren [29, 27].

Wichtig ist auch, dass jeder unabhängig und parallel arbeiten kann. So darf beispielsweise ein Tipp eines Benutzers einen anderen nicht aus dem Menü werfen [29]. Beim Diamond-Touch aus dem Jahre 2001 war es möglich Tipps Benutzern zuzuordnen [11]. Aber auch Begrenzungen für den Umkreis, in denen Gesten stattfinden können sind eine Möglichkeit. Als wichtiges Beispiel dafür wurde bereits der private Arbeitsbereich genannt.

2 Grundlagen

Aber auch hygienische Bedenken spielen für manche Benutzer eine Rolle, wenn andere Menschen mit ihnen die gleiche Multi-Touch-Oberfläche berühren. Ebenso unangenehm für manche Benutzer ist die unabsichtliche Berührung fremder Hände.

Insgesamt unterstützen die technischen Möglichkeiten die Arbeit im Team. Es ist allerdings darauf zu achten, dass die Technik im Hintergrund bleibt und nicht selbst zum Problem wird. Interaktionen müssen nahtlos ineinander übergehen [27].

Besonderheit Multi-Touch-Wand

Multi-Touch-Wände unterscheiden sich in einigen Punkten grundlegend von anderen Multi-Touch-Geräten, zum einen aufgrund ihrer Größe, vor allem aber auch durch ihre vertikale Ausrichtung [29]. Üblicherweise arbeitet man im Stehen und muss, um alle Bereiche der Wand zu erreichen, auch laufen. Durch die fehlende Handauflage während der Bedienung, wird die Ermüdung der Arme durch die Schwerkraft beschleunigt. Insgesamt werden also andere Muskelgruppen aktiv, als bei einem Tablet oder einem Multi-Touch-Tisch. Nutzt man die ganze Fläche der Multi-Touch-Wand aus, kommt man nicht umhin, immer wieder ein Stück zu gehen, um den Überblick nicht zu verlieren. Einen Lösungsansatz bieten sogenannte Zwischeneingabe-Geräte in der Größe von Smartphones oder Tablets, wodurch man wieder auf Distanz gehen kann oder aber auch intelligente Bedienkonzepte direkt an der Wand, die zumindest die Wege verkürzen und durch optische Zusammenfassungen je nach Auflösung der Oberfläche für Überblick sorgen können. Es stellte sich heraus, dass Wände für eine Zusammenarbeit von mehreren Personen weniger geeignet sind als Tische [27]. Dies liegt vor allem daran, dass vor einer Multi-Touch-Wand typischerweise ein Benutzer für eine gewisse Zeit eine Präsentationsrolle einnimmt, in der die anderen in eine Zuschauerrolle zurückfallen. Je nach Szenario kann dies jedoch auch erwünscht sein. Der Vorteil von Multi-Touch-Wänden ist sicherlich, dass beispielsweise große Prozesse, wie sie in der Praxis häufig anzutreffen sind, ohne oder mit bedeutend weniger Scrollen darstellbar sind.

2.2 Zeichensysteme und Gesten

Arbeitet man mit einer Multi-Touch-Oberfläche, gibt es zwei unterschiedliche Herangehensweisen sie mit Fingern zu bedienen. Zum einen über Menüs, wie man sie von herkömm-

lichen Programmen, die mit Maus und Tastatur bedient werden, kennt. Zum anderen über Gesten.

Während sich Menüstrukturen von Programmen im Laufe der Zeit in ihrem Aufbau angeglichen haben und aufgrund ihrer Beschriftung (meist durch hierarchische Ordnung unterstützt) auch relativ klar in ihren Wirkungen sind, gibt es nur sehr wenige Multi-Touch-Gesten, die ähnlich weit verbreitet und eindeutig sind. Diese wiederum lassen sich in Kategorien einteilen wie "physikalische Gesten", die virtuelle Objekte direkt manipulieren und "symbolische Gesten", die mehr im übertragenen Sinne in Beziehung zur Funktion stehen. Dabei gibt es verschiedene Abstufungen in wie weit dies der Fall ist. So kann eine Geste einen Sachverhalt bildlich nachzeichnen, sich auf bestimmte Konventionen beziehen oder auch vollkommen willkürlich sein.

Im Folgenden wird in Kapitel 2.2.1 die Gebärdensprache als Vorbild für Gesten-basierte Kommunikation auf ihre Übertragbarkeit auf Multi-Touch-Geräte untersucht, in Kapitel 2.2.2 wird die Bedeutung alltäglicher Symbole für die Erstellung von Bedienkonzepten untersucht und in Kapitel 2.2.3 wird das Apple Multi-Touch Gesture Dictionary als weitere Möglichkeit zur Konzeption von Multi-Touch-Programmen vorgestellt.

2.2.1 Gebärdensprache als Vorbild

Kommunikation findet bei uns zum größten Teil über Körpersprache, wie Mimik und Gestik statt, begleitet von gesprochener Sprache. Hinzu kommen die konkrete Situation, der Kulturkreis, in dem man sich gerade befindet und tagesbedingte Befindlichkeiten der Gesprächsteilnehmer. All dies scheint viel zu komplex für die Interpretation durch einen Computer. Dennoch werden aufgrund zahlreicher Forschungen Fortschritte auf diesem Gebiet gemacht. So ist man bereits heute in der Lage bei der Spracherkennung rudimentär Gefühlsregungen aus dem eindimensionalen Medium Schall heraus zu erkennen. Auch Gesichtserkennung und andere Anwendungen aus dem Bereich Computer Vision sind nichts völlig neues mehr. Konzentriert man sich alleine auf Gesten wird man schnell feststellen, dass im Alltag nur sehr allgemeine Gesten auftauchen, die für sich genommen keine große Aussagekraft haben. Kommunikation setzt sich bei Menschen aus vielen kleinen Einzelteilen zusammen. Dies merkt man vor allem bei Menschen, die sich der Gebärdensprache bedienen müssen. Werden Gesten strategisch entwickelt und ausgenutzt, lassen sich damit aber viele Defizite ausgleichen. Gebärden werden hierbei durch vier Parameter bestimmt: Handkonfiguration, Handorientierung, Bewegungsausführung und Ort der Bewegung. Also

2 Grundlagen

durchaus Aspekte, die bei Multi-Touch-Gesten ebenfalls eine Rolle spielen.

Während in der gesprochenen Sprache Informationen zwingend nur aufeinander folgend wiedergegeben werden können, profitiert die Gebärdensprache von einer impliziten Sichtbarmachung von Informationen durch die Art, wie die Gesten durchgeführt werden. Als Beispiel ließe sich die Information "ich - Buch - geben - dir - schnell" folgendermaßen zerlegen: Beginnt die Geste bei Person A und endet sie bei Person B, bekommt Person B das Buch von A. Wer wem etwas gibt, wird also durch die Bewegungsrichtung ersichtlich. Wird diese Bewegung schnell oder langsam durchgeführt wird ebenfalls sofort klar, was gemeint ist. Die Bewegung steht in direktem Verhältnis zur Semantik. Handelt es sich wie in diesem Fall um ein Buch, wird dieses imaginär hochgehalten [3].

Ein menschliches Gehirn welches darauf trainiert wurde, kann somit auch komplexere Anweisungen in Gestensprache entgegennehmen. Für einen Computer hingegen müssen drastische Vereinfachungen getroffen werden, wie das Projizieren auf zwei Dimensionen und die Analyse von Bewegung und Ausdehnung eines ovalen Fingerabdrucks allein. Mehr kann der Computer nicht auswerten. Mehr kann aber auch der Mensch nicht dazu beitragen. Da Gesten in der Vergangenheit nur im Kontext zwischenmenschlicher Kommunikation aufgetreten sind und das in einer eher unbewussten und allgemeinen Form, ist das Arbeiten mit Gesten auf Computern eine sehr individuelle, erforschende Angelegenheit. Es gibt keinen allgemeingültigen Konsens wie etwas mit dem Finger zu bedienen ist, auch nicht innerhalb einer Kulturgemeinschaft. Experimente [31, 20, 9, 13] versuchen zumindest in grobem Maße herauszufinden auf welchen größten Nenner man sich für allgemeine Aufgaben am ehesten einigen kann, indem vor allem die Benutzer selbst herangezogen werden passende Gesten zu finden. Hierbei ergeben sich aber auch Überschneidungen, so dass eine Geste für mehrere Funktionen in Frage kommt.

2.2.2 Symbole im Alltag

Im Alltag begegnen uns viele Kommunikationsformen, denen wir uns oft nicht mehr bewusst sind. Wer einen Führerschein besitzt und schon länger im Straßenverkehr unterwegs ist, verarbeitet Verkehrsschilder automatisch, ohne darüber explizit nachdenken zu müssen. Diese Symbole verhalten sich für eine bestimmte Gruppe von Menschen wie eine Sprache. Genauso werden regelmäßige Benutzer von Computern mit bestimmten Symbolen ihres Betriebssystems vertraut sein. Hier besteht also für die Gestaltung von Multi-Touch-Oberflächen ein möglicher Anknüpfungspunkt. Um mit einer zweidimensionalen Fläche mit

seinen Fingern kommunizieren zu können, kann man die Formen dieser Symbole, darunter zählen auch Vorgehensweisen, nachahmen. Zum Beispiel ein Fragezeichen, um die Hilfe aufzurufen. Dem Auge bzw. Gehirn sind diese bereits vertraut und müssen nicht neu gelernt werden. Sie ergeben sich aus der Alltagserfahrung und unterliegen daher zusätzlich einem Konsens. In Studien wurde festgestellt, dass sich viele Gesten auf die Maus-Umgebung zurückführen ließen, auch wenn den Probanden völlig freie Hand bei der Bearbeitung der Testaufgaben gelassen wurde und keine Windows-spezifischen Elemente verwendet wurden [31, 15]. Als Beispiel seien hier der Papierkorb und Kontextmenüs genannt. Ebenso wie die Konvention eines Doppelklicks mit der Maus.

Es gibt Funktionen, die man in fast jeder Art von Programm wiederfindet, unabhängig von der darunterliegenden Technologie, wie z.B. Markieren, Editieren, Löschen, Scrollen, Verschieben oder Zoomen. Für letzteres hat sich die Pinch-Geste in Multi-Touch-Anwendungen zum Teil schon als Quasi-Standard durchgesetzt. Von diesen allgemeinen Funktionen unterscheiden sich spezifische Funktionen, die für bestimmte Programme zugeschnitten sind, wie beispielsweise die Modellierung von Prozessen. Doch nicht jede Funktion wird gleich oft eingesetzt. Dies muss also zwangsläufig Auswirkung auf die Auswahl der Gesten haben. Eine Funktion, die oft verwendet wird, muss schnell zugänglich sein. Das bedeutet für die Geste, dass sie schnell und einfach ausführbar sein muss und so eindeutig, dass es zu wenig Fehlinterpretationen kommen kann. Wird eine Funktion hingegen nur selten verwendet, kann die dazugehörige Geste auch komplexer sein.

2.2.3 Apples Multi-Touch Gesture Dictionary

Apple ist sich des Problems bewusst, dass es kaum einen Konsens darüber gibt, welche Geste für welche Funktion stehen soll. Weshalb sie eine komplett neue, eigenständige Gesten-Sprache für die Bedienung von Multi-Touch-Geräten entwickelt haben [12]. Wie jede andere Sprache auch, muss diese vom Benutzer erst erlernt werden. In diesem Fall gibt es ähnlich wie beim Musizieren mit einer Gitarre bestimmte Grundstellungen der Finger, die sogenannten "Chords". Diese kann man nun nach oben, unten, links rechts und in die vier entsprechenden Diagonalen bewegen ("Motion"). Hinzu kommen Drehungen nach links und rechts, sowie eine Pinch- und Anti-Pinch-Bewegung und das Tippen auf der Stelle (s. Abbildung 2.1 und 2.2). Jeder Chord besitzt also 13 verschiedene Motions. Und jede Hand ist in der Lage 25 Chords zu erzeugen, denn jeder Finger kann unabhängig von einander gehoben werden. Weitere Unterscheidungen ergeben sich durch den Abstand der

2 Grundlagen

einzelnen Finger zueinander und ob sie flach aufliegen oder vertikal. Damit ergeben sich über 300 unterscheidbare Kombinationen und im Zusammenspiel mit der zweiten Hand noch deutlich mehr. Diese müssen natürlich nicht alle mit Funktionalität belegt werden, aber es zeigt was theoretisch möglich wäre. Damit kommt man einer Gebärdensprache für die Multi-Touch-Oberfläche schon sehr nahe. Um die Gesten lernen zu können, wurde eine Programm entwickelt und patentiert, welches als Wörterbuch fungiert und in dem man Funktionen und Gesten nachschlagen kann [12]. Da das Programm im Hintergrund laufen kann, kann es auch aus einem anderen Programm heraus aufgerufen werden. Dies passiert zum Beispiel dann automatisch, wenn auf einen Chord nach einer gewissen Zeit keine Bewegung erfolgt. Der Benutzer hat dann die Möglichkeit nachzuschauen, welche Funktionen mit diesem Chord auslösbar sind. Es gibt auch Spiele, in denen man die Benutzung dieser Gesten trainieren kann. Aber auch selbst definierte Chord-Eintragungen in das Wörterbuch sind möglich.

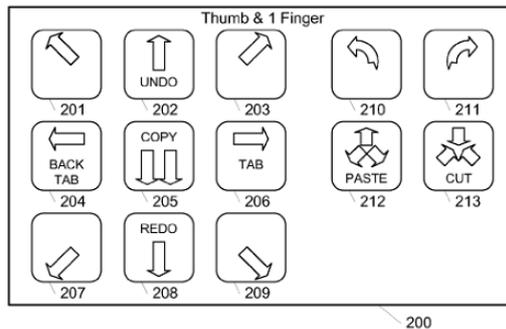


Abbildung 2.1: Apple-Chord am Beispiel mit einem Daumen und einem Finger [1]

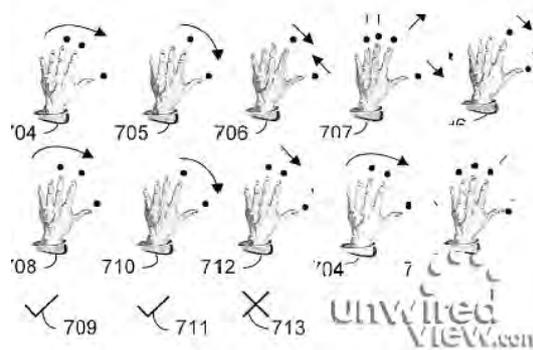


Abbildung 2.2: Chord und Motions visualisiert [8]

2.3 Zusammenfassung der Grundlagen

Als erstes wurden in Kapitel 2.1.1 die unterschiedlichen Benutzertypen vorgestellt. Diese waren: geübte bzw. ungeübte regelmäßige Benutzer und geübte bzw. ungeübte sporadische Benutzer. Jeder Benutzertyp priorisiert unterschiedliche Anforderungen an ein Programm. In Kapitel 2.1.2 gingen wir auf verschiedene Anwendungsszenarien ein und besprachen spezifische Problemstellungen der Multi-Touch-Technologie, wie die Präzision beim Tippen, die Verdeckung der Multi-Touch-Oberfläche während des Tippens, die Ermüdung der Gliedmaßen und die Arbeit im Team. In Kapitel 2.2 sprachen wir über die Rolle von Gebärdensprache und Alltagssymbolen für die Entwicklung von Multi-Touch-Gesten und über systematische Gestenentwicklung. Auf der Grundlage dieser Informationen werden in Kapitel 3 verschiedene Ansätze zur Modellierung von Prozessen mit Hilfe von Multi-Touch-Gesten entwickelt.

2 Grundlagen

3 Konzepte zur Prozessmodellierung auf Multi-Touch-Geräten

Ziel dieses Kapitels ist es Wissen über Multi-Touch-Technologie im Allgemeinen auf Business Process Management im Speziellen anzuwenden und so eine gebrauchstaugliche Bedienung für die Modellierung von Prozessabläufen auf Multi-Touch-Geräten wie dem Apple iPad zu finden. Um diese Kombination zu erforschen, werden im Folgenden, in Kapitel 3.1 und Kapitel 3.2, zwei konträre Ansätze zur Gesten-basierten Modellierung entwickelt. Anschließend wird in Kapitel 3.3 eine optimale Lösung aus den Vorzügen beider Ansätze hergeleitet.

Zugrunde gelegt wird hierbei ein blockorientierter Prozessgraph [26] mit einem festen Layout-Raster, d.h. der Benutzer hat keinen bzw. nur sehr beschränkt Einfluss auf die Positionierung der Prozesselemente, die von einem Layout-Algorithmus positioniert werden. Dies unterstützt eine korrekte Modellierung und verbessert dadurch die Verständlichkeit. Das feste Layout-Raster ermöglicht auch mehr Freiheiten bei der Gestaltung von grundsätzlichen Modellierungsfunktionen, da keine Gesten für eine manuelle Layout-Funktion besetzt werden. Sollte jedoch dem Benutzer die automatische Formatierung nicht zusagen, können trotzdem im gewissen Rahmen Anpassungen durch Verschieben einzelner Elemente (z.B. die Sortierung von Datenelementen) innerhalb des Layout-Rasters vorgenommen werden. Als Notation wird die Business Process Modeling Notation (BMPN) verwendet [2].

3.1 Entwicklung einer Gesten-basierten Modellierung

Im Folgenden wird ein Konzept vorgestellt, das die elementaren Modellierungsfunktionen auf Multi-Touch-Gesten abbildet. Die elementaren Modellierungsfunktionen zur Modellierung von Prozessmodellen sind:

3 Konzepte zur Prozessmodellierung auf Multi-Touch-Geräten

- F1: Aktivität einfügen
- F2: Element benennen
- F3: Verzweigungsblock einfügen
- F4: Verzweigung einfügen
- F5: Datenelement einfügen
- F6: Lese- und Schreib-Kanten setzen
- F7: Subprozess bilden und auflösen
- F8: Element löschen

sowie folgende Hilfsfunktionen:

- H1: Markieren
- H2: Rückgängig und Wiederholen
- H3: Verschieben und Kopieren
- H4: Hilfe aufrufen

3.1.1 Gesten-Set für die Modellierungsfunktionen

F1: Aktivität einfügen

Die Aktivität bildet die kleinste Einheit eines Prozesses, einen einzelnen Prozessschritt. Eine elementare und daher sehr häufig zu verwendende Funktion ist das Erstellen einer neuen Aktivität. Deshalb sollte sie möglichst einfach und schnell aufrufbar sein. Eine erste intuitive Herangehensweise ist die bildliche Repräsentation einer Aktivität, in etwa ein Rechteck, nachzuzeichnen (s. Abbildung 3.1 a). Das skizzenhafte Zeichnen bildet seit frühester Kindheit eine einfache und naheliegende Art der nonverbalen Ausdrucksform [10]. Man sieht etwas und ahmt es nach, um es zu beschreiben.

Flüssiger in der Durchführung wird die Bewegung, wenn diese zu einer Kreisform vereinfacht wird (s. Abbildung 3.1 b). Dies stellt zwar noch einen relativ direkten Bezug zwischen Geste und Funktion her, benötigt aber im Vergleich zur Komplexität der Aufgabe noch viel Zeit bei der Durchführung. Auch benötigen Rechteck und Kreis viel Platz. Hat der Benutzer weit heraus gezoomt, ist es schwierig die genaue Position zu bestimmen. Deshalb braucht es eine weitere Abstraktion und Verkürzung der Geste.

3.1 Entwicklung einer Gesten-basierten Modellierung

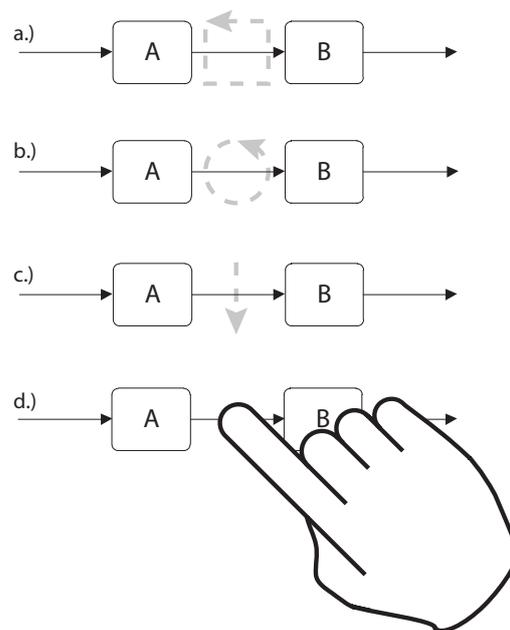


Abbildung 3.1: F1: Aktivität einfügen

Die Geste lässt sich so auf eine gerade Bewegung reduzieren, die die Kante an der Stelle kreuzt, an der die neue Aktivität eingefügt werden soll (s. Abbildung 3.1 c.). Diese Geste ist einfach und schnell, könnte aber den Nachteil haben mit einer Art Durchstreichen oder Ausschneiden der Kante verwechselt zu werden.

Reduziert man das Erstellen einer neuen Aktivität weiter auf ein Tippen oder Doppeltippen wie in Abb. 3.1 d.) zu sehen, gewinnt man ein Maximum an Geschwindigkeit. Man verliert dafür aber auch an Präzision, wenn man bedenkt, dass neue Aktivitäten auf dünnen Kanten eingefügt werden sollen. Mit dem weiter unten beschriebenen Mechanismus kann dies zwar vernachlässigt werden. Aber bei unbeabsichtigtem Berühren der Multi-Touch-Oberfläche kann dies den Prozess ungewollt oder gar unbewusst verändern. Auch besteht zwischen einem bloßen Tippen und dem Einfügen einer neuen Aktivität kein eindeutiger Bezug. Selbst wenn aus dem Kontext nichts anderes sinnvoll mit dieser Geste ausgelöst werden kann als eine neue Aktivität einzufügen, sollte man aus Konsistenzgründen darauf verzichten und den Tipp oder Doppeltipp für etwas anderes, einheitliches aufbewahren. Denn für gewöhnlich bringt man diese mit Markieren, Öffnen oder Editieren in Verbindung.

3 Konzepte zur Prozessmodellierung auf Multi-Touch-Geräten

Eine Bewegung besitzt zudem einen größeren Informationsgehalt als ein einzelner Punkt und daher ist die Strich-Geste weniger anfällig für unabsichtliches Auslösen einer Einfüge-Funktion. Da sie auch schnell auszuführen ist, soll diese für "F1: Aktivität einfügen" verwendet werden.

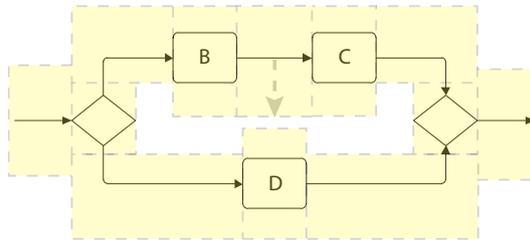


Abbildung 3.2: Erkennungszone des Prozessgraphen

Um den Fingern eine größere Zielfläche zu ermöglichen und auch Gesten dann noch eindeutig zuordnen zu können, wenn sie nicht exakt das gewünschte Element treffen, ist es sinnvoll den Prozessgraphen in Zonen zu unterteilen, in der jedes Element einen Sicherheitsrand erhält, der ungenutzt ist. Dabei ist darauf zu achten, dass sich die Zonen der Elemente nicht gegenseitig überlappen, um Gesten eindeutig den Elementen zuordnen zu können (s. Abbildung 3.2).

F2: Element benennen

Da Aktivitäten oder Datenelemente meist einen sprechenden Namen haben, ist es notwendig, dem Benutzer nach deren Erstellung anzubieten, diese zu benennen. Es muss aber auch möglich sein zu einem späteren Zeitpunkt den Namen zu verändern. Dazu benutzt man einen Doppeltipp, um anzuzeigen, dass das Element auf das getippt wurde, editiert werden soll. Anschließend öffnet sich eine große Schreibfläche (s. Abbildung 3.3), in die der Benutzer mit dem Finger hineinschreiben kann. Eine Texterkennungssoftware erkennt die Handschrift in Echtzeit, damit der Benutzer kontrollieren kann, ob der Name richtig erkannt wurde. Falls nicht, streicht er den betreffenden Teil durch und versucht es erneut. Reicht der vorgesehene Platz nicht aus, scrollt der Benutzer – mit einer Wischbewegung mit mindestens zwei Fingern gegen die Leserichtung – nach rechts. Um die Eingabe zu bestätigen, tippt er auf die Aktivität mit dem neuen Namen. Nicht benannte Aktivitäten werden automatisch durchnummeriert.

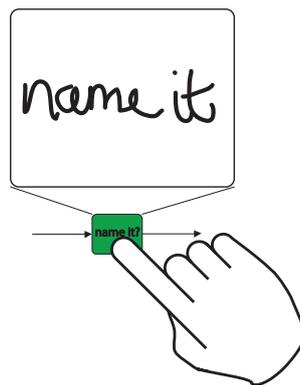


Abbildung 3.3: F2: Element benennen

F3: Verzweigungsblock einfügen

Ein Verzweigungsblock bildet eine Struktur, die mit genau einer Kontrollflusskante erreicht wird und ebenfalls über eine Kontrollflusskante verlassen wird. Dies ist ein so genannter Kontrollblock [26]. Die beiden Begrenzer dieses Kontrollblockes bilden die Verzweigungsknoten. Innerhalb dieses Kontrollblockes gehen von den Verzweigungsknoten oft mehrere Kanten aus, sogenannte Verzweigungen, die den weiteren Ablauf des Prozesses beschreiben. Je nach Art des Verzweigungsknoten handelt es sich hierbei z.B. um sich gegenseitig ausschließende XOR-Verzweigungen oder um parallele AND-Verzweigungen (d.h. es werden alle Verzweigungen durchlaufen).

Durch die beiden Verzweigungsknoten ergeben sich zwei Möglichkeiten Verzweigungsblöcke einzufügen: entweder muss Start- und Endpunkt einer Verzweigung festgelegt werden, die dann den umschließenden Block markieren oder es wird ein leerer Verzweigungsblock an einer Stelle mit Start- und Endpunkt zusammen liegend eingefügt. Im Folgenden wird versucht eine intuitive Geste für diese Funktion herzuleiten.

Es bieten sich verschiedene Symboliken an, die leicht mit dem Finger nachgezeichnet werden können. Man könnte zum Beispiel das "Plus" und "X" aus den BPMN-Gateways entlehnen, um damit AND- und XOR-Verzweigungsknoten zu realisieren. Dies hätte den Vorteil, dass die Symbole verwendet werden, die bereits aus der Prozess-Modellierung bekannt sind und somit auch intuitiv vom Benutzer verwendet werden können. Ein Nachteil jedoch ist, dass diese beiden Gesten bzw. Symboliken sich sehr ähnlich sind, was gerade bei mobilen Geräten, die Haltungswchsel der Geräte leicht ermöglichen, zu Verwechslungen

3 Konzepte zur Prozessmodellierung auf Multi-Touch-Geräten

führen kann. Außerdem könnte man ein "X" als Multi-Touch-Geste auch mit Löschen assoziieren und ein "Plus" mit Vergrößern. Nicht unerwähnt soll bleiben, dass beide Gesten ein Absetzen des Fingers erfordern und dadurch die Erkennung erschwert wird. Die Betrachtung zeigt, dass diese beiden Symboliken nicht für den Einsatz geeignet sind.

Eine alternative Lösung für die AND-Verzweigung ist das Nachzeichnen des kaufmännischen "&". Man könnte es in einer Bewegung ohne Absetzen durchziehen; dennoch dauert die Geste eine gewisse Zeit. Eine entsprechende Analogie für die XOR-Verzweigung ist nicht vorhanden.

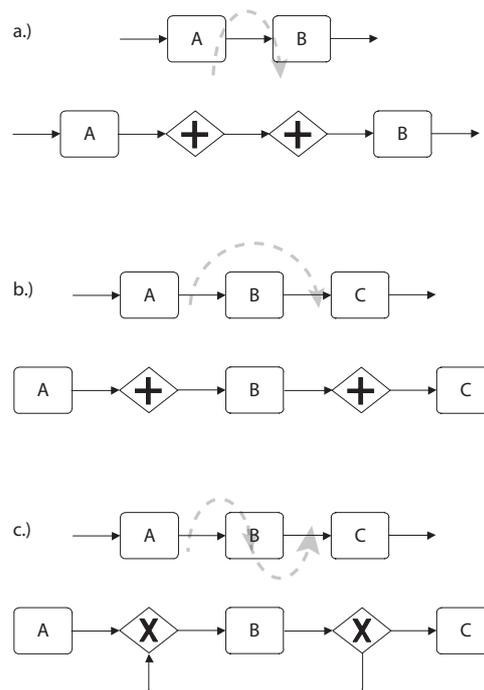


Abbildung 3.4: F3: Verzweigungsblock einfügen

Abhilfe könnten die Symbole für Konjunktion (\wedge) und Disjunktion (\vee) aus der Informatik oder Mathematik (\cap , \cup) schaffen. Man zeichnet die Formen dieser logischen Operatoren, also in etwa Halbkreise, mit dem Finger nach. An den beiden Enden, die die Kanten schneiden, werden die Verzweigungsknoten eingefügt. Möchte man die Verzweigungsknoten zusammen liegend einfügen, formt man den Halbkreis sehr eng, so dass er zwischen zwei Elemente passt (s. Abbildung 3.4 a). Möchte man bereits bestehende Elemente in den

3.1 Entwicklung einer Gesten-basierten Modellierung

Verzweigungsblock mit einschließen, spreizt man den Halbkreis entsprechend, so dass die Geste vor den Elementen beginnt und dahinter abschließt (s. Abbildung 3.4 b). Bei den Vorschlägen aus den beiden vorherigen Absätzen müsste man, um einen umgebenden Verzweigungsblock zu erhalten, entweder mit zwei Fingern gleichzeitig an Start- und Endpunkt die entsprechenden Gesten ausführen oder den Umweg gehen, die äußeren Elemente nach innen zu verschieben.

Für einen Schleifenblock werden die Gesten der AND- und XOR-Verzweigungsblöcke hintereinander gehängt, so dass ein liegendes "S" entsteht (s. Abbildung 3.4 c). Der Benutzer schneidet den Kontrollfluss an der Stelle, an der er einen Schleifenknoten eingefügen möchte. Er zeichnet einen Halbkreis und schneidet den Kontrollfluss ein zweites mal. Der Benutzer setzt den Finger nun aber nicht wie bei einem AND-Verzweigungsblock ab, sondern schließt in einer durchgehenden Bewegung den nächsten Halbkreis auf der anderen Seite des Kontrollflusses an, so dass eine S-förmige Figur entsteht. Sobald der zweite Halbkreis den Kontrollfluss wieder schneidet, lässt der Benutzer los und es wird an dieser Stelle der zweite Schleifenknoten eingefügt. Der Benutzer schneidet auf diese Weise den Kontrollfluss an drei Stellen und an den äußeren zwei werden die Schleifenknoten eingefügt. Dieses Vorgehen funktioniert sowohl für leere Schleifenblöcke, als auch für umgebende Schleifenblöcke, indem die Geste entsprechend gestreckt wird.

Hat man einen leeren Verzweigungsblock realisiert und will nachträglich die umgebenden Aktivitäten darin einschließen, schiebt man die äußeren Aktivitäten einzeln oder im Block in den Zwischenraum des neuen Verzweigungsblockes oder direkt auf die Verzweigungsknoten. Dazu können die Hilfsfunktionen "H3: Verschieben und Kopieren" in Kapitel 3.1.2 verwendet werden.

F4: Verzweigung einfügen

Ein Verzweigungsblock besteht anfangs nur aus einer einzigen Verzweigung. Möchte man z.B. parallele Abläufe modellieren, muss man dem Verzweigungsblock zunächst eine leere Kante, also eine neue Verzweigung, hinzufügen, auf der dann weitere Elemente platziert werden können.

Hierzu zeichnet der Benutzer mit seinem Finger eine Linie von dem öffnenden Verzweigungsknoten zum schließenden Verzweigungsknoten. Dieses Vorgehen gleicht dem Drag & Drop-Prinzip (s. Abbildung 3.5 a), nur dass in diesem Fall nichts verschoben, sondern

3 Konzepte zur Prozessmodellierung auf Multi-Touch-Geräten

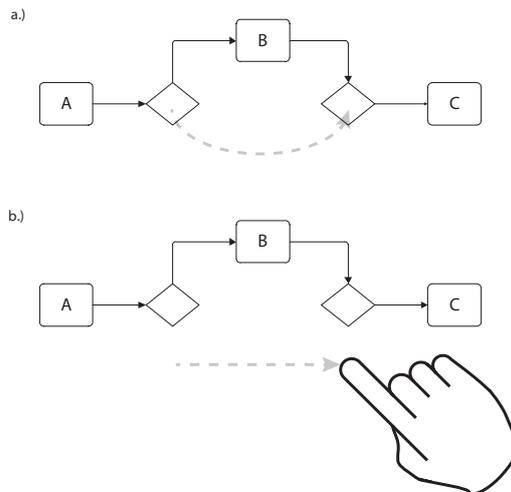


Abbildung 3.5: F4: Verzweigung einfügen

eine Linie gezogen wird. Dabei ist gerade wenn man die Geste in der Zone und nicht direkt auf den Element ausführt, darauf zu achten eine waagerechte Linie zu ziehen, da sonst Verwechslungsgefahr mit anderen Gesten, wie das Einfügen eines Verzweigungsblockes, bestehen kann (s. Abbildung 3.5 b).

F5: Datenelement einfügen

Da Datenelemente nur im Zusammenhang mit Aktivitäten verwendet werden, bietet es sich an, deren Erstellen direkt mit einer Interaktion auf einer Aktivität zu verknüpfen. Hierbei wird der Finger auf eine Aktivität gelegt und von dieser dann eine Linie nach oben über eine gewisse Begrenzung gezogen. Sobald diese (gestrichelte) Begrenzung erreicht wird, kann der Benutzer seinen Finger wieder anheben, um das Datenelement zu erzeugen, welches gleichzeitig mit einer Schreib-Kante von der ausgehenden Aktivität versorgt wird (s. Abbildung 3.6). Man kann aber auch oberhalb der Begrenzung starten und den Finger auf eine Aktivität ziehen und erhält ebenfalls ein neues Datenelement. Diesmal allerdings verbunden mit einer Lese-Kante zur entsprechenden Aktivität. Die Reihenfolge bzw. Richtung entscheidet folglich darüber, welche Art von Kante zwischen Datenelement und Aktivität gesetzt wird. Will man ein alleinstehendes Datenelement erstellen, ohne Kante, führt man die gleiche Geste, wie für das Erstellen einer neuer Aktivität aus, also einen senkrechten Strich, nur oberhalb besprochener Begrenzung.

3.1 Entwicklung einer Gesten-basierten Modellierung

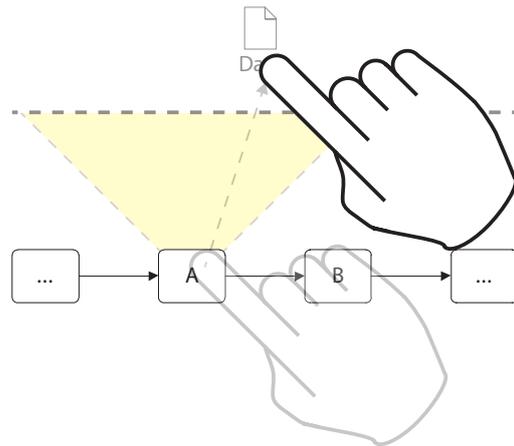


Abbildung 3.6: F5: Datenelement einfügen

F6: Lese- und Schreib-Kanten setzen

Nachdem nun Gesten für die Erstellung von Datenelementen vorgestellt wurden, werden nun Gesten eingeführt um bestehende Datenelemente mit bestehenden Aktivitäten zu verbinden. Um neue Lese- und Schreib-Kanten zu erstellen, bedient man sich der gleichen Geste wie bei der Erstellung von Verzweigungen für bestehende Verzweigungsblöcke. Man zieht also eine Linie von einem Datenelement zu einer Aktivität und erhält auf diese Weise eine Lese-Kante; umgekehrt eine Schreib-Kante. Da der Kontext der Verwendung klar getrennt ist, kann die Geste doppelt verwendet werden, ohne dass es zu Verwechslungen bzw. Mehrdeutigkeiten kommt.

F7: Subprozess bilden und auflösen

Ein weiteres Konstrukt, das bei der Modellierung von Prozessen verwendet wird, bildet der Subprozess. Ein Subprozess ist eine Aktivität, die selbst wiederum für einen Teilprozess steht. Damit ist der Subprozess ein wichtiges Instrument zur Bestimmung des Abstraktionsniveaus eines Prozesses. Bei vielen Aktivitäten, z.B. ab 50, helfen Subprozesse die Übersichtlichkeit zu behalten. Hierfür berührt man zwei Elemente, die zusammen einen gültigen Kontrollblock umschließen und bewegt die Finger aufeinander zu, bis einem angeboten wird, die eingeschlossenen Elemente zu einem Subprozess zusammenzufassen und lässt die Oberfläche los (s. Abbildung 3.7).

3 Konzepte zur Prozessmodellierung auf Multi-Touch-Geräten

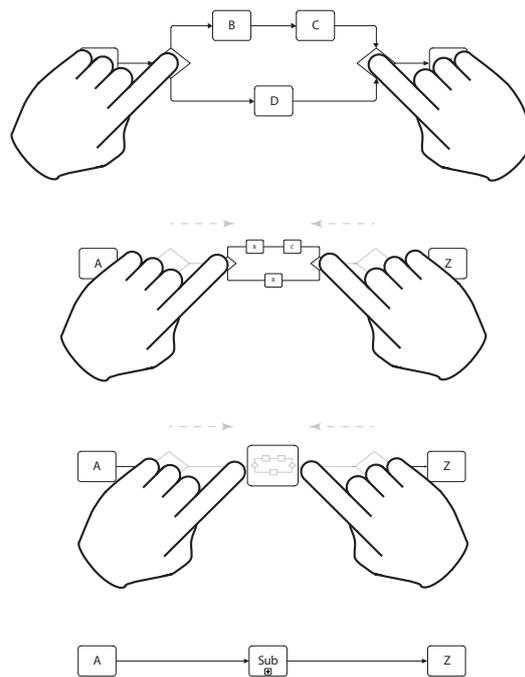


Abbildung 3.7: F7: Subprozess bilden

Umgekehrt kann man einen Subprozess durch das Auseinanderziehen wieder auflösen. Es ist aber auch möglich nur den eingeschlossenen Subprozess zu betrachten oder zu bearbeiten, ohne ihn aufzulösen. Dazu führt man eine Wisch-Geste nach rechts auf dem Subprozess aus. Der Subprozess klappt optisch auf und die inneren Elemente werden sichtbar. Es ist nun möglich mit der Pinch-Geste näher heranzuzoomen und die inneren Elemente zu betrachten oder zu bearbeiten. Hat man seine Arbeit beendet, klappt man den Subprozess, durch die entgegengesetzte Wischbewegung (nach links) auf den sichtbaren Rand des Subprozess, wieder zu.

F8: Element Löschen

Um Aktivitäten und andere Modellelemente zu löschen, bietet es sich an in Analogie zur realen Welt, diese Elemente durchzustreichen. Die Möglichkeiten reichen dabei von einem einfachen Querstrich, über ein Kreuz bis hin zu Zick-Zack-Linien (s. Abbildung 3.8). Diese unterscheiden sich allerdings in Dauer und Komplexität der Durchführung. Abbildung 3.8 a.) zeigt eine Möglichkeit des Durchstreichens wie sie auch auf einem Blatt Papier Anwendung

3.1 Entwicklung einer Gesten-basierten Modellierung

findet und daher naheliegend ist. In Abbildung 3.8 b.) wird die Aktivität mit einem Kreuz durchgestrichen, was etwas abstrakter ist, aber ebenfalls der Intuition entspricht. Allerdings ist hier ein Absetzen des Fingers erforderlich, was die Geste komplexer macht. Abbildung 3.8 c.) zeigt eine Geste, welche von der Form her an ein Verbotsschild erinnern soll, aber in einer durchgängigen Bewegung durchgeführt werden kann. Abbildung 3.8 d.) schließlich verkürzt alle voran gegangenen Lösungen zu einem diagonalen Strich über die Aktivität, so dass die Geste schnell und einfach durchzuführen ist. Das Löschen einer Aktivität stellt gewissermaßen das Gegenteil zur Funktion "F1: Aktivität einfügen" dar, was sich auch in der Gegenbewegung dieser Geste widerspiegelt.

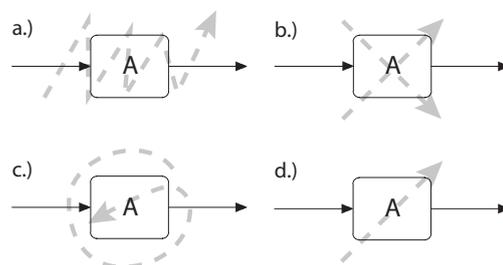


Abbildung 3.8: F8: Element löschen

Da das Löschen von Elementen eine kritische Funktion darstellt, ist es insbesondere bei der Geste aus Abbildung 3.8 d.), die schnell durchzuführen ist, wichtig, dass der Benutzer eine Sicherheitsabfrage bestätigen muss, bevor das Element gelöscht wird. Bei den ersten drei Gesten (Abbildung 3.8 a, b und c) spielt das aufgrund der Länge der Gesten eine geringere Rolle. Eine Sicherheitsabfrage würde sogar stören, da der Benutzer für diese Gesten schon einen gewissen Aufwand treiben muss, und somit seine Löschabsicht bereits in der Durchführung der Geste zum Ausdruck bringt.

Das Löschen einer Aktivität durch ein Herausziehen aus dem Prozessgraphen würde in Konflikt mit den Funktionen "F4 Verzweigung einfügen" und "F6: Lese- und Schreib-Kanten setzen" geraten, weshalb für das Löschen ein diagonaler Strich über das Element, wie er in Abbildung 3.8 d.) zu sehen ist, verwendet wird.

3.1.2 Gesten-basierte Hilfsfunktionen

Neben den grundlegenden Funktionen F1 bis F8, die für die Modellierung von Prozessen notwendig sind, gibt es Hilfsfunktionen, die das Modellieren erleichtern und aus Sicht einer benutzerfreundlichen Oberfläche ebenso notwendig sind. Die Hilfsfunktionen H1 bis H4 werden im Folgenden besprochen:

H1: Markieren

H2: Rückgängig und Wiederholen

H3: Verschieben und Kopieren

H4: Hilfe aufrufen

H1: Markieren

Zum Markieren eines einzelnen Elements mit Hilfe einer Geste reicht ein einfacher Tipp auf das Element. Zum Markieren von mehreren Elementen auf einmal umkreist man mit dem Finger die entsprechenden Elemente. Dieses Vorgehen ist sehr intuitiv, da ein Kreis etwas einschließt und es so von anderen Dingen hervorhebt. Man kennt es vom Arbeiten mit Stift und Papier.

H2: Rückgängig und Wiederholen

Die Rückgängig-Hilfsfunktion macht die letzte(n) Funktion(en) rückgängig. Ein erster Ansatz etwas rückgängig zu machen, wäre die Geste, die man rückgängig machen möchte, in umgekehrter Reihenfolge nochmals durchzuführen. Wenn also zum Beispiel mit einem senkrechten Strich von oben nach unten eine neue Aktivität erstellt wurde, kann dies rückgängig gemacht werden, indem man einen Strich von unten nach oben zieht. Dadurch muss sich aber der Benutzer, je nachdem wie weit er zurück will, die Reihenfolge zahlreicher Gesten, die er ausgeführt hat merken und umgekehrt ausführen. Weshalb dieser Weg vom Benutzer nicht akzeptiert wird.

Es gibt zumindest in der österreichischen Gebärdensprache eine Geste oder eine Gebärde die Rückgängig im Sinne des Computers meint [5]. Grob beschrieben vollziehen beide Hände symmetrisch mit gestreckten und geschlossenen Zeige- und Mittelfingern eine

3.1 Entwicklung einer Gesten-basierten Modellierung

schwungvolle Drehung nach rechts vom Körper weg, so dass von vorne betrachtet am Anfang der Handrücken zu sehen ist und hinterher die Handinnenfläche. Auf der beiliegenden CD befindet sich zur besseren Veranschaulichung ein Video davon.

Dies würde sogar ungefähr der Geste entsprechen, die im MacOS Lion von Apple zum Einsatz kommt: mit zwei Fingern horizontal nach rechts ziehen. Diese Geste wird in den Gesten-basierten Ansatz übernommen. Auf diese Weise lassen sich auch schnell viele Funktionen hintereinander rückgängig machen. Die Wiederholen-Geste verläuft entsprechend umgekehrt.

H3: Verschieben und Kopieren

Möchte man Änderungen in der Reihenfolge der Elemente vornehmen, etwa weil sich Abläufe geändert haben, ist es nützlich, wenn man diese als Ganzes verschieben kann, anstatt sie zu löschen und an anderer Stelle neu einfügen zu müssen. Um ein Element von einer Kante zu lösen und zu verschieben, führt man einen Doppeltipp darauf aus, belässt den Finger beim zweiten Kontakt allerdings auf der Multi-Touch-Oberfläche. Das Element hebt sich sichtbar durch einen kleinen Schatten vom Prozessgraphen und kann nun per Drag & Drop an die gewünschte Stelle geschoben werden (s. Abbildung 3.9 a). Sobald das Element losgelassen wird, rastet es wieder in den Prozessgraphen ein (s. Abbildung 3.9 b).

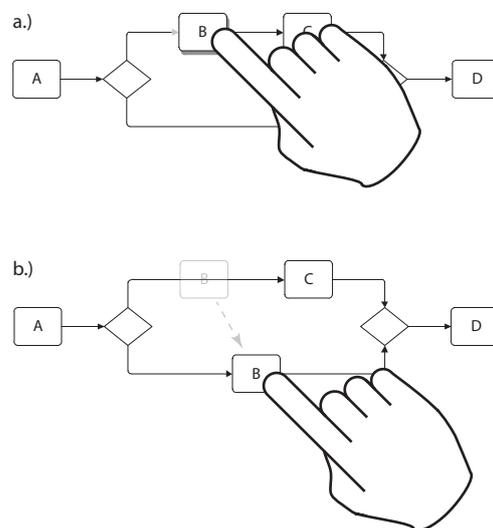


Abbildung 3.9: H3: Verschieben und Kopieren

3 Konzepte zur Prozessmodellierung auf Multi-Touch-Geräten

Eine seltener verwendete Funktion als das Verschieben ist das Kopieren von Prozessfragmenten eines Prozessgraphen. Betrachtet man beide Funktionen als verwandt ist eine Erweiterung der Verschieben-Geste für die Entwicklung eines Kopiervorgangs denkbar. Um ein Element zu kopieren, führt man die gleiche Geste wie beim Verschieben mit einem Tipp mehr durch. Konkret bedeutet dies, man tippt nun drei statt zwei mal auf das gewünschte Element, wobei der dritte Tipp auf der Multi-Touch-Oberfläche gehalten wird. Dies passt von der Analogie insofern zusammen, als dass auch im Programm durch das Kopieren nun ein Mehr entsteht.

Ein alternativer Ansatz für das Kopieren erlaubt das Absetzen des Fingers während des Kopiervorgangs, so dass das Element auch an einer weiter entfernten Stelle bequem eingefügt werden kann. Hierzu setzt man mit der Außenseite des Fingers am Element an und rollt ihn darüber ab, so als würde man seinen Fingerabdruck auftragen. Eine Rückmeldung erfolgt oberhalb des Fingers über ein kleines Symbol in Form einer grünen Lampe aus einem Kopiergerät, die gerade etwas abscannt. Der Benutzer weiß dadurch, dass seine Geste richtig erkannt und das Element in die Zwischenablage kopiert wurde.

Verfolgt man den ikonographischen, animierten Ansatz weiter und auch die Möglichkeit das Verschieben über ein Ausschneiden und Einfügen zu realisieren, ergibt sich beispielsweise folgendes: Man berührt das auszuschneidende Element und wackelt ohne loszulassen mit dem Finger etwas hin und her. Das Wackeln löst eine Animation aus, die den Finger an einen Staubsauger erinnern lässt. Das Element wird immer kleiner, bis es vom Finger vollständig aufgesogen wurde. Später lässt man das Element auf die gleiche Weise an der gewünschten Position wieder erscheinen. Eine weitere Lösung das Element wie mit einer Pinzette aufzunehmen, wäre entweder zu umständlich oder würde mit der Pinch-Geste kollidieren. Außerdem würde es zwei Finger benötigen.

Für die Übernahme in ein lauffähiges Programm spricht aber aufgrund der Nähe zur herkömmlichen Bedienung von Computern die Variante mit den Tipps.

H4: Hilfe aufrufen

Für den Fall, dass sich der Benutzer nicht mehr auf der Oberfläche zurechtfindet, nicht mehr weiß wie eine bestimmte Geste durchgeführt wird oder sonst eine Frage hat, bedarf es einer Hilfe. Zum einen wird diese explizit über eine symbolische Geste aufgerufen, etwa ein gezeichnetes Fragezeichen oder ein Kreuz, wie bei der "ersten Hilfe"; beide könnten auf

3.1 Entwicklung einer Gesten-basierten Modellierung

spezifische Elemente angewandt werden, um so kontextabhängig weiter zu helfen. Zum anderen wird die Hilfe aber auch implizit nach mehreren unerkannten Gesten oder durch unkoordinierte, nervöse Tipps in einem gewissen Umfang automatisch ausgelöst.

3.1.3 Erlernen der Gesten-basierten Prozessmodellierung

Da einem neuen Benutzer die Gesten zu Beginn nicht bekannt sein können und z.B. ein Suchen der Funktionalität in Menüs nicht möglich ist, bedarf es einer initialen Einführung in die Bedienung. Hierfür absolviert der Benutzer eine kurze Übung nach dem Nachahmungsprinzip. Anhand eines zu modellierenden Beispielprozesses werden dem Benutzer nacheinander die Gesten vorgeführt und erläutert, die nötig sind, um den Prozess zu vervollständigen. Dabei werden alle Funktionen mindestens einmal abgerufen. Sobald dem Benutzer ein Bewegungsablauf klar ist, soll er anschließend selbst die Geste ausführen. Wurde sie erkannt, wird die entsprechende Funktion ausgelöst und nach dem Erfolgserlebnis zur nächsten Übungs-Einheit übergegangen. Es wird dann die nächste zu übende Geste vorgeführt, die der Benutzer wiederholen soll. Abschließend kann er sich noch einmal selbst testen, indem er einen weiteren Prozess ohne fremde Hilfe vervollständigt.

3.1.4 Bewertung des Gesten-basierten Ansatzes

Der Gesten-basierte Ansatz setzt trotz sehr einfach gehaltener Gesten, die zum größten Teil mit einer Hand und ohne abzusetzen durchführbar sind, ein Lernen eben dieser voraus. Es wird einem Anfänger nicht leicht fallen sich all diese verschiedenen Gesten initial zu merken, selbst wenn versucht wurde, sie mit Semantik und Symbolcharakter (zum Teil aus Fachgebieten) anzureichern. Insgesamt ist es auch schwierig zu entscheiden, welche Geste diese Kriterien am ehesten erfüllt, da es oft sehr gleichwertige Lösungen sind. Die zusätzliche kognitive Belastung, sich mehr auf die Bedienung als die eigentliche Arbeitsaufgabe konzentrieren zu müssen, wird daher zu Beginn zu einer Verlangsamung und Leistungsabnahme beim Modellieren führen. Um zumindest einen ersten Anhaltspunkt, eine erste Rückmeldung zu liefern, wird vorgeschlagen, die Geste, die der Benutzer aktuell ausführt, mit einer Art Schweif zu versehen. Dadurch sieht der Benutzer, dass das Programm reagiert und auf was es reagiert. So kann auch die aufgrund der hohen Bedienungsfreiheit vorhandene Fehleranfälligkeit bzw. Fehlinterpretations-Wahrscheinlichkeit von Gesten durch den Benutzer leichter ausgeglichen werden. Einen weiteren Kritikpunkt,

der bei Multi-Touch-Programmen nahezu inhärent ist, bilden die langen Wege, die auf der Oberfläche zurückgelegt werden. Vor allem bei größeren Bildschirmen, wie Whiteboards wird ein längeres Arbeiten dadurch unzumutbar. Dennoch bietet dieser Ansatz den direktesten Zugang zur Modellierung ohne Zwischeneingabe-Geräte oder extra benötigte Werkzeuge. Das heißt, mit etwas Übung lassen sich mit der Gesten-basierten Modellierung schnell und sauber Prozesse skizzieren. Für regelmäßige Benutzer (s. Kapitel 2.1.1) von Prozessmodellierungen lohnt es sich den Einarbeitungsaufwand zu Beginn in Kauf zu nehmen.

3.2 Entwicklung einer menügeführten Modellierung

Nachdem im vorherigen Abschnitt ein Gesten-basiertes Set für die Modellierung entwickelt und besprochen wurde, folgt an dieser Stelle der konträre Ansatz mit der menügeführten Modellierung. Da Menüs auf unterschiedliche Weisen realisiert werden können, werden hierbei zwei verschiedene Varianten vorgestellt. Zum einen eine Drag & Drop-Menüleiste (s. Kapitel 3.2.1) und zum anderen ein Menü, welches vollständig über Schieberegler (s. Kapitel 3.2.2) bedient wird. Allerdings wird diesmal nicht jede Funktion einzeln vorgestellt. Vielmehr soll die grundlegende Herangehensweise vorgestellt werden und die jeweiligen Modellierungsfunktionen weitestgehend analog zum Schema aufgerufen werden können. Am Ende in Kapitel 3.2.3 werden die Vor- und Nachteile des menügeführten Ansatzes gegenüber des Gesten-basierten diskutiert.

3.2.1 Drag & Drop-Menüleiste

Die einfachste Variante ein Menü für Multi-Touch-Oberflächen umzusetzen, ist die Übernahme der Konzepte von herkömmlichen Programmen mit graphischer Oberfläche. Der einzige Unterschied hierbei ist, dass man nun bis zu zehn Finger gleichzeitig als Person einsetzen kann, diese allerdings mehr Platz brauchen als Mauszeiger und einem die Sicht auf die Oberfläche versperren. Das heißt ein Menü für Multi-Touch-Oberflächen kann keine exakte Kopie der Menüs herkömmlicher Programme sein. Es muss an diese Umstände entsprechend angepasst werden. Es bietet sich beispielsweise an, Funktionen direkt auf der Oberfläche zugreifbar zu machen, so dass in einer Art Baukastenprinzip, der Prozess zusammengesetzt werden kann. Dies kann auch vorgefertigte Muster einschließen, wie

3.2 Entwicklung einer menügeführten Modellierung

man sie aus ARIS Express kennt [6]. Hierarchien sollten flach gehalten werden, um direkten Zugang zu den benötigten Funktionen zu ermöglichen. Ebenfalls sollten die Funktionen so angeordnet sein, dass das Ziel bei der Auswahl nicht verdeckt und so aus den Augen verloren wird.

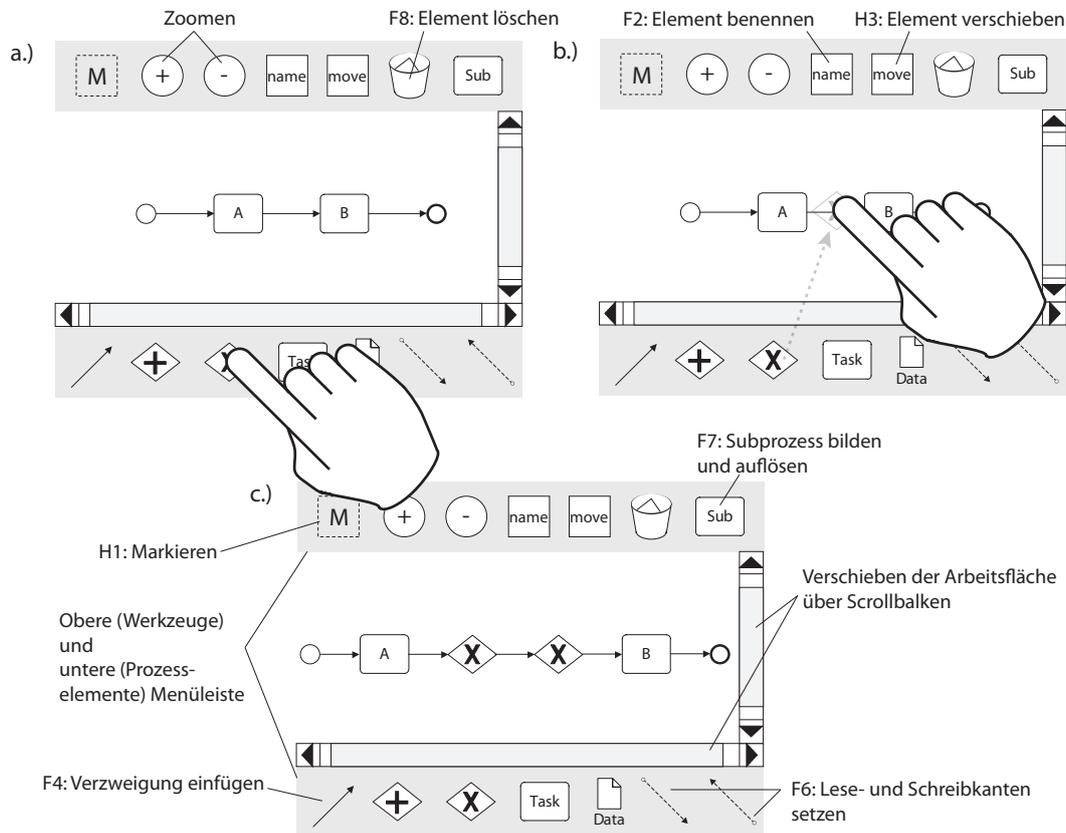


Abbildung 3.10: Drag & Drop-Modellierungskonzept

Im Wesentlichen funktioniert das vorgestellte Konzept (s. Abbildung 3.10) über zwei Menüleisten. In der unteren Menüleiste befindet sich ein Drag & Drop-Angebot der benötigten Elemente, aus denen ein Prozess aufgebaut ist. Der Benutzer zieht mit dem Finger ein Element aus der Menüleiste genau an die Stelle im Prozessgraphen, die er braucht und es wird automatisch an dieser Stelle eingefügt. In der oberen Menüleiste hingegen stehen Werkzeuge zur Verfügung, die mit einem einfachen Tipp aktiviert werden können. So verändert sich mit einem Tipp auf Markieren der Cursor, so dass um die gewünschten Elemente ein Rahmen gezogen werden kann, um sie zu markieren. Umgekehrt würde man,

3 Konzepte zur Prozessmodellierung auf Multi-Touch-Geräten

bei vorhandener Erfahrung mit verschiedenen Programmen, z.B. beim Löschen eines Elements erwarten, zuerst auf das zu löschende Element tippen zu müssen und anschließend erst auf das entsprechende Symbol (hier ein Papierkorb). Da es aber gleichwertig ist, ob zuerst auf das Element oder das Werkzeug getippt wird, wird aus Konsistenzgründen vorgeschlagen, immer zuerst auf das Werkzeug zu tippen und anschließend das Element auszuwählen, auf dem die Funktion durchgeführt werden soll.

Die Menüleisten haben den Vorteil, dass keine Gesten gelernt werden müssen. Nachteilig ist aber, dass dem Prozessgraphen kostbarer Platz weggenommen wird. Abbildung 3.10 veranschaulicht diesen Effekt. In der Praxis könnten die Symbole etwas verkleinert werden, um weitere Elemente unterbringen zu können und die Arbeitsfläche in der vertikalen Ausdehnung zu vergrößern. Dadurch verlängern sich jedoch gleichzeitig die Wege vom Menü zum Prozessgraphen, die zurückgelegt werden müssen. Sollte der Benutzer nicht mit dem Drag & Drop-Prinzip vertraut sein, so hat er dennoch eine Übersicht über alle Modellierungs-Funktionen, die ihm zu Verfügung stehen und dadurch einen Anreiz gezielter Probe zu tippen. Auf diese Weise erhält er eine konkretere Rückmeldung vom Programm. Beispielweise könnte ein einfacher Tipp auf ein Prozess-Element dazu führen, dass es schon einmal auf der Oberfläche erscheint. Dadurch wird deutlicher, dass der Benutzer dieses Element an seine gewünschte Stelle schieben kann, als es vielleicht direkt aus dem Menü heraus ersichtlich wäre.

3.2.2 Slider-Menüleiste

Die zweite Variante einer menügeführten Oberfläche versucht das Platzdefizit auszugleichen, das vor allem auf Smartphones das entscheidende Kriterium darstellt, ob eine Anwendung darauf nutzbar ist. Die Oberfläche besteht aus fünf waagerechten Elementen die untereinander angeordnet sind (s. Abbildung 3.11). Bei den Elementen handelt es sich um Slider, also Schieberegler, die teilweise waagerecht, teilweise aber auch zusätzlich vertikal gedreht werden können, um so Platz für verschiedene Funktionen zu bieten. Der oberste Slider bildet das Funktionen-Menü. Hier sind alle Elemente aufgelistet, aus denen ein Prozessmodell bestehen kann. Der zweite Slider darunter bildet die Datenelement-Ebene. Diese steht in engem Kontakt mit der dritten Ebene, welche die Aktivitäten bzw. den Kontrollfluss zeigt. Die vierte Ebene enthält Knöpfe zur Steuerung des Programms und Umsetzung der Befehle zur Modellierung, da diese Oberfläche nicht über Drag & Drop funktioniert. Die unterste Ebene zeigt eine Übersichtskarte, die Unterstützung bieten soll den Gesamtüber-

3.2 Entwicklung einer menügeführten Modellierung

blick zu wahren, da man mit der zweiten und dritten Ebene nur einen kleinen Ausschnitt des Prozessgraphen sehen kann. Die Übersichtskarte lässt sich auch mit einem Tipp auf den entsprechenden Knopf auf die gesamte Oberfläche vergrößern.

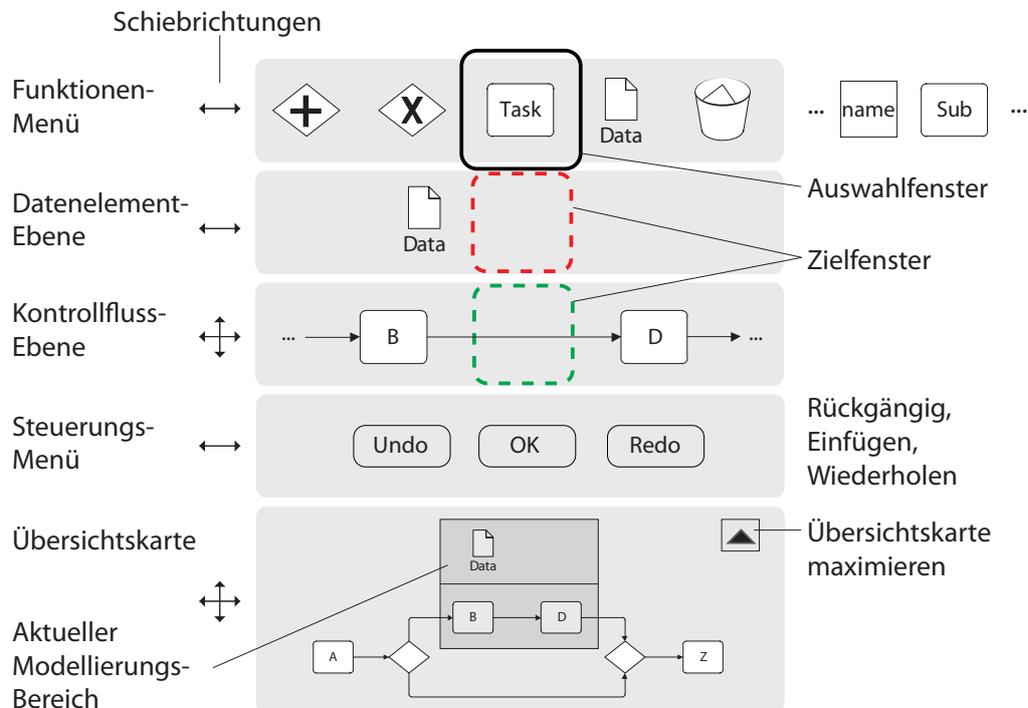


Abbildung 3.11: Das Sliderkonzept

Das Element, welches man in seinen Prozessgraphen einfügen will, platziert man in der Mitte des obersten Sliders, dem Auswahlfenster des Funktionenmenüs. Die Stelle im Prozessgraphen, an der das Element eingefügt werden soll, wird im zweiten bzw. dritten Slider ebenfalls mittig, im sogenannten Zielfenster, platziert. Währenddessen gibt das Zielfenster über seine Farbe einen Hinweis darauf, ob die Funktion auf dem gewählten Prozessfragment durchführbar ist. Ist die Kombination aus Funktion und Prozessfragment syntaktisch korrekt, wird in der Übersichtskarte eine Vorschau gezeigt und das Zielfenster leuchtet grün; anderenfalls ist es rot. Hat man nun alles in Deckung zu einander gebracht, löst man mit einem Tipp auf "OK" im vierten Slider die gewünschte Funktion aus. Waren beide Zielfenster rot wird eine Fehlermeldung ausgegeben.

Da es bei einem Datenelement eigentlich keine Rolle spielt, an welcher Stelle es im Pro-

3 Konzepte zur Prozessmodellierung auf Multi-Touch-Geräten

zessmodell auftaucht, und nur die Verbindungen (Schreib-/Lese-Kanten) zu den Aktivitäten relevant sind, kann eine genaue Platzierung im zweiten Slider vernachlässigt werden. Will man nun aber eine Lese-Kante zwischen einem bestimmten Datenelement und einer Aktivität setzen, so muss man Lese-Kante im ersten Slider, das Datenelement im zweiten und die Aktivität im dritten Slider in der Mitte in Deckung zueinander bringen, sodass alle Elemente übereinander liegen und dann auf "OK" tippen. Da Slider eindeutig arbeiten müssen, wird auf der dritten Ebene, der Kontrollfluss-Ebene, immer nur eine Kontrollflusskante angezeigt. Das heißt, bei einem Verzweigungsblock muss man die gewünschte Verzweigung über eine vertikale Scrollbewegung des dritten Sliders auswählen.

Funktionen, wie Umbenennen und Markieren, lassen sich ebenfalls im ersten Slider abbilden. Um eine Aktivität umzubenennen, schiebt man sie in das Zielfenster des dritten Sliders und schiebt die Funktion "F2: Element benennen" in das Auswahlfenster des ersten Sliders. Mit einem Tipp auf "OK" öffnet sich ein Dialog mit Tastatur, in dem man der Aktivität einen Namen geben kann. Bei der Markierungsfunktion muss ein Umweg über eine öffnende und eine schließende Klammer gegangen werden, da nicht der gesamte Bereich des Prozessgraphen im dritten Slider abgebildet werden kann. Markieren besteht also grundsätzlich aus zwei Schritten. Man setzt z.B. zuerst die linke Begrenzung, d.h. man bringt Aktivität und Klammer mittig in eine Linie und bestätigt mit "OK", und dann auf die gleiche Weise die rechte Begrenzung. Somit ist alles innerhalb dieses Rahmens markiert und es lässt sich darauf eine neue Funktion anwenden, z.B. die Subprozess-Funktion, die alles zu einem Subprozess zusammenfasst.

3.2.3 Bewertung des menügeführten Ansatzes

Der größte Vorteil von Menüs besteht in der Einfachheit ihres Aufbaus und der Schnelligkeit mit der sie von Benutzern verstanden werden. Alle Funktionen sind entweder sofort sichtbar oder können durch systematisches Durchsuchen des Menüs entdeckt werden. Insofern sind sie auch selbsterklärend. Vor allem, wenn statt Symbolen auch noch Beschriftungen verwendet werden, die klar bezeichnen, wozu eine Funktion dient. Diese Informationen müssen in der Konsequenz aber auch irgendwo untergebracht werden, was zu Lasten der Arbeitsfläche geht. Dafür bieten Menüleisten gerade für die Mehrheit der Benutzer, die noch keine Erfahrung mit Multi-Touch-Geräten sammeln konnten, einen vertrauten Anhaltspunkt. Ein weiterer Vorteil ist die im Vergleich zur Geste umfangreichere Rückmeldung, die man erhält, wenn man eine Funktion ausgewählt hat. Negativ zu bewerten sind, zumindest bei

3.3 Hybridlösung basierend auf Gesten und Menüs

den oben vorgestellten Lösungen, ebenfalls die langen Wege, die jedes mal mit dem Finger zurückgelegt werden müssen und die zahlreichen Tipps, die nach einiger Zeit zur Ermüdung führen können. Durch das einfach gehaltene Baukastenprinzip können Prozesse nur in kleinen Schritten aufgebaut werden. Für jedes Element muss der Finger zurück zum Menü geführt werden. Abhilfe können vorgefertigte Muster bieten, die Standardkonstellationen abdecken. Insgesamt ist der Arbeitsfluss dennoch mittelmäßig zu bewerten.

3.3 Hybridlösung basierend auf Gesten und Menüs

Da beide Ansätze, Gesten-basiert (s. Kapitel 3.1) und menügeführt (s. Kapitel 3.2), in ihrer Reinform Nachteile haben, wird versucht die Stärke beider Ansätze zu kombinieren. Ein Hauptproblem von Menüs und Gesten ist, dass der Weg, der jedes mal zurückgelegt werden muss, sehr lang sein kann. Ein Mittel, um den Arbeitsfluss zu verbessern, stellen Kontextmenüs dar. Also Miniaturmenüs, die an Ort und Stelle bei Bedarf eingeblendet werden und die Funktionen enthalten, die im konkreten Kontext verwendet werden können. Im Folgenden wird in Kapitel 3.3.1 das Hybrid-Set vorgestellt und anschließend in Kapitel 3.3.3 bewertet.

3.3.1 Hybrid-Set zur Prozessmodellierung

F1: Aktivität einfügen

Zum Einfügen einer Aktivität berührt man diese, oder falls es sich um einen leeren Prozessgraphen handelt den Startknoten, und zieht den Finger gedrückt rechts (bzw. links) entlang der Kontrollflusskante hinaus. Dies wird vor allem dann wichtig, wenn bei einer Verzweigung mehrere Kanten in Frage kommen. Daraufhin öffnet sich ein Kontextmenü in Form eines senkrechten Sliders auf der entsprechenden Seite der Aktivität. Nun kann die Art des einzufügenden Elements ausgewählt werden. Dies passt von einer Analogie zum Prozess auch insofern zusammen, dass sich ein Prozessschritt aus dem vorhergehenden ergibt. In diesem Fall ist das eine neue Aktivität, es stehen aber auch Verzweigungsblöcke zur Auswahl (s. Abbildung 3.12).

3 Konzepte zur Prozessmodellierung auf Multi-Touch-Geräten

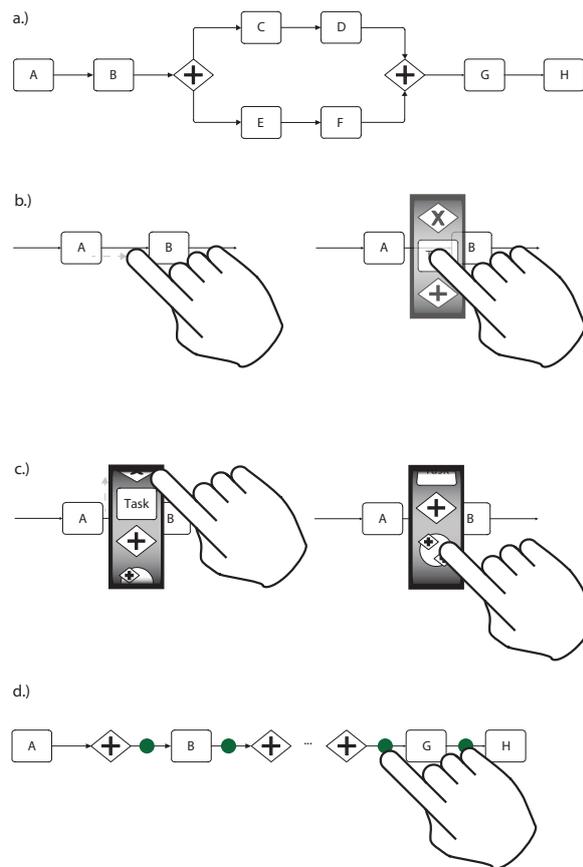


Abbildung 3.12: Slider als Kontextmenü

F2: Element umbenennen

Da der Doppeltipp als Editierfunktion etabliert ist und sehr einfach in der Ausführung ist, wird er in die Hybridlösung übernommen (s. Kapitel 3.1.1). Allerdings wird anstatt der Texterkennung auf eine virtuelle Tastatur zurückgegriffen, um ein gewohntes und schnelleres Schreiben zu ermöglichen.

F3: Verzweigungsblock einfügen

Ein neuer Verzweigungsblock kann ebenso wie eine Aktivität über ein Slider-Kontextmenü eingefügt werden. In Abbildung 3.12 a.) soll dies an einem beispielhaften Prozess gezeigt werden. Der Aufruf des Slider-Kontextmenüs erfolgt analog zum Aufruf bei der Funktion

3.3 Hybridlösung basierend auf Gesten und Menüs

“F1: Aktivität einfügen” (s. Abbildung 3.12 b). Der Unterschied zu einer Aktivität ist allerdings, dass ein Verzweigungsblock aus zwei Elementen besteht - einem öffnendem und einem schließendem Verzweigungsknoten. Diese können auf zwei verschiedene Arten platziert werden. Zum einen direkt hintereinander, zum anderen aber auch um bereits bestehende Strukturen herum. Um den Komfort zu erhöhen sind diese beiden Varianten separat im Menü vertreten. Wird die erste Variante verwendet, tippt man auf das entsprechende Symbol und der Verzweigungsblock wird vollständig eingefügt. Will man einen umgebenden Verzweigungsblock realisieren, tippt man ebenfalls auf das zugehörige Symbol (s. Abbildung 3.12 c). Daraufhin wird ein Verzweigungsknoten eingefügt. Der zweite, zugehörige Knoten muss separat platziert werden. Dabei werden alle nicht in Frage kommenden Stellen, etwa innerhalb bereits bestehender Verzweigungsblöcke, ausgeblendet und im Gegenzug die relevanten Stellen hervorgehoben (s. Abbildung 3.12 d). Der Prozess schrumpft also für die Dauer der Funktion zusammen, so dass zum einen Verletzungen der Blockstruktur von vorneherein verhindert werden, zum anderen aber auch die Wege für den Finger minimiert werden. Mit einem Tipp an die gewünschte und gültige Stelle erscheint der zweite Verzweigungsknoten und der Verzweigungsblock ist vollständig. Tippt der Benutzer auf eine nicht gültige Stelle, z.B. abseits des Prozesses, wird die Funktion abgebrochen.

F4: Verzweigung einfügen

Da eine Verzweigung immer nur zu einem bestehenden Verzweigungsblock hinzugefügt werden muss, bietet es sich an ihn im Slider-Kontextmenü für Verzweigungsknoten mitaufzunehmen. Dies hat gegenüber dem Ziehen einer Linie nach dem Drag & Drop-Prinzip, wie bei der Gesten-basierten Bedienung, den Vorteil, dass nicht beide Verzweigungsknoten auf der Oberfläche angezeigt werden müssen.

F5: Datenelement einfügen

Das Vorgehen, ein Datenelement zu erstellen wird vom Gesten-basierten Ansatz übernommen (s. Kapitel 3.1.1). Man vollzieht in der Datenelement-Ebene einen kurzen senkrechten Strich nach unten. Dieses Vorgehen gleicht also der Funktion “F1: Aktivität einfügen” aus Kapitel 3.1.1, nur dass die Geste in diesem Kontext auf oder oberhalb der Begrenzung der Datenelemente durchgeführt wird.

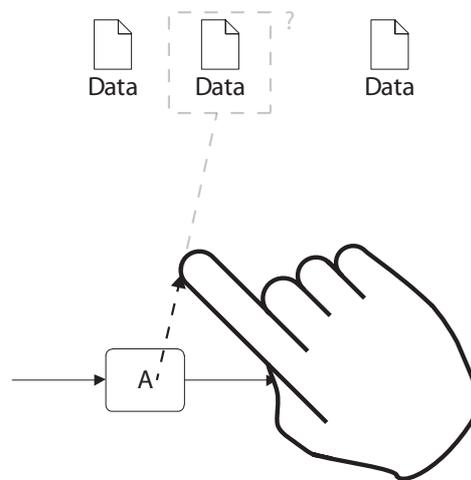


Abbildung 3.13: Setzen einer Lesekante

F6: Lese- und Schreib-Kanten setzen

Um eine Schreib-Kante von einer Aktivität zu einem Datenelement zu setzen, wird das gleiche Konzept wie beim Gesten-basierten Ansatz gewählt. Der Finger verlässt gedrückt eine Aktivität über den oberen Rand, woraufhin sich eine Kante mit Pfeil bildet. Als zusätzliche Hilfe wird hier bereits vom Programm antizipiert welches Datenelement anvisiert wird (s. Abbildung 3.13). Der Benutzer kann somit schon an dieser Stelle loslassen, wenn er mit dem Vorschlag des Systems übereinstimmt und muss den Finger nicht bis zur Begrenzung führen. Entsprechendes gilt für eine Lese-Kante, die von einem Datenelement zu einer Aktivität führt.

F7: Subprozess bilden und auflösen

Um einen Subprozess zu bilden, markiert man durch Einkreisen die gewünschten Elemente und ein Kontextmenü wird automatisch an der Stelle aufgerufen, an der man losgelassen hat. Dieses bietet einem unter Anderem an diese Elemente zu einem Subprozess zusammenzufassen. Diese Funktion erscheint nicht bei einem Kontextmenü, welches von einer einzigen Aktivität aufgerufen wird, da es nicht weiter zusammengefasst werden kann. Umgekehrt kann ein Subprozess über das Kontextmenü geöffnet oder in seine Bestandteile aufgelöst werden.

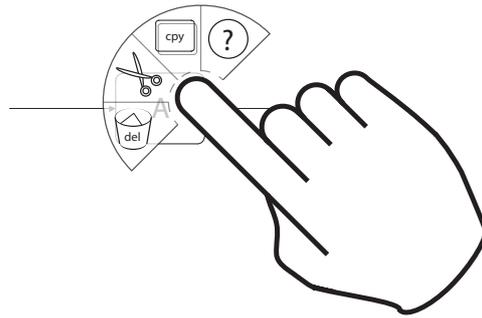


Abbildung 3.14: Element-bezogenes Kontextmenü

F8: Element Löschen

Um ein Element zu löschen, muss das Kontextmenü über dem betreffenden Element aufgerufen werden. Dies geschieht mit einem Doppeltipp, bei dem die zweite Berührung mit der Oberfläche allerdings gehalten wird. Anschließend kann der Benutzer direkt die gewünschte Funktion auswählen (s. Abbildung 3.14). Dies stellt einen Kompromiss zwischen Geschwindigkeit und Konsistenz dar, denn auf diese Weise lässt sich das Kontextmenü auch abseits einer Aktivität öffnen, was in Kapitel 3.3.2 beschrieben wird, und benötigt dafür zwei Tipps und eine kleine Bewegung.

3.3.2 Hilfsfunktionen

Auch in der Hybridlösung sind Hilfsfunktionen vorgesehen, die die Effizienz und den Komfort beim Modellieren auf einer Multi-Touch-Oberfläche steigern und so zur Zufriedenheit des Benutzers beitragen.

H1: Markieren

Einfaches Antippen und Einkreisen sind intuitive Methoden, Elemente zu markieren. Daher werden sie im Hybridansatz übernommen (s. Kapitel 3.1.2).

Element-bezogen: H3: Verschieben und Kopieren, H4: Hilfe aufrufen

Mit Elementen können verschiedene allgemeine Funktionen durchgeführt werden, die in jeder Anwendung Standard sind. Da es zwar gute Ansätze und Ideen gibt, diese durch eine Geste aufzurufen, die aber noch keine Allgemeingültigkeit besitzen und dadurch schwer erlernbar sind, wird hierfür auf das Kontextmenü zurückgegriffen. Neben dem Löschen von Elementen stehen auch eine kontextsensitive Hilfe, sowie das Verschieben und Kopieren von Elementen zur Verfügung. Das Verschieben eines Elements wird beim Kontextmenü allerdings über ein Ausschneiden und Einfügen desselben gelöst.

Nicht Element-bezogen: H2: Rückgängig und Wiederholen, H4: Hilfe aufrufen

Rückgängig und Wiederholen sind Funktionen, die sich nicht auf ein spezifisches Element beziehen, sondern global zur Verfügung stehen. Daher findet man sie, wenn man das Kontextmenü außerhalb des Prozessgraphen aufruft. Aber auch die Hilfe muss nicht nur auf ein bestimmtes Element bezogen sein und steht hier im Allgemeinen zur Verfügung.

3.3.3 Bewertung der Hybridlösung

In der Hybridlösung wurde versucht, die Nachteile, die sich aus den beiden konträren Ansätzen ergeben auszugleichen und deren Vorteile zu vereinen. Es wurde dabei auf ein Kontextmenü zurückgegriffen, um die Wege zu verkürzen und somit direkteren Zugang zu Funktionen zu ermöglichen, für die es keine etablierten Standard-Gesten gibt. Außerdem kann durch den Einsatz des Slider-Kontextmenüs, die Länge der Kanten zwischen Aktivitäten verkürzt werden, da sie nicht mehr direkt antippbar sein müssen. So wird mehr Platz für Aktivitäten geschaffen. Darüber hinaus werden beim Einfügen von Verzweigungsblöcken unwesentliche Teile des Prozessgraphen ausgelassen, so dass über weite Strecken ein Verzweigungsknoten präzise eingefügt werden kann. Dies war beim Gesten-basierten Ansatz nicht möglich. Die Zielvorrichtung beim Setzen von Lese- und Schreib-Kanten ist dabei eine Erweiterung des Gesten-basierten Ansatzes, die man auch dort schon verwenden könnte.

Eine Möglichkeit verschiedenen Benutzervorlieben zu entsprechen, wäre Gesten und Menüs redundant anzubieten, so dass eine Funktion auf verschiedene Arten durchgeführt werden kann, so wie es in herkömmlichen Programmen auch auffächernde Menüs, Leisten mit

kleinen Icons und sogenannte Shortcuts, also Abkürzungen über Tastenkombinationen der Tastatur, gibt. Auch könnte man Experten erlauben, sich ein eigenes Gesten-Set zu erstellen, um so ihrem Individualisierungswunsch nachzukommen. Bei all diesen Vorschlägen, müsste aber sichergestellt werden, dass sich nichts überschneidet und alles klar von einander unterschieden werden kann. Ein weiteres Problem ergibt sich dann, wenn Benutzer mit ihrem eigenen Gesten-Set an einem Tisch zusammenarbeiten wollen und ihnen die Tipps nicht zugeordnet werden können. Überschneiden sich unter den Benutzern keine Gesten, könnten aber alle in das Programm geladen und gleichzeitig zur Verfügung gestellt werden.

3.4 Fazit der Konzeption

Es lässt sich festhalten, dass der Gesten-basierte Ansatz mit etwas Erfahrung vor allem eine schnelle Möglichkeit bietet Prozesse zu skizzieren, ohne durch Menüs im Arbeitsfluss unterbrochen zu werden. Die hier vorgestellte Variante stößt allerdings auf größeren Bildschirmen zum Teil auf Probleme, etwa wenn umgebende Verzweigungsblöcke über weite Strecken eingefügt werden sollen.

Der menügeführte Ansatz ist etwas für Benutzer, die herkömmliche Programme gewohnt sind und noch nicht alle Funktionen der Prozessmodellierung verinnerlicht haben. Auf diese Weise haben sie diese immer im Blick und müssen sich auch keine besonderen Gesten merken. Dafür erreicht man hiermit keine so hohe Modellierungsgeschwindigkeit wie mit speziellen Gesten.

Die Hybridlösung geht wie oben besprochen einen Kompromiss ein. Ob dieser allerdings den Anforderungen der Praxis tatsächlich gerecht wird, können erst Studien mit einem Prototypen zeigen.

3 Konzepte zur Prozessmodellierung auf Multi-Touch-Geräten

4 Experimentelle Untersuchung

Entwickelt man ein neues Bedienkonzept für ein Programm, kann dies auf zwei Weisen erfolgen. Entweder man überlässt es Usability-Designern oder man involviert die späteren Benutzer des Programms. Usability-Designer halten sich dabei an grundlegende Gestaltungsrichtlinien und entwickeln systematisch einer inneren Logik und Erfahrungswerten folgend ein Gesamtkonzept für die Bedienung, wie wir es in Kapitel 3 gesehen haben. Wie viel dieses leisten kann, hängt dabei stark von der Erfahrung und dem Fachwissen des Experten ab. Eine Studie, die eine Auswahl der eigentlichen Benutzer miteinbezieht, macht sich die Weisheit der Vielen zu nutze. Diese sind Experten ihrer eigenen fachlichen Fähigkeiten und Vorlieben. Dabei gehen sie zwar unvorbereitet vor, doch genau dies hilft, eine aus ihrer Sicht intuitive Vorgehensweise zu finden. Je mehr Benutzer an der Studie teilnehmen, desto robuster wird das resultierende Bedienkonzept gegenüber einzelnen Ausreißern (d.h. individuelle Vorlieben einzelner Benutzer). In der Praxis wird dieses Vorgehen aus Zeit- und Kostengründen, aber vor allem auch aufgrund fehlender Überzeugungsarbeit gegenüber dem Management oft nicht eingesetzt. Dabei amortisieren sich die Kosten durch effizientere Bedienoberflächen und zufriedenerer Mitarbeiter schon nach kurzer Zeit.

Im Rahmen dieser Arbeit wurde eine experimentelle Untersuchung durchgeführt, bei der den Benutzern verschiedene auf einander aufbauende Aufgaben zur Prozessmodellierung gestellt wurden. Dabei hatten diese freie Hand, wie sie die Aufgaben lösen. Anschließend wurde ausgewertet, welche Wege zur Modellierung sich dabei ergeben haben und ob welche davon einen Konsens erzielen konnten. Im Folgenden also nun die Vorbereitung und Durchführung der Untersuchung in Kapitel 4.1, die Auswertung der Untersuchung in Kapitel 4.2, das daraus resultierende Gesten-Set in Kapitel 4.2.2 und abschließende Bemerkungen in Kapitel 4.2.3. Der Ablauf ist noch einmal in Abbildung 4.1 visualisiert.

4 Experimentelle Untersuchung

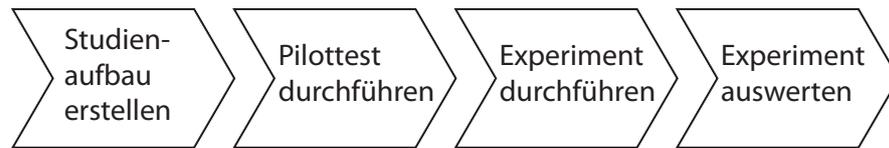


Abbildung 4.1: Schematischer Ablauf der experimentellen Untersuchung

4.1 Vorbereitung und Durchführung der experimentellen Untersuchung

Zunächst wurden acht Aufgaben formuliert, die elementare Funktionen der Prozessmodellierung abdeckten und dadurch in 15 Minuten abgearbeitet werden konnten. Die Benutzer mussten hierfür den angefangenen Prozess eines Bestellvorgangs zu Ende modellieren. Dabei stieg der Schwierigkeitsgrad der Aufgaben kontinuierlich an, um den Benutzer langsam ins Thema und an die Multi-Touch-Oberfläche hineinzuführen. Die Reihenfolge der Aufgaben blieb für alle Benutzer dieselbe. Die Aufgaben werden im Detail in der Auswertung (s. Kapitel 4.2) besprochen.

Insgesamt gab es zwei Pilotversuche, die die Machbarkeit der Aufgaben zeigen sollten, sowie 26 weitere Versuche zu denen es jeweils eine Videoaufzeichnung gibt. Insgesamt nahmen 19 Männer und 7 Frauen teil; 7 davon waren Mitarbeiter des Instituts Datenbanken und Informationssysteme der Universität Ulm, 17 Studenten und 2 Sonstige (Zahnarzthelferin, Automobilkauffrau). Es gab einen Linkshänder; die restlichen Benutzer waren Rechtshänder. 14 Benutzer gaben an, bereits Erfahrung im Umgang mit Multi-Touch-Oberflächen zu haben. Das Alter wurde nicht gesondert erfasst, es nahmen aber überwiegend junge Leute teil (in den Achtzigern geboren). Ein Großteil war bereits mit der Thematik Business Process Management vertraut.

Als Testgerät stand ein Apple iPad in Kombination mit einem Multi-Touch-fähigen Zeichenprogramm namens "Doodle Buddy" zur Verfügung [18]. Die Testaufgaben wurden jeweils als statische Einzelbilder in den Hintergrund geladen und die Benutzer konnten darauf ihre Gesten ausführen, welche als farbige Linien aufgezeichnet wurden. Die Benutzer hatten also die Spur ihrer Geste dauerhaft vor Augen. Anschließend wurde das gesamte Bild als

4.1 Vorbereitung und Durchführung der experimentellen Untersuchung

Testergebnis der jeweiligen Aufgabe abgespeichert. Abbildung 4.2 zeigt beispielhaft den Aufbau einer solchen Aufgabe. Am oberen Rand steht der Name des Prozesses und die Nummer der Aufgabe. In der Mitte befindet sich der Prozess mit der Aufzeichnung der Gesten, sowie vom Benutzer gezeichnete Elemente. Zum Teil wurde dafür vom Benutzer extra die Farbe gewechselt. Am unteren Rand befindet sich der Aufgabentext.

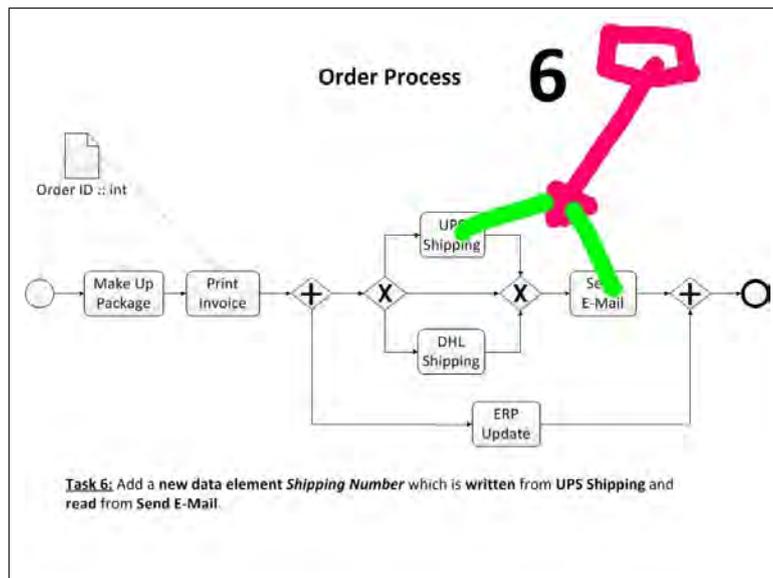


Abbildung 4.2: Prinzipieller Aufbau der Aufgaben mit Lösung eines Benutzers

Die Studie fand in einem Raum statt, in den die Benutzer einzeln hinein geführt wurden. Die Kamera stand neben dem Stuhl auf dem die Benutzer Platz nahmen und filmte von oben und etwas seitlich das Tablet. Die Probanden konnten jederzeit Fragen an den Versuchsleiter stellen, wenn ihnen eine Aufgabe nicht ganz klar war und waren aufgefordert laut zu denken und so ihr Vorgehen zu kommentieren, so dass dies ebenfalls aufgezeichnet werden konnte.

Das Ziel war zum einen heraus zu finden, wie Benutzer die Prozessmodellierung, mit welcher sie täglich zu tun haben (Mitarbeiter und Studenten des Instituts) intuitiv, mit Hilfe der Multi-Touch-Technologie durchführen und welchen Einfluss Erfahrungen in der Multi-Touch-Technologie hat.

4.2 Auswertung der experimentellen Untersuchung

Als Ergebnis der Untersuchung lagen $26 * 8 = 208$ einzelne Gesten vor, die sich auf eine Gesamtspielzeit der Videos von 6 Stunden 25 Minuten erstreckten. Gesten wurden dabei in drei Hauptkategorien eingeteilt: zeichnerische Lösungen, reine Gesten und menügeführte Lösungen. Der Unterschied zwischen einer zeichnerischen Lösung und einer Geste lässt sich an einem Beispiel erläutern: soll eine Kante von A nach B gezogen werden, wäre eine Drag & Drop-Bewegung eine Geste; eine gestrichelte Linie mit Pfeilspitze jedoch eine zeichnerische Lösung. Der Unterschied liegt im mentalen Modell des Benutzers, der entweder eher beschreibend, skizzierend vorgeht, oder abstrakt auf das Nötigste verkürzt. Beide Vorgehensweisen sind verwandt und nicht immer klar zu trennen. Sie stehen jedoch im Kontrast zur Benutzung eines Menüs. Im Anhang A.1 befindet sich die Auswertung in Tabellen- und Diagrammform mit den genauen Zahlen. An dieser Stelle soll eine Zusammenfassung der Ergebnisse vorgestellt werden. Insgesamt ließ sich beobachten, dass 57 zeichnerische Lösungen, 93 reine Gesten und 58 menügeführte Lösungen zum Einsatz kamen.

4.2.1 Detaillierte Auswertung der Aufgaben

Im Folgenden werden die Ergebnisse der experimentellen Untersuchung im Einzelnen vorgestellt. Die Aufgaben bestanden aus den Funktionen F1 bis F8, erfolgten im Experiment allerdings nicht in dieser Reihenfolge und wurden auch nicht immer vollständig abgefragt. Zur Veranschaulichung wird jeder Aufgabe ein beispielhaftes Bild der Lösungen hinzugefügt. Außerdem werden die Häufigkeiten der benutzten Gesten in einer Tabelle kategorisch zusammengefasst.

1. Aufgabe - F1: Aktivität erstellen

In Abbildung 4.3 sehen wir den initialen Prozess, wie er den Benutzern als erstes vorgelegt wurde. Er stellt einen Bestellvorgang dar, wie er in einem Unternehmen abgewickelt werden könnte. Hierbei liegt gleich zu Beginn eine Bestellnummer als Datenelement vor, welche für das Verpacken der Bestellung benötigt wird. Nach dem Verpacken wird in einem XOR-Verzweigungsknoten entschieden, ob das Paket über UPS oder DHL verschickt wird.

4.2 Auswertung der experimentellen Untersuchung

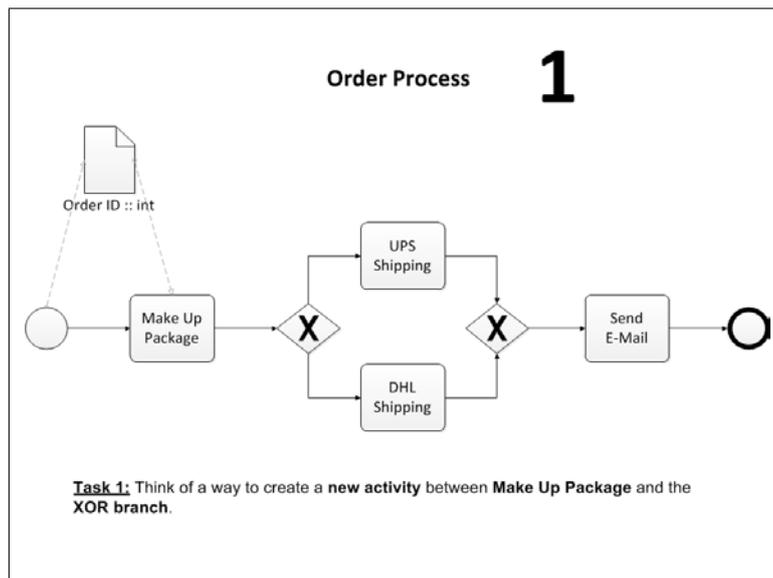


Abbildung 4.3: Initialer Prozess der Aufgaben

Anschließend wird eine E-Mail an den Kunden verschickt, um den Bestellvorgang abzuschließen. Der Benutzer musste nun den Bestell-Prozess erweitern, indem er eine weitere Aktivität zwischen dem Verpacken und dem Versenden der Bestellung einfügen sollte.

Bei der ersten Aufgabe waren sieben Benutzer überrascht, dass es sich nicht um ein funktionsfähiges Programm handelt, welches ihre Gesten und Anweisungen direkt umsetzt. In zwei Fällen haben Benutzer, welche noch nie mit BPM-Programmen gearbeitet haben, die Aufgabe zuerst nicht verstanden, was an der Terminologie und der englischen Sprache gelegen haben könnte.

Zehn Benutzer fügten die Aktivität über ein Kontextmenü ein, davon haben fünf dieses über einen einfachen Tipp auf der Kante aufgerufen (s. Abbildung 4.4 a). Aber auch Gesten wie Pinch, Linien und Pfeilspitzen gingen dem Kontextmenü voraus.

Neun Benutzer zeichneten die Aktivität hinein. Allerdings geschah dies auf höchst unterschiedliche Weise. So wurden vertikale Rechtecke auf die Kante gezeichnet, durchgehend oder mit Absetzen und es wurden Vierecke oberhalb der Kante gezeichnet, die dann mit Kanten oder Pfeilen versorgt wurden, um sie mit dem Modell zu verknüpfen.

Sechs Benutzer zogen eine bereits vorhandene Aktivität am Rand der Oberfläche per Drag & Drop auf die Kante.

4 Experimentelle Untersuchung

Eine Lösung hob sich von den anderen ab, indem sie mit einem Tipp auf den XOR-Verzweigungsknoten begann und eine von oben auf die Kante zeigende spitze Klammer folgte, über die mit dem Finger der Name der Aktivität geschrieben wurde.

Ein Blick auf die Tabelle 4.1 zeigt, dass die Lösungen relativ ausgewogen zwischen zeichnerischen und menügeführten Lösungen verteilt waren.

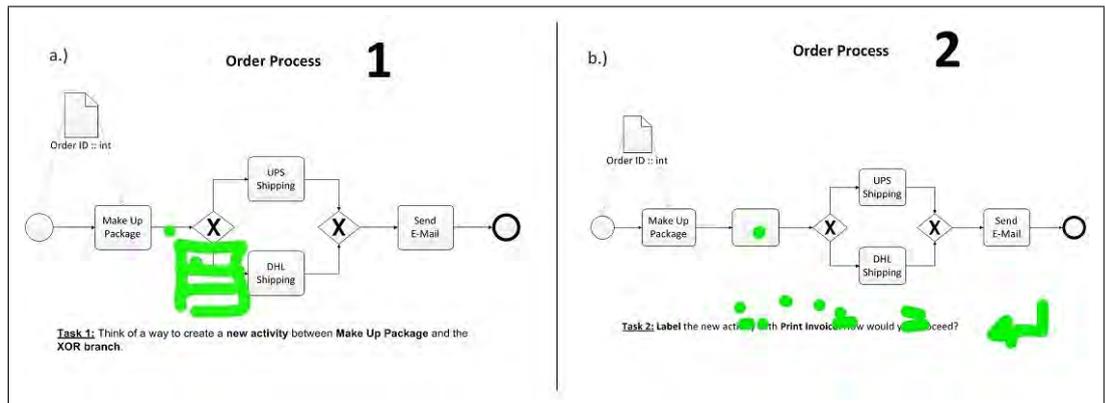


Abbildung 4.4: Aufgabe 1 gelöst über ein Kontextmenü

Tabelle 4.1: Ergebnisse der Aufgabe 1

	Gesamt	Frauen	Männer	Erfahrung	keine Erfahrung
	26	7	19	14	12
Zeichnerische Lösung	9	3	6	3	6
Reine Geste	6	1	5	4	2
Menügeführte Lösung	11	3	8	7	4

2. Aufgabe - F2: Element benennen

In der zweiten Aufgabe war der Benutzer aufgefordert, die soeben erstellte Aktivität "Print Invoice" (deutsch: Rechnung drucken) zu benennen.

15 Benutzer entschieden sich dafür eine Tastatur, wie sie in Abbildung 4.4 b.) angedeutet wurde, zu verwenden, 11 schrieben direkt mit dem Finger handschriftlich in die Aktivität, wobei einigen zuerst nicht bewusst war, dass sie eine Tastatur verwenden konnten.

4.2 Auswertung der experimentellen Untersuchung

10 mal wurde die Funktion über einen einfachen Tipp angestoßen (2 mal so abgeschlossen), sechs Benutzer starteten die Benennung mit einem Doppeltipp und zwei über einen Langtipp und zwei mit einer Pinch-Geste für die Vergrößerung der Aktivität, so dass hineingeschrieben werden konnte. Auch ein Einkreisen wurde ein mal verwendet, um den Fokus auf die Aktivität zu setzen. Manche Benutzer haben dies gar nicht bedacht und einfach drauf los geschrieben. Drei Benutzer verwendeten ein Kontextmenü und zwei einen extra Button zum Aufrufen der Tastatur. Während auch mit dem Finger versucht wurde in die Aktivität hinein zu schreiben, wurde oft ein extra Fenster angezeichnet, in das der Text geschrieben werden sollte, selbst wenn Benutzer mit einer Tastatur arbeiteten. Auffällig war, dass jeder, der mit dem Finger schrieb seinen Text abkürzte, als der Platz ausging. Ebenso war von Mausclicks und Rechtsclicks vereinzelt die Rede.

Ein Vorgehen stach heraus, bei dem der Text über die zur Verfügung stehende Fläche mit dem Finger geschrieben wurde und anschließend mit der Hand auf die betreffende Aktivität gezogen wurde.

Wie in Tabelle 4.2 zu sehen, setzte sich der Einsatz einer Tastatur, welcher als Geste interpretiert wurde, solange die Tastatur ohne weitere Hilfsmittel aufgerufen wurde, gegenüber der zeichnerischen Lösung, dem Schreiben mit dem Finger, durch.

Tabelle 4.2: Ergebnisse der Aufgabe 2

	Gesamt	Frauen	Männer	Erfahrung	keine Erfahrung
	26	7	19	14	12
Zeichnerische Lösung	8	4	4	3	5
Reine Geste	16	3	13	10	6
Menügeführte Lösung	2	0	2	1	1

3. Aufgabe - F6: Lese-Kante setzen

Bevor nun vor dem Versenden der Bestellung die dazugehörige Rechnung ausgedruckt werden kann, muss die entsprechende Bestell-Nummer bereitgestellt werden. Der Benutzer musste also die Aktivität "Print Invoice" über eine Lese-Kante mit dem Datenelement "Order ID :: int" verbinden.

15 Benutzer zogen eine direkte Linie vom Datenelement zur Aktivität. Ein mal davon wurde erst ein entsprechendes Werkzeug dafür aus einer Menüleiste ausgewählt und ein weite-

4 Experimentelle Untersuchung

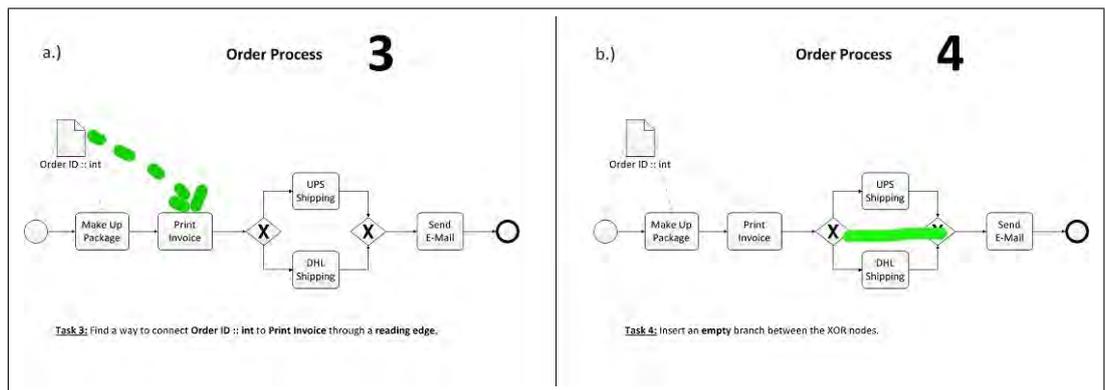


Abbildung 4.5: Zeichnerische Lösung einer Lese-Kante in Aufgabe 3

Der Benutzer tippte zuerst Start- und Zielelement an, bevor die Linie gezogen wurde. Ein anderer Benutzer tippte zuerst das Datenelement an und verlangte, dass daraufhin eine spezielle Andockmöglichkeit erscheint, die man anfassen kann, um eine Kante zu ziehen. Zwei Benutzer sprachen sich für ein Kontextmenü aus, mit dem man auch nachträglich den Typ (Lese-/Schreib-Kante) der Kante ändern kann, ausgelöst einmal durch einen einfachen Tipp und einmal durch einen Doppeltipp.

Fünf Personen malten eine gestrichelte Linie mit Pfeilspitze, wie in Abbildung 4.5 a.) zu sehen, sowie eine ohne Pfeilspitze. Zum Teil wurde dies bewusst so gemacht, um den Typ der Kante deutlich zu machen, zum Teil wurde dies nur getan, um der Vorlage gerecht zu werden. Drei Personen malten einen durchgehenden Pfeil. Ein mal wurde die Lese-Kante per Drag & Drop auf das Datenelement und die Aktivität gezogen und ein mal wurde die Kante über ein Kontextmenü gesetzt, nachdem Start- und Zielelement angetippt wurden.

In Tabelle 4.3 erkennt man deutlich, dass vor allem männliche und erfahrene Benutzer auf eine Geste zurückgriffen, während über die Hälfte der Frauen eine zeichnerische Lösung bevorzugten.

Tabelle 4.3: Ergebnisse der Aufgabe 3

	Gesamt	Frauen	Männer	Erfahrung	keine Erfahrung
	26	7	19	14	12
Zeichnerische Lösung	8	4	4	2	6
Reine Geste	15	1	14	10	5
Menügeführte Lösung	3	2	1	2	1

4. Aufgabe - F4: Verzweigung einfügen

Nachdem die Vorbereitungen des Versands abgeschlossen sind, soll der Benutzer dem XOR-Verzweigungsblock, der den Versand modelliert, eine leere Verzweigung hinzufügen. Diese wird nicht näher bestimmt.

17 Benutzer führten die gleiche Geste aus wie in der vorherigen Aufgabe (d.h. beispielsweise wenn ein Benutzer eine Linie vom Datenelement zur Aktivität zog dann hat er auch in dieser Aufgabe die beiden Verzweigungsknoten mit einer Linie verbunden; s. Abbildung 4.5 b). Die gestrichelten Pfeile wurden jedoch alle zu durchgehenden. Eine gestrichelte Linie ohne Pfeilspitze blieb gestrichelt. Die Bewegungsrichtung war immer von links nach rechts. Benutzer, die vorher Start- und Zielelemente auswählten, stiegen auf Kontextmenüs um, die drei mal durch Langtipps ausgelöst wurden und ein mal durch einen einfachen Tipp. Einer der Benutzer brachte den Vorschlag in einem Kontextmenü neben einer leeren Verzweigung auch eine mit einer Aktivität bestückten Verzweigung und weitere anzubieten und legte keinen Wert darauf, ob der öffnende oder schließende Verzweigungsknoten angewählt wurde.

Auch in Tabelle 4.4 erkennt man den gleichen Trend wie in der vorherigen Aufgabe. Allerdings werden auch die Umsteiger auf die Kontextmenüs bei den erfahrenen Benutzern sichtbar.

Tabelle 4.4: Ergebnisse der Aufgabe 4

	Gesamt	Frauen	Männer	Erfahrung	keine Erfahrung
	26	7	19	14	12
Zeichnerische Lösung	8	4	4	2	6
Reine Geste	12	1	11	7	5
Menügeführte Lösung	6	2	4	5	1

5. Aufgabe - F3: Verzweigungsblock einfügen

Um den Versandvorgang auch im Computer des Unternehmens abzubilden, soll parallel zum Versand des Pakets und der E-Mail ein ERP-Update (Enterprise Resource Planning) gemacht werden. Hierzu musste der Benutzer allerdings nur den umgebenden AND-Verzweigungsblock um den bestehenden XOR-Verzweigungsblock und die Aktivität "Send

4 Experimentelle Untersuchung

E-Mail" herstellen. Die restlichen Schritte wurden ausgespart, da der Benutzer Ähnliches bereits in Aufgabe 1 und Aufgabe 2 zeigte.

Vier Benutzer zogen eine Linie vom öffnenden Verzweigungsknoten zum schließenden Verzweigungsknoten, woraufhin sich ein Kontextmenü öffnen sollte. In sieben weiteren Fällen wurde das Kontextmenü anders aufgerufen: zum Beispiel wurde von der Kante aus eine Linie nach oben gezogen, ein Langtipp an der Stelle gehalten oder es wurden die Elemente um den Verzweigungs-Startpunkt angetippt oder auch die äußersten Elemente, die der Verzweigung noch angehören sollten. Auch die Idee mögliche Verzweigungs-Endpunkte vom System anzeigen zu lassen und selbst dann nur noch bestätigen zu müssen, tauchte in diesem Kontext auf. Vier Benutzer markierten den einzuschließenden Bereich. So wurde ein Kasten um die betreffenden Elemente gezogen und (mit einem "Rechtsklick" durch den Mittelfinger) ein Kontextmenü aufgerufen, welches einem unter Anderem anbot die Elemente mit einer AND-Verzweigung zu umgeben. Als Korrektur zum Rechtsklick bot der Benutzer einen einfachen Tipp an, da in dieser Situation nichts anderes Sinn ergab, als ein Menü aufzurufen. Als problematisch sah der Benutzer jedoch diese Verwendung bei einzelnen Elementen, die evtl. nur markiert werden sollen. Er vermochte es jedoch nicht dieses Dilemma spontan aufzulösen. Eine Variante eines anderen Benutzers ersetzte das Kontextmenü durch das Anbringen von je einem Plus vor und hinter den gezogenen Kasten. Ein weiterer Benutzer kreiste die Elemente ein und zeichnete eine Raute mit Plus davor. Der vierte Benutzer schließlich bediente sich eines Multi-Selektionswerkzeuges aus einer unteren Menüleiste mit dem er die Elemente einzeln nacheinander durch Antippen markierte und beim letzten durch einen Langtipp ein Kontextmenü aufrief.

Insgesamt wurde in acht Fällen eine Raute mit einem Plus gezeichnet. Dabei gab es allerdings unterschiedliche Konstellationen, die den Schluss zulassen, dass es auch beim Zeichnen nicht die eine, intuitive Vorgehensweise gibt. So gab es unterschiedliche Verknüpfungen der Verzweigungen mit dem Prozessgraphen. In einem Fall wurde die Raute über Pfeile mit den Elementen verbunden, in anderen wurde sie direkt auf die Kante gesetzt. Doch auch dort kam es aufgrund von Platzproblemen zu unterschiedlichen Behelfsmethoden, oder es wurde hingenommen. So wurde beispielsweise ein Strich von der Kante nach oben gezogen, auf den dann die Raute gezeichnet wurde. Eine andere Lösung war die umgebenden Elemente mit einer Pinch-Geste zur Seite zu schieben. Oder die Rauten wurden auf die Verbindungslinie gezeichnet, die den Start und Endpunkt des Verzweigungsblockes durch Schneiden der Kontrollflusskante markieren sollte. Diese Linie wurde in sieben Fällen gezogen und enthielt teilweise bereits eine neue Aktivität, obwohl dies nicht gefordert war

4.2 Auswertung der experimentellen Untersuchung

(s. Abbildung 4.6 a.). Fünf Benutzer zeichnerten zwei Rauten, drei Benutzer allerdings nur eine. Ein Benutzer nahm beim Zeichnen gleich mehrere Anläufe, weil er mit dem Ergebnis nicht zufrieden war. Dies zeigt die Komplexität der Aufgabe aus Sicht der Benutzer.

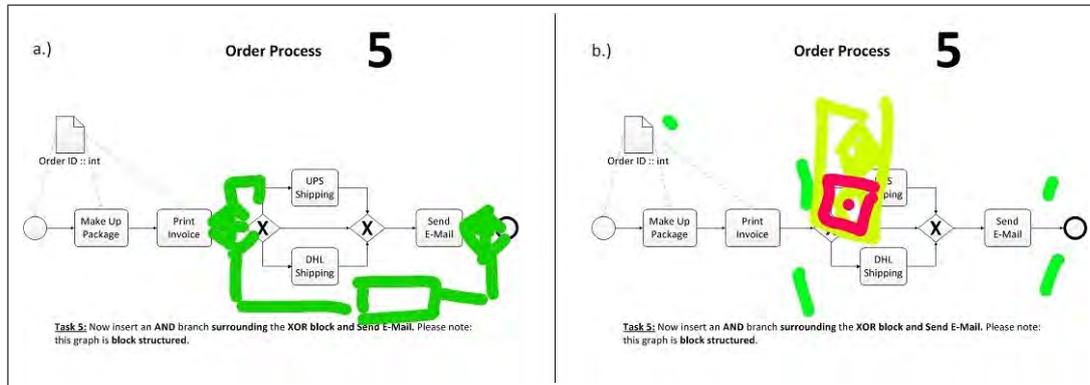


Abbildung 4.6: Gezeichneter Verzweigungsblock um bestehende Elemente löst Aufgabe 5

Fünf Benutzer verwendeten ein Drag & Drop-Menü, dieses allerdings auf drei verschiedene Arten. In allen Fällen wurde der erste AND-Verzweigungsknoten auf die Kante geschoben, aber danach kam es zu unterschiedlichen Handlungsweisen: dreimal wurde der zweite Verzweigungsknoten wie der erste von der Menüleiste auf die andere Stelle geschoben, einmal wurde es vom erstplatzierten heraus an die zweite Stelle geschoben und ein andermal wurde der betreffende Kontrollblock eingekreist und auf den öffnenden Verzweigungsknoten geschoben.

Es gab bei dieser Aufgabe lediglich zwei Gesten-basierte Lösungen. Eine bestand darin mit zwei Fingern gleichzeitig einen Strich an den entsprechenden Stellen herunter zu ziehen, die andere diese Striche nach oben zu ziehen. Weiter wurde angesprochen für einen XOR-Verzweigungsblock eine senkrechte Pinch-Geste mit beiden Händen an den Stellen durchzuführen, da ja etwas auseinander gehen (s. Abbildung 4.6 b.). Auch über eine öffnende und eine schließende Pinch-Bewegung für einen öffnenden und einen schließenden Verzweigungsknoten wurde nachgedacht.

Insgesamt gab es 20 unterscheidbare Lösungsansätze, die man grob gruppieren kann in die Kategorien: Kontextmenü, Raute zeichnen, Einkreisen, Drag & Drop und Gesten. Es gab sogar eine Lösung, die diese Vielfalt in sich vereinigte. Dabei wurde zuerst die Kante selektiert, mit einer Pinch-Geste Platz geschaffen, eine Raute hinein gemalt, diese selektiert und ein Kontextmenü geöffnet, über welches die Art der Verzweigung bestimmt wurde.

4 Experimentelle Untersuchung

Anschließend wurde von dort eine Linie an das Ende gezogen, woraufhin sich der zweite Verzweigungsknoten entsprechend bildete.

Tabelle 4.5 zeigt ein ausgeglichenes Verhältnis zwischen den verschiedenen Hauptkategorien. Lediglich bei der Unterscheidung zwischen erfahrenen und unerfahrenen Benutzern ist ein Trend in Richtung Menü bzw. zeichnerische Lösung erkennbar.

Tabelle 4.5: Ergebnisse der Aufgabe 5

	Gesamt	Frauen	Männer	Erfahrung	keine Erfahrung
	26	7	19	14	12
Zeichnerische Lösung	9	4	5	2	7
Reine Geste	8	1	7	5	3
Menügeführte Lösung	9	2	7	7	2

6. Aufgabe - F5: Datenelement einfügen

In der sechsten Aufgabe musste der Benutzer selbst ein Datenelement namens "Shipping Number" anlegen. Anschließend sollte es von der Aktivität "UPS Shipping" über eine Schreib-Kante beschrieben werden und von der Aktivität "Send E-Mail" über eine Lese-Kante gelesen werden.

Es gab 11 zeichnerische Lösungen, die sich verschiedener geometrischer Formen bedienten. Neben einem Dreieck und einem Kreis wurden vor allem viereckige Figuren gezeichnet, sowohl durchgehend als auch mit Absetzen. Es gab Versuche das Vorbild des Datenelements exakt nachzuahmen oder es mit einem diagonalen Balken in der rechten oberen Ecke anzudeuten. Aber auch ein Rechteck mit zwei waagerechten Balken in der Mitte, die an einen Brief erinnern sollten, wurde gezeichnet. Den Benutzern war es offensichtlich wichtig die Art des Knotens im Prozessgraphen optisch zu unterscheiden. Einmal sollte ein Rechteck dadurch als Datenelement erkannt werden, dass es durch entsprechende Kanten mit dem Prozessgraphen verbunden wurde. Die Kanten wurden zum größten Teil auf die gleiche Weise wie in Aufgabe 3 gezogen, einmal allerdings wurden aus gestrichelten Pfeilen doch durchgehende, da sie einfacher zu zeichnen waren. Dem Benutzer war es offenbar nicht bewusst, dass das Programm dies trotzdem unterscheiden können würde. Bei einer anderen Lösung wurde ein Kontextmenü vorgeschlagen, falls aus der Richtung der Kanten nicht klar würde, ob es sich um eine Lese- oder Schreib-Kante handelt.

4.2 Auswertung der experimentellen Untersuchung

Sechs Benutzer zogen das Datenelement aus einer Menüleiste heraus auf die Arbeitsfläche (s. Abbildung 4.7 a) und ein mal direkt auf die Aktivität, welche somit gleich mit einer Schreib-Kante verbunden werden sollte. Der Datentyp sollte hierbei entweder direkt aus der Menüleiste heraus auswählbar sein oder hinterher über ein Kontextmenü. Es wurde auch erstmals eine ausfahrbare Menüleiste angesprochen, die sich erst zeigt, sobald man mit dem Finger an den Rand der Oberfläche tippt. Ein Benutzer, der überwiegend Menüs zur Lösung der Aufgaben heranzog, stellte im Gegensatz zu Aufgabe 3, die Kanten nun nicht mehr über ein Drag & Drop her, sondern zeichnete gepunktete Linien.

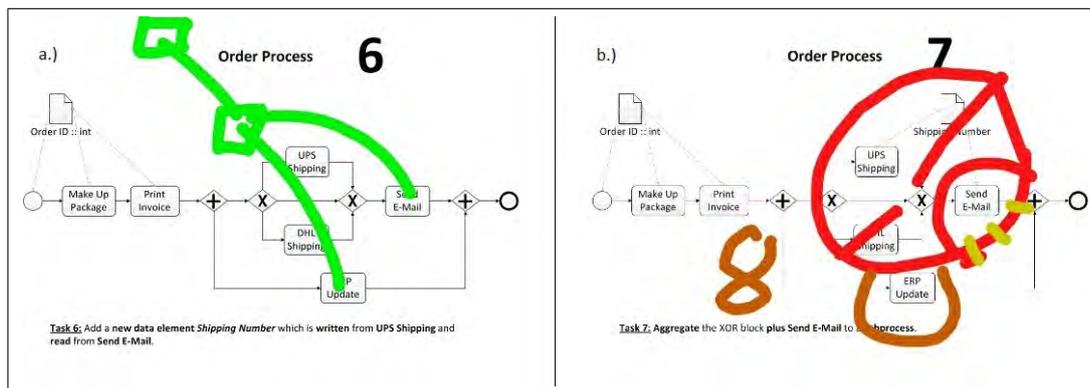


Abbildung 4.7: Drag & Drop aus Menüleiste als Lösung für Aufgabe 6

Sieben Benutzer verwendeten ein Kontextmenü. Drei mal davon wurde es über einen Langtipp abseits des Prozesses aufgerufen, einmal über einen einfachen Tipp, einmal als Ergänzung zu oben angesprochenem Kreis, worüber der genaue Datentyp eingestellt werden sollte und zwei mal nach einer Linienbewegung aus der Aktivität heraus. Diese Linienbewegung wurde zusätzlich ein mal ohne Kontextmenü vorgeschlagen, so wie es im Gestenbasierten Ansatz dieser Arbeit beschrieben wurde (s. Kapitel 3.1.1).

Beim Benennen des Datenelements entschied sich ein Benutzer auf einmal für die Tastatur, obwohl dieser sich in Aufgabe 2 noch klar für das Schreiben mit dem Finger ausgesprochen hatte. Ein andere Lösung bestand darin, den Namen direkt auf die Arbeitsfläche zu schreiben und diesen dann über Linien mit den Aktivitäten zu verbinden. Eine ähnliche Lösung zeichnete das Datenelement zusätzlich. Hier wurde aufgrund der Größe der Schrift ein Bedienstift vorgeschlagen. Ein Benutzer verlangte, dass die Elemente erst angetippt werden müssen, um spezielle Andockmöglichkeiten freischalten zu können, die man anfassen muss, wenn man die Kanten ziehen möchte. Als weitere Idee wurde genannt, das bereits vorhandene Datenelement-Konstrukt zu kopieren und für den neuen Verwendungs-

4 Experimentelle Untersuchung

zweck einzusetzen. Es wurde aber keine Handlungsanweisung genannt. Einem Benutzer war bei dieser Aufgabe noch nicht bewusst, dass sie sich eine Menüleiste dazu erfinden dürfen, weshalb zuerst ein Kontextmenü mit einem Langtipp geöffnet wurde.

In Tabelle 4.6 ist wieder der Trend erkennbar, dass erfahrene Benutzer überwiegend zu Menüs greifen, während unerfahrene Benutzer wieder zeichnerische Lösungen bevorzugen. Mit Gesten scheint diese Aufgabe kaum zu lösen zu sein.

Tabelle 4.6: Ergebnisse der Aufgabe 6

	Gesamt	Frauen	Männer	Erfahrung	keine Erfahrung
	26	7	19	14	12
Zeichnerische Lösung	10	4	6	3	7
Reine Geste	4	0	4	2	2
Menügeführte Lösung	12	3	9	9	3

7. Aufgabe - F7: Subprozess bilden

Um den Prozess übersichtlicher zu gestalten, soll der Benutzer den XOR-Verzweigungsblock und die Aktivität "Send E-Mail" zu einem Subprozess zusammenfassen.

12 Benutzer benutzten ein Rechteck in ihrer Lösung, acht einen Kreis, vier nahmen ein Werkzeug zu Hilfe und zwei umrandeten die Elemente in einer freien Form, wovon einer dies mit zwei Fingern einer Hand auf einmal tat. Für 12 war eine Geste wie ein Rechteck oder ein Kreis nur der Auslöser für ein Kontextmenü, welches die Funktion eindeutig machen sollte und für sechs, darunter fünf Rechtecke, stand die Geste für sich. Dabei wurde das Kontextmenü acht mal automatisch aufgerufen und sonst durch bestimmte Tipps. 6 mal wurde das Datenelement nicht berücksichtigt. In einem Fall zog der Benutzer zügig einen Kreis um die Aktivitäten, welcher dadurch recht eng anliegend saß, während der Rest akkurater arbeitete. Auch wurde beobachtet, dass ein Benutzer während der Ausführung der Geste kurz pausierte, überlegte und dann die Geste zu Ende führte.

Eine Lösung stach dadurch hervor, bei der der Benutzer die Elemente auswählte, ausschchnitt und eine neue Aktivität erzeugte, in die er die Elemente wieder einzufügen versuchte. Ein weiterer Vorschlag war, den umkreisten Elementen weitere hinzuzufügen oder abzuziehen, indem man an der Umrandung ansetzte und weitere Halbkreise nach außen oder

4.2 Auswertung der experimentellen Untersuchung

innen um die gewünschten Elemente zog, die wieder an der ersten Umrandung endeten (s. Abbildung 4.7 b). Auch zwei Pinch-Gesten, sowohl bei einem Rechteck als auch Kreis, wurden beobachtet.

Im Gegensatz zur Aufgabe 6 liegen hier in Tabelle 4.7 die Gesten deutlich vorne. Lediglich bei den weiblichen Benutzern gibt es ein ausgeglichenes Verhältnis.

Tabelle 4.7: Ergebnisse der Aufgabe 7

	Gesamt	Frauen	Männer	Erfahrung	keine Erfahrung
	26	7	19	14	12
Zeichnerische Lösung	3	2	1	1	2
Reine Geste	19	3	16	10	9
Menügeführte Lösung	4	2	2	3	1

8. Aufgabe - F8: Element löschen

Als letzte Aufgabe soll die anfangs erstellte Aktivität "Print Invoice" wieder gelöscht werden.

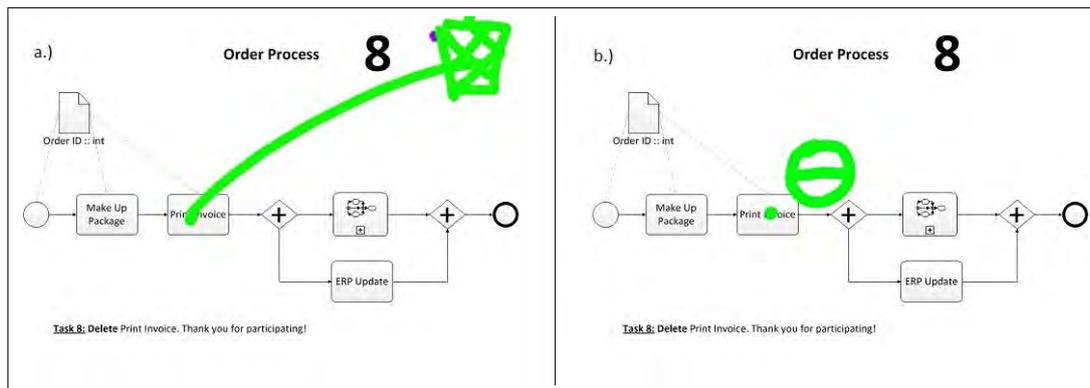


Abbildung 4.8: Lösung Aufgabe 8: Aktivität wird in einen Papierkorb gezogen

Neun Benutzer verwendeten ein Kontextmenü des Elements, welches in vier Fällen über einen Langtipp aufgerufen wurde. Einmal kam die Idee auf, das Kontextmenü über einen Klick auf den Rand der Aktivität zu öffnen, da ein einfacher Tipp die Aktivität nur selektieren und ein doppelter sie umbenennen sollte.

Sechs Benutzer strichen die Aktivität mit einem X durch, wobei es allerdings auf vier unterschiedliche Weisen gezeichnet wurde. Ein Benutzer löschte die Aktivität mit einem ein-

4 Experimentelle Untersuchung

fachen diagonalen Strich über das Element und eine weitere deutete dies mit Zick-Zack-Linien an. Ein Benutzer sprach das Problem an, dass man bei einem X eventuell zuerst die Lese-Kante erwischen könnte, was zu Verwechslungen führen könnte. Als Nebenbemerkung an dieser Stelle sei erwähnt, dass ein Benutzer, der die Aufgabe über ein Kontextmenü löste, das X als Visualisierung für das Löschen einsetzte. Ein anderer Benutzer, welcher zuerst ein Kontextmenü erwähnte, entschied sich aus Trendgründen für die X-Geste.

Fünf Benutzer bevorzugten es die Aktivität aus dem Prozess heraus zu ziehen, wobei dies zwei nach oben heraus taten, einer nach unten und zwei sich eine Art Papierkorb in der rechten oberen Ecke vorstellten (s. Abbildung 4.8 a).

Zwei Benutzer arbeiteten mit einer Werkzeugleiste und weitere zwei lehnten ihr Vorgehen aus Gewohnheit an das Löschen von Programmen (sogenannten Apps) im Apple iOS an (s. Abbildung 4.8 b).

Eine auffällige Variante war, dass ein Strich aus der Aktivität nach oben heraus gezeichnet wurde, über den der Befehl "delete" mit dem Finger geschrieben wurde.

Auffällig bei dieser Aufgabe war die Anzahl der vorgeschlagenen Varianten von einzelnen Benutzern, die teilweise bis zu drei Lösungsmöglichkeiten anboten. Insgesamt wurde vier mal eine Sicherheitsabfrage vorgeschlagen, oder auch Bestätigungen, was mit den restlichen Elementen zu tun sei. Zweimal wurde die Kante beispielsweise extra durchgestrichen. Andere automatisierten alles, wo es keine Konflikte gab. Ein Benutzer stellte die Frage, wie das Löschen eines Subprozesses interpretiert werden sollte.

Tabelle 4.8 zeigt ein ausgeglichenes Verhältnis zwischen reinen Gesten und menügeführten Lösungen.

Tabelle 4.8: Ergebnisse der Aufgabe 8

	Gesamt	Frauen	Männer	Erfahrung	keine Erfahrung
	26	7	19	14	12
Zeichnerische Lösung	2	1	1	0	2
Reine Geste	13	4	9	7	6
Menügeführte Lösung	11	2	9	7	4

4.2.2 Resultierendes Gesten-Set aus der experimentellen Untersuchung

Im Folgenden wird ein Gesten-Set vorgestellt, das auf der zahlenmäßigen Auswertung der experimentellen Untersuchung beruht. Man könnte es als Quintessenz der eben besprochenen Einzelergebnisse betrachten. Dabei wurde darauf geachtet, welche Hauptkategorie (zeichnerische Lösung, reine Geste, menügeführte Lösung) der Benutzung überwiegt und davon die Methode mit der größten Übereinstimmung ausgewählt. Diese Entscheidung ist gerade bei den Aufgaben "F3: Verzweigungsblock erstellen" und "F5: Datenelement einfügen" nicht unstrittig. So gab es bei letzterem zum Beispiel weitaus mehr zeichnerische Lösungen als andere, doch unterschieden sich diese wiederum stark von einander in ihrer konkreten Ausgestaltung. Bei der Aufgabe "F3: Verzweigungsblock erstellen" gewann eine Lösung mit nur vier Übereinstimmungen, da es eine Vielzahl verschiedener Lösungen gab. Des Weiteren wird hier die Inkonsistenz dieses Gesten-Sets als Schwäche von demokratischen Designs offengelegt. So sind verschiedene Mischformen erkennbar. Vor allem das Kontextmenü ist in der Hälfte aller Aufgaben involviert, wird aber jedes mal auf eine andere Weise geöffnet. Ein weiteres Beispiel ist, dass eine Aktivität über ein Kontextmenü eingefügt wird, ein Datenelement jedoch eingezeichnet wird.

F1: Aktivität einfügen Ein einfacher Tipp auf die Kante öffnet ein Kontextmenü. Der Benutzer wählt die Funktion "F1: Aktivität einfügen".

F2: Element benennen Ein einfacher Tipp auf die Aktivität setzt einen Cursor in das Bezeichnungsfeld und eine Tastatur wird eingeblendet. Das Umbenennen beendet der Benutzer mit einem Tipp auf die Eingabetaste der Tastatur.

F3: Verzweigungsblock einfügen Der Benutzer zieht eine Linie vom gewünschten Startpunkt des Verzweigungsblockes zum dazugehörigen Ende. Beim Loslassen öffnet sich ein Kontextmenü, aus dem der Benutzer wählen kann, welchen Typ eines Verzweigungsblockes er einfügen möchte.

F4: Verzweigung einfügen Eine Linie, die zwischen den Verzweigungsknoten gezogen wird, erstellt eine neue, leere Verzweigung.

F5: Datenelement einfügen Das System interpretiert ein Viereck als neues Datenelement, wenn es abseits des Prozessgraphen gezeichnet wird.

4 Experimentelle Untersuchung

F6: Lese- und Schreib-Kanten setzen Der Benutzer zieht wie beim Gesten-basierten Ansatz eine Linie zwischen Datenelement und Aktivität. Die Richtung bestimmt, ob es eine Lese- oder Schreib-Kante ist. Führt die Bewegung von der Aktivität zum Datenelement, so wird dies als Schreibzugriff interpretiert, andersherum als Lesezugriff.

F7: Subprozess bilden und auflösen Der Benutzer zeichnet ein Rechteck um die auszulagernden Aktivitäten, woraufhin sich ein Kontextmenü öffnet, über das er die Funktion abschließen kann. Das Rechteck kann dabei durchgehend oder mit Absetzen gezeichnet werden. Ebenso die Richtung spielt keine Rolle.

F8: Element löschen Der Benutzer führt einen Langtipp über der Aktivität aus, um das Kontextmenü zu öffnen und wählt die Funktion "F8: Element löschen".

Dieses Resultat bestätigt die Arbeit an einer Hybridlösung basierend auf einer Kombination von Gesten und Menüs (s. Kapitel 3.3). Es wird für die Hälfte aller Funktion im resultierenden Gesten-Set der experimentellen Untersuchung ein Menü eingesetzt, dennoch gibt es keine festen Menüleisten wie sie in Kapitel 3.2 vorgestellt wurden. Sondern es sind Kontextmenüs, die dann aufgerufen werden, wenn es keine intuitive Geste dazu gibt. Umgekehrt liegt die Stärke der Gesten oder Zeichnungen darin, Sachverhalte zu beschreiben. Verbindungen wie Lese-Kanten herzustellen oder eine Menge zu definieren (Markieren, Subprozess bilden) ist über eine Geste leichter zu bewerkstelligen als über ein Menü. Vergleicht man nun konkret die in Kapitel 3.3 vorgestellte Hybridlösung mit dem hier vorgestellten Gesten-Set erkennt man viele Ähnlichkeiten. So werden vier Funktionen ("F1: Aktivität einfügen", "F3: Verzweigungsblock einfügen", "F7: Subprozess bilden und auflösen", "F8: Element löschen") in beiden Ansätzen über Kontextmenüs aufgerufen. Da in der Hybridlösung allerdings großer Wert auf eine konsistente Bedienung gelegt wird, kommt es zwangsläufig zu Unterschieden in der genauen Durchführung des Aufrufs des Kontextmenüs. Die Funktion "F2: Element benennen" unterscheidet sich in der Hybridlösung nur durch einen Doppeltipp statt des einfachen Tipps von dem hier vorgestellten Gesten-Set. Dies lässt sich darauf zurückführen, dass ein einfacher Tipp in der Hybridlösung bereits für das Markieren eines einzelnen Elements reserviert ist. "F4: Verzweigung einfügen" gleicht hier dem Vorgehen aus der Gesten-basierten Modellierung aus Kapitel 3.1.1. In der Hybridlösung wurde dieses Vorgehen aufgrund des Slider-Kontextmenüs nicht übernommen. Die einzige Funktion, die sich hier grundlegend von der Hybridlösung unterscheidet ist "F5: Datenelement einfügen". Im Gegenzug ist "F6: Lese- und Schreib-Kanten setzen" identisch.

4.2.3 Abschließende Bemerkungen

Es ist festzustellen, dass es von der Aufgabe abhängt, welche Hauptkategorie von Geste (zeichnerische Lösungen, reine Gesten, menügeführte Lösungen) jeweils bevorzugt wird. So haben beispielsweise zeichnerische Lösungen über die ersten sechs Aufgaben konstant einen Anteil von einem Drittel, während Gesten und Menüs größeren Schwankungen unterliegen. Reine Gesten werden in fünf Fällen den anderen Varianten vorgezogen und in zwei weiteren Fällen fast als ebenbürtig angesehen. Menüs haben einen Gesamtanteil von 27,4%. Wobei man bei der Interpretation wissen muss, dass es bei der ersten und bei den letzten drei Aufgaben zu Hybridlösungen von Gesten und Kontextmenüs gekommen ist. Dies wird in der Auswertung als Geste interpretiert, da sie die Funktion initiiert hat. Eine Ausnahme bilden hierbei einfache Tipps, da sie zu unbestimmt sind und das Kontextmenü die entscheidende Rolle spielt. Ein Menü wird mengenmäßig jedoch in keiner Aufgabe den anderen Lösungswegen vorgezogen. Zeichnerische Lösungen werden in fünf Fällen bevorzugt und in einem als fast gleichwertig angesehen.

Unterscheidet man zwischen Benutzern mit und ohne Erfahrung mit Multi-Touch-Geräten ergibt sich folgende Verteilung der Ergebnisse: bei Benutzern ohne Erfahrung mit Multi-Touch-Oberflächen wurden 41 zeichnerische Lösungen, 38 reine Gesten und 17 menügeführte Funktionen verzeichnet. Diese Zahlen entsprechen auch in etwa den prozentualen Anteilen. Dem gegenüber stehen erfahrene Benutzer mit 16 zeichnerischen Lösungen, 56 reinen Gesten und 40 menügeführten Lösungen. Es ist deutlich zu sehen, dass unerfahrene Benutzer häufiger zu zeichnerischen Lösungen greifen als erfahrene Benutzer. Während diese wiederum häufiger auf Menüs zurückgreifen als unerfahrene Benutzer. Auch reine Gesten benutzen sie etwas häufiger. Außerdem kamen hier bei der ersten und letzten Aufgabe die Hybridlösungen öfter zum Einsatz. Erfahrene Benutzer übertragen also die Bedienung ihrer eigenen Multi-Touch-Geräte auf die Bedienung einer Prozessmodellierung auf Multi-Touch-Geräten.

Unterscheidet man zwischen weiblichen und männlichen Benutzern ergibt sich folgende Verteilung: bei den weiblichen Benutzern sind 26 (46,4%) zeichnerische Lösungen, 14 (25,0%) reine Gesten und 16 (28,6%) menügeführte Lösungen zu verzeichnen. Bei den männlichen Benutzern wurden 31 (20,4%) zeichnerische Lösungen, 79 (52,0%) reine Gesten und 42 (27,6%) menügeführte Lösungen umgesetzt. Anteilig betrachtet greifen weibliche und männliche Benutzer gleich oft auf ein Menü zurück. Sie unterscheiden sich al-

4 Experimentelle Untersuchung

lerdings diametral in der Benutzung von Gesten und Zeichnungen. So greifen männliche Benutzer doppelt so häufig auf reine Gesten zurück wie weibliche Benutzer.

Es lässt sich weiter festhalten, dass es bei 26 Benutzer sehr viele verschiedene Vorgehensweisen bei der inhaltlichen Auseinandersetzung mit den Aufgaben gibt, doch einige Dinge, die nichts mit der Thematik Prozessmodellierung, aber mit dem Umgang mit Multi-Touch-Technologien an sich zu tun haben, von den meisten geteilt werden. So tauchten immer wieder Begriffe und Konzepte aus dem Alltag mit herkömmlichen Computern auf, wie Kontextmenüs, die über einen Rechtsklick mit der Maus geöffnet werden sollten. Im Wesentlichen ist ein Multi-Touch-Programm auf dem Apple iPad im mentalen Modell des Benutzers ein Programm, welches auf einem Computer läuft, der wiederum mit Maus und Tastatur zu bedienen ist. Es gab dabei auch nur einen Benutzer, der das Apple iPad längerfristig in die Hand nahm und einen der es auf ein Mäppchen lehnte. Alle anderen fassten es über die meiste Zeit nicht an, außer wenn sie die Oberfläche berührten. Ein Benutzer benutzte durchgehend seinen Mittelfinger für die Tipps, während alle anderen fast ausschließlich ihren Zeigefinger benutzten. Aber keiner benutzte, wie es in Kapitel 2.1.2 angesprochen wurde, zwei oder mehr Finger gleichzeitig für einen einfachen Tipp. Dies geschah nur bei der Pinch-Geste. Es gab aber auch keine besonders großen Elemente, die dies provoziert hätten. Sie waren aber auch nicht zu klein, denn die Treffer-Quote war sehr hoch bis auf vereinzelte unbeabsichtigte Berührungen. Beidhändige Gesten kamen nur vereinzelt und nur für das Zoomen vor. Keiner kam von sich aus auf die Idee eine andere Eingabemethode, wie "Take-Off" oder "Shift", zu wählen als das direkte Tippen auf das gewünschte Element, was auch nicht anders zu erwarten war. Fast keiner der Benutzer wünschte sich von sich aus einen Bedienstift. Keiner der Beteiligten versuchte per Spracheingabe eine Funktion auszulösen. Es wurde aber auch nicht extra darauf hingewiesen, dass dies möglich sei. Diese Technologie ist noch zu wenig verbreitet, als dass jemand damit rechnet, diese zur Verfügung zu haben. Etwas problematisch bei der Untersuchung war der Einsatz des Zeichen-Programms als Aufzeichnungsmethode in der Hinsicht, dass ausgeführte Gesten und zusätzliche zur Veranschaulichung dienende Zeichnungen der Benutzer ineinander flossen, so dass diese nicht immer auseinander gehalten werden konnten. Offen bleibt auch der Einfluss der Sichtbarmachung der Spur auf die Zeichenfreudigkeit der Benutzer.

5 Zusammenfassung

Diese Arbeit stellt die Möglichkeiten von Multi-Touch-Gesten für ein Programm im Bereich Geschäftsprozessmodellierung vor. Es werden dabei aufbauend auf Grundlagenliteratur zur Multi-Touch-Technologie verschiedene Ansätze zur Bedienung eines fiktiven Programms entwickelt und diskutiert. Dabei werden Varianten aus zwei Extremen, nämlich zum einen die alleinige Bedienung über Gesten und zum anderen die alleinige Bedienung über Menüs, vorgestellt, diskutiert und zu einer Hybridlösung kombiniert. Demnach hat die hier vorgestellte Gesten-basierte Variante aufgrund des Lernens der Gesten eine höhere Einstiegshürde, vermittelt aber einen direkteren Kontakt mit der Multi-Touch-Oberfläche und erhöht die Arbeitsgeschwindigkeit. Die menügeführten Varianten bilden einen Gegenpol dazu. Die Bedienung ist schneller ersichtlich, verlangsamt aber durch häufigere Tipps und längere Wege den Arbeitsfluss des Modellierers. Die Hybridlösung versucht die Interaktion durch Kontextmenüs und implizite Gesten nahe am Prozess zu halten und somit ein gutes Verhältnis zwischen Geschwindigkeit und Einfachheit zu gewinnen.

Unabhängig von den eigenen Ergebnissen, wird in einer experimentellen Untersuchung evaluiert, welche Bedienung von einzelnen Benutzern als richtig empfunden wird. Dabei wird festgestellt, dass es eine Rolle spielt, ob der Benutzer bereits von Multi-Touch-Geräten vorgeprägt ist. Ist er es nicht, versteht er Multi-Touch eher als eine Plattform auf dem er seine Vorstellungen im Sinne der Prozessmodellierung skizzieren kann. Bei erfahrenen Benutzern tauchen vor allem typische Computer-Metaphern auf. Sie benutzen vermehrt Gesten und Menüstrukturen.

Insgesamt verspricht die Multi-Touch-Technologie weiteres Potential für die Verbreitung und Verbesserung der Geschäftsprozessmodellierung aufgrund ihrer Mobilitäts- und Gruppenarbeitstauglichkeit, welche aber noch weitergehend erforscht werden müssen.

5 Zusammenfassung

A Anhang

A.1 Ergebnisse der experimentellen Untersuchung im Überblick

Legende:

F1: Aktivität einfügen

F2: Element benennen

F3: Verzweigungsblock einfügen

F4: Verzweigung einfügen

F5: Datenelement einfügen

F6: Lese- und Schreib-Kanten setzen

F7: Subprozess bilden und auflösen

F8: Element löschen

Z: Zeichnerische Lösung

G: Reine Geste

M: Menügeführte Lösung

Tätigk.: Tätigkeit

ST: Student

MA: Mitarbeiter des Instituts Datenbanken und Informationssysteme der Universität Ulm

Sonst: Sonstige

Erfahr.: Erfahrung mit Umgang mit Multi-Touch-Geräten

Geschl.: Geschlecht

m: männlich

w: weiblich

A Anhang

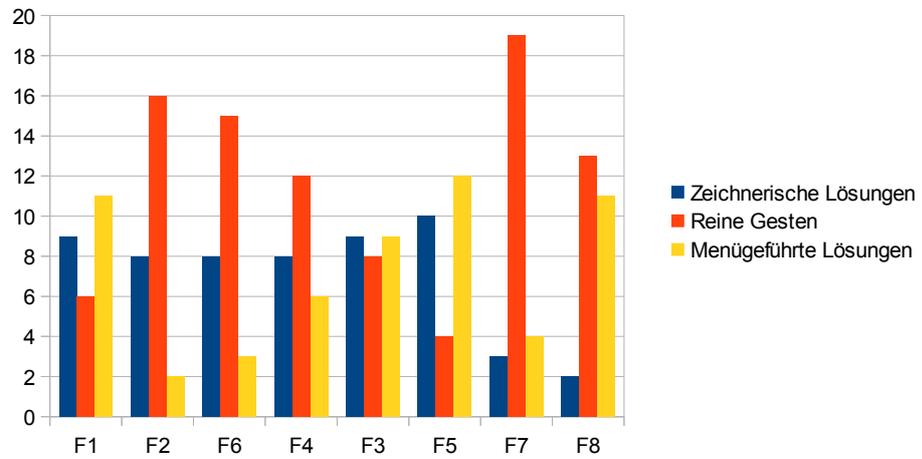


Abbildung A.1: Gesamtergebnis der Bedienkonzepte im Überblick

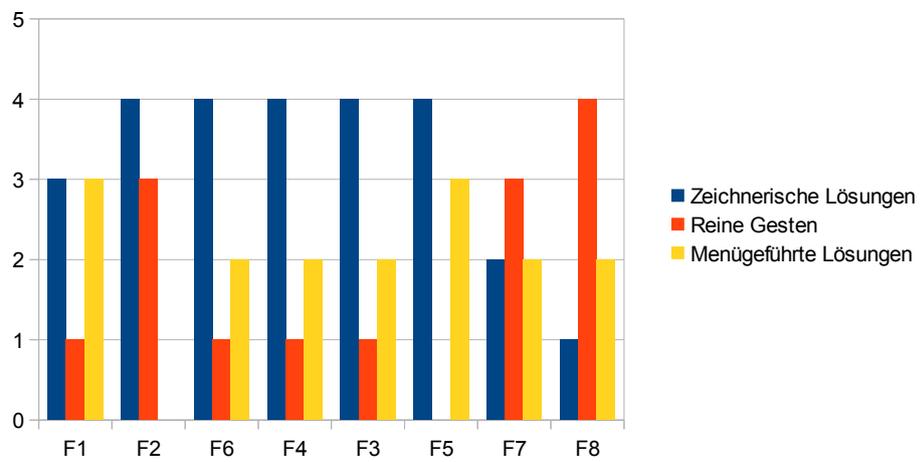


Abbildung A.2: Verteilung der Bedienkonzepte bei den weiblichen Benutzer

A.1 Ergebnisse der experimentellen Untersuchung im Überblick

Tabelle A.1: Ergebnisse der experimentellen Untersuchung im Überblick

Benutzer	F1	F2	F6	F4	F3	F5	F7	F8	Tätigk.	Erfahr.	Geschl.
1	M	G	G	G	G	G	G	G	ST	ja	m
2	Z	G	Z	Z	Z	Z	Z	G	ST	ja	w
3	Z	G	Z	Z	Z	Z	Z	G	ST	nein	w
4	M	G	G	G	G	M	G	G	ST	ja	m
5	G	G	G	G	G	M	G	G	ST	ja	m
6	Z	G	Z	Z	Z	Z	G	G	MA	nein	m
7	G	Z	Z	Z	Z	Z	G	Z	MA	nein	w
8	M	G	G	G	G	M	G	G	MA	nein	w
9	G	M	G	G	G	G	G	Z	MA	nein	m
10	M	Z	G	G	Z	Z	G	M	ST	nein	m
11	M	G	G	G	M	M	G	G	MA	ja	m
12	M	Z	M	M	M	M	G	M	MA	ja	m
13	M	M	G	M	M	M	G	M	MA	ja	m
14	M	Z	G	G	G	G	G	M	ST	nein	m
15	Z	G	Z	Z	Z	Z	G	G	ST	nein	m
16	Z	Z	Z	Z	Z	Z	G	G	Sonst	nein	w
17	G	G	G	G	G	G	G	M	ST	ja	m
18	Z	G	Z	Z	Z	Z	Z	M	ST	nein	m
19	M	Z	M	M	M	M	M	M	Sonst	nein	w
20	Z	G	Z	Z	Z	Z	G	M	ST	ja	m
21	G	G	G	G	G	Z	G	G	ST	ja	m
22	G	G	G	M	M	M	M	M	ST	ja	m
23	M	Z	M	M	M	M	M	M	ST	ja	w
24	Z	Z	G	M	M	M	G	M	ST	ja	m
25	M	G	G	G	M	M	M	G	ST	ja	m
26	Z	G	G	G	M	M	G	G	ST	nein	m

Tabelle A.2: Zusammenfassung der Ergebnisse Allgemein

Benutzer	F1	F2	F6	F4	F3	F5	F7	F8	Tätigk.	Erfahr.	Geschl.
Zeichnerische Lösung	3	3	2	2	2	3	1	0	16	14,29%	
Reine Geste	5	10	10	7	5	2	10	7	56	50,00%	
Menügeführte Lösung	6	1	2	5	7	9	3	7	40	35,71%	

A Anhang

Tabelle A.3: Ergebnisse der experimentellen Untersuchung nach Geschlecht sortiert

Benutzer	F1	F2	F6	F4	F3	F5	F7	F8	Tätigk.	Erfahr.	Geschl.
1	M	G	G	G	G	G	G	G	ST	ja	m
4	M	G	G	G	G	M	G	G	ST	ja	m
5	G	G	G	G	G	M	G	G	ST	ja	m
6	Z	G	Z	Z	Z	Z	G	G	MA	nein	m
9	G	M	G	G	G	G	G	Z	MA	nein	m
10	M	Z	G	G	Z	Z	G	M	ST	nein	m
11	M	G	G	G	M	M	G	G	MA	ja	m
12	M	Z	M	M	M	M	G	M	MA	ja	m
13	M	M	G	M	M	M	G	M	MA	ja	m
14	M	Z	G	G	G	G	G	M	ST	nein	m
15	Z	G	Z	Z	Z	Z	G	G	ST	nein	m
17	G	G	G	G	G	G	G	M	ST	ja	m
18	Z	G	Z	Z	Z	Z	Z	M	ST	nein	m
20	Z	G	Z	Z	Z	Z	G	M	ST	ja	m
21	G	G	G	G	G	Z	G	G	ST	ja	m
22	G	G	G	M	M	M	M	M	ST	ja	m
24	Z	Z	G	M	M	M	G	M	ST	ja	m
25	M	G	G	G	M	M	M	G	ST	ja	m
26	Z	G	G	G	M	M	G	G	ST	nein	m
2	Z	G	Z	Z	Z	Z	Z	G	ST	ja	w
3	Z	G	Z	Z	Z	Z	Z	G	ST	nein	w
7	G	Z	Z	Z	Z	Z	G	Z	MA	nein	w
8	M	G	G	G	G	M	G	G	MA	nein	w
16	Z	Z	Z	Z	Z	Z	G	G	Sonst	nein	w
19	M	Z	M	M	M	M	M	M	Sonst	nein	w
23	M	Z	M	M	M	M	M	M	ST	ja	w

Tabelle A.4: Zusammenfassung der Ergebnisse der weiblichen Benutzer

Benutzer	F1	F2	F6	F4	F3	F5	F7	F8	Tätigk.	Erfahr.	Geschl.
Zeichnerische Lösung	3	4	4	4	4	4	2	1	26	46,43%	
Reine Geste	1	3	1	1	1	0	3	4	14	25,00%	
Menügeführte Lösung	3	0	2	2	2	3	2	2	16	28,57%	

A.1 Ergebnisse der experimentellen Untersuchung im Überblick

Tabelle A.5: Zusammenfassung der Ergebnisse der männlichen Benutzer

Benutzer	F1	F2	F6	F4	F3	F5	F7	F8	Tätigk.	Erfahr.	Geschl.
Zeichnerische Lösung	6	4	4	4	5	6	1	1	31	20,39%	
Reine Geste	5	13	14	11	7	4	16	9	79	51,97%	
Menügeführte Lösung	8	2	1	4	7	9	2	9	42	27,63%	

Tabelle A.6: Ergebnisse der experimentellen Untersuchung nach Erfahrung sortiert

Benutzer	F1	F2	F6	F4	F3	F5	F7	F8	Tätigk.	Erfahr.	Geschl.
3	Z	G	Z	Z	Z	Z	Z	G	ST	nein	w
6	Z	G	Z	Z	Z	Z	G	G	MA	nein	m
7	G	Z	Z	Z	Z	Z	G	Z	MA	nein	w
8	M	G	G	G	G	M	G	G	MA	nein	w
9	G	M	G	G	G	G	G	Z	MA	nein	m
10	M	Z	G	G	Z	Z	G	M	ST	nein	m
14	M	Z	G	G	G	G	G	M	ST	nein	m
15	Z	G	Z	Z	Z	Z	G	G	ST	nein	m
16	Z	Z	Z	Z	Z	Z	G	G	Sonst	nein	w
18	Z	G	Z	Z	Z	Z	Z	M	ST	nein	m
19	M	Z	M	M	M	M	M	M	Sonst	nein	w
26	Z	G	G	G	M	M	G	G	ST	nein	m
1	G	G	G	G	G	G	G	G	ST	ja	m
2	Z	G	Z	Z	Z	Z	Z	G	ST	ja	w
4	M	G	G	G	G	M	G	G	ST	ja	m
5	G	G	G	G	G	M	G	G	ST	ja	m
11	M	G	G	G	M	M	G	G	MA	ja	m
12	M	Z	M	M	M	M	G	M	MA	ja	m
13	M	M	G	M	M	M	G	M	MA	ja	m
17	G	G	G	G	G	G	G	M	ST	ja	m
20	Z	G	Z	Z	Z	Z	G	M	ST	ja	m
21	G	G	G	G	G	Z	G	G	ST	ja	m
22	G	G	G	M	M	M	M	M	ST	ja	m
23	M	Z	M	M	M	M	M	M	ST	ja	w
24	Z	Z	G	M	M	M	G	M	ST	ja	m
25	M	G	G	G	M	M	M	G	ST	ja	m

A Anhang

Tabelle A.7: Zusammenfassung der Ergebnisse der Benutzer mit Erfahrung

Benutzer	F1	F2	F6	F4	F3	F5	F7	F8	Tätigk.	Erfahr.	Geschl.
Zeichnerische Lösung	3	3	2	2	2	3	1	0	16	14,29%	
Reine Geste	5	10	10	7	5	2	10	7	56	50,00%	
Menügeführte Lösung	6	1	2	5	7	9	3	7	40	35,71%	

Tabelle A.8: Zusammenfassung der Ergebnisse der Benutzer ohne Erfahrung

Benutzer	F1	F2	F6	F4	F3	F5	F7	F8	Tätigk.	Erfahr.	Geschl.
Zeichnerische Lösung	6	5	6	6	7	7	2	2	41	42,71%	
Reine Geste	2	6	5	5	3	2	9	6	38	39,58%	
Menügeführte Lösung	4	1	1	1	2	3	1	4	17	17,71%	

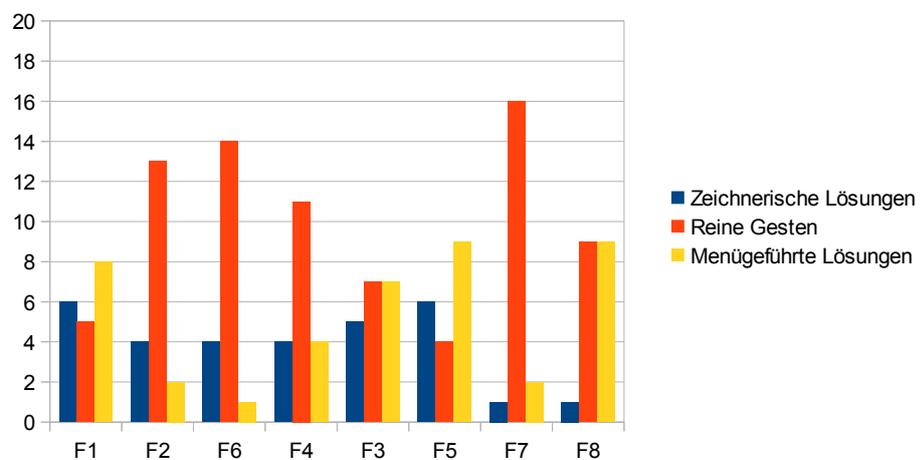


Abbildung A.3: Verteilung der Bedienkonzepte bei den männlichen Benutzer

A.1 Ergebnisse der experimentellen Untersuchung im Überblick

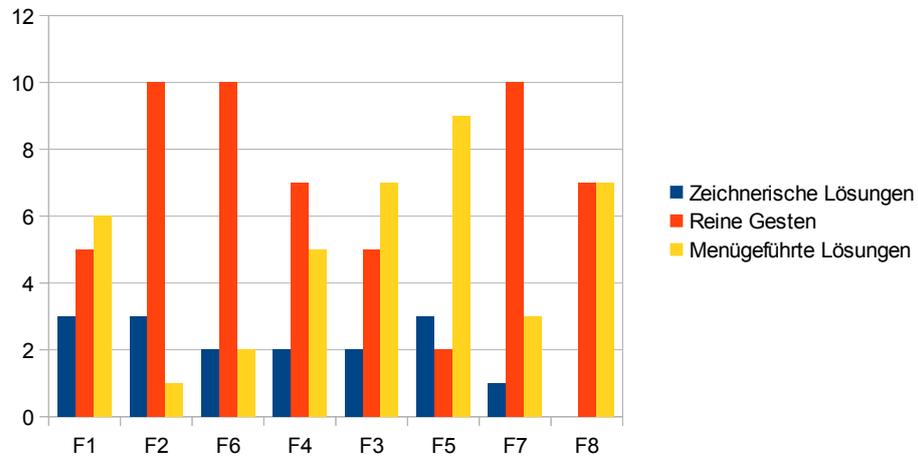


Abbildung A.4: Verteilung der Bedienkonzepte bei den erfahrenen Benutzer

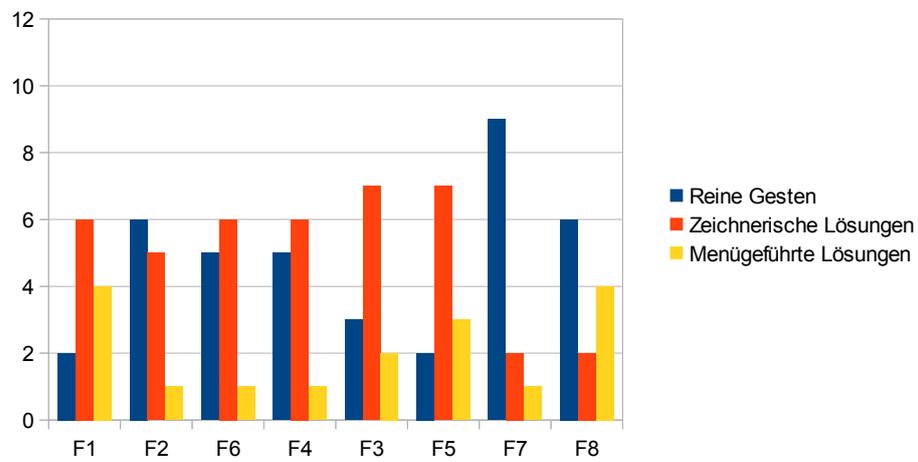


Abbildung A.5: Verteilung der Bedienkonzepte bei den unerfahrenen Benutzer

Literaturverzeichnis

- [1] *Apple files iPhone gesture patent.* <http://www.macnn.com/blogs/2007/08/02/apple-files-iphone-gesture-patent.html>. Version: September 2011
- [2] *Business Process Model and Notation (BPMN).* <http://www.omg.org/spec/BPMN/2.0/PDF>. Version: Oktober 2011
- [3] *Gebärdensprache.* <http://de.wikipedia.org/wiki/Gebärdensprache>. Version: Oktober 2011
- [4] *Jeff Han zeigt seinen bahnbrechenden Touchscreen.* http://www.ted.com/talks/lang/ger/jeff_han_demos_his_breakthrough_touchscreen.html. Version: Oktober 2011
- [5] *LedaSila - Lexical Database for Sign Languages.* <http://ledasila.uni-klu.ac.at/TPM/public/signword.asp?SID=10911>. Version: Oktober 2011
- [6] *Modellierung beschleunigen.* <http://www.ariscommunity.com/help/aris-express/36080/de>. Version: Oktober 2011
- [7] *Multitasking Gesten fürs iPad.* <http://www.apple.com/de/ios/features.html#more>. Version: Oktober 2011
- [8] BIELINIS, Stasys: *Apple wants to teach us Multi-Touch gesture language.* <http://www.unwiredview.com/2008/07/03/apple-wants-to-teach-us-multi-touch-gesture-language/>. Version: September 2011
- [9] BURMESTER, Micheal ; KOLLER, Franz ; HÖFLACHER, Christine: *Touch it, move it, scale it - Multitouch. Untersuchung zur Usability und Erlernbarkeit von Multitouch-Interaktionen am Beispiel des Multitouch-Tisches Microsoft Surface.* <http://www.uid.com/wissen/uid-veroeffentlichungen/>. Version: November 2009
- [10] CHERUBINI, Mauro ; VENOLIA, Gina ; DELINE, Rob ; KO, Andrew J.: *Let's go to the whiteboard: how and why software developers use drawings.* In: *Proceedings of the*

Literaturverzeichnis

- SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM, 2007 (CHI '07), S. 557–566
- [11] DIETZ, Paul ; LEIGH, Darren: DiamondTouch: a multi-user touch technology. In: *Proceedings of the 14th annual ACM symposium on User interface software and technology*. New York, NY, USA : ACM, 2001 (UIST '01), S. 219–226
- [12] ELIAS, John G. ; WESTERMAN, Wayne C. ; HAGGERTY, Myra M.: *Multi-touch gesture dictionary*. <http://appft1.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PG01&p=1&u=%2Fnetahtml%2FPTO%2Fsrchnum.html&r=1&f=G&l=50&s1=%2220070177803%22.PG NR.&OS=DN/20070177803&RS=DN/20070177803>. Version: Januar 2007
- [13] EPPS, Julien ; LICHMAN, Serge ; WU, Mike: A study of hand shape use in tabletop gesture interaction. In: *CHI '06 extended abstracts on Human factors in computing systems*. New York, NY, USA : ACM, 2006 (CHI EA '06), S. 748–753
- [14] FARHI, Paul: *CNN Hits The Wall for the Election*. <http://www.washingtonpost.com/wp-dyn/content/article/2008/02/04/AR2008020402796.html>. Version: Oktober 2011
- [15] FRISCH, Mathias ; HEYDEKORN, Jens ; DACHSELT, Raimund: Investigating multi-touch and pen gestures for diagram editing on interactive surfaces. In: *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*. New York, NY, USA : ACM, 2009 (ITS '09), S. 149–156
- [16] HALL, Anthony D. ; CUNNINGHAM, James B. ; ROACHE, Richard P. ; COX, Julie W.: Factors affecting performance using touch-entry systems: Tactual recognition fields and system accuracy. In: *Journal of Applied Psychology* 73 (1988), Nr. 4, S. 711–720
- [17] HAN, Jefferson Y.: Low-cost multi-touch sensing through frustrated total internal reflection. In: *Proceedings of the 18th annual ACM symposium on User interface software and technology*. New York, NY, USA : ACM, 2005 (UIST '05), S. 115–118
- [18] INC., Pinger: *Doodle Buddy for iPad – Paint, Draw, Scribble, Sketch – It's Addictive!* <http://itunes.apple.com/de/app/id364201083?mt=8>. Version: Oktober 2011
- [19] KAY, Alan C.: A Personal Computer for Children of All Ages. In: *Proceedings of the ACM annual conference - Volume 1*. New York, NY, USA : ACM, 1972 (ACM '72)

- [20] LE HONG, Sylvia ; BIESTERFELDT, Jakob: *Weltweit berührt - Studie zur Untersuchung kultureller Unterschiede und Gemeinsamkeiten bei der gestenbasierten Bedienung von Multitouch-Oberflächen*. <http://www.uid.com/wissen/uid-veroeffentlichungen/>. Version: Juli 2010
- [21] LEGANCHUK, Andrea ; ZHAI, Shumin ; BUXTON, William: Manual and cognitive benefits of two-handed input: an experimental study. In: *ACM Trans. Comput.-Hum. Interact.* 5 (1998), Dezember, S. 326–359
- [22] LIN, Min ; GOLDMAN, Rich ; PRICE, Kathleen J. ; SEARS, Andrew ; JACKO, Julie: How do people tap when walking? An empirical investigation of nomadic data entry. In: *Int. J. Hum.-Comput. Stud.* 65 (2007), September, S. 759–769
- [23] OPPERMANN, Reinhard ; MURCHNER, Bernd ; REITERER, Harald ; KOCH, Manfred: *Softwareergonomische Evaluation - Der Leitfaden EVADIS II*. <http://www.ariscommunity.com/help/aris-express/36080/de>. Version: 1992
- [24] ORDING, Bas ; FORSTALL, Scott ; CHRISTIE, Greg ; LEMAY, Stephen O. ; CHAUDHRI, Imran: *Portable electronic device with multi-touch input*. <http://stadium.weblogsinc.com/engadget/files/apple-ptz-patent.pdf>. Version: Oktober 2010
- [25] POTTER, R. L. ; WELDON, L. J. ; SHNEIDERMAN, B.: Improving the accuracy of touch screens: an experimental evaluation of three strategies. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM, 1988 (CHI '88), S. 27–32
- [26] REICHERT, Manfred ; DADAM, Peter (Hrsg.): *Dynamische Ablaufänderungen in Workflow-Management-Systemen*. <http://dbis.eprints.uni-ulm.de/433/>. Version: Juli 2000
- [27] SALANDER, Anders: Using table-top displays to enhance creativity and collaboration in design projects. In: *Proceedings of Umea's 13th Student Conference in Computer Science*, 2009, S. 125–142
- [28] VOGEL, Daniel ; BAUDISCH, Patrick: Shift: a technique for operating pen-based interfaces using touch. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM, 2007 (CHI '07), S. 657–666
- [29] WANG, C. ; LEE, H. ; SMEATON, A.F.: Designing interaction for a multi-touch wall. (2010)

Literaturverzeichnis

- [30] WANG, Feng ; REN, Xiangshi: Empirical evaluation for finger input properties in multi-touch interaction. In: *Proceedings of the 27th international conference on Human factors in computing systems*. New York, NY, USA : ACM, 2009 (CHI '09), S. 1063–1072
- [31] WOBROCK, Jacob O. ; MORRIS, Meredith R. ; WILSON, Andrew D.: User-defined gestures for surface computing. In: *Proceedings of the 27th international conference on Human factors in computing systems*. New York, NY, USA : ACM, 2009 (CHI '09), S. 1083–1092

Name: Benjamin Rudner

Matrikelnummer: 651595

Erklärung

Ich erkläre, dass ich die Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Ulm, den

Benjamin Rudner