

# Aufgabenmodellierung am Multi-Touch-Tisch

Jens Eggers, Adrian Hülsmann, Gerd Szwillus

Institut für Informatik, Fachgruppe Mensch-Computer-Interaktion, Universität Paderborn

## Zusammenfassung

In frühen Phasen der Entwicklung von Benutzungsschnittstellen kommen zunehmend Aufgabenmodelle zum Einsatz, um die Handlungen der Benutzer vor und nach Einführung des Systems zu dokumentieren und zu modellieren. Die entstehenden Aufgabenhierarchien sind typischerweise große Graphen, die computerunterstützt, zudem oftmals kollaborativ in kleinen Teams, erstellt und bearbeitet werden. Derartige Aufgaben sind prädestiniert dafür, an großen interaktiven Displays durchgeführt zu werden. In diesem Paper diskutieren wir ein Konzept zur Unterstützung der kollaborativen Bearbeitung von Aufgabenmodellen an einem Multi-Touch-Tisch, das zukünftig umgesetzt wird.

## 1 Einleitung

In modellbasierten Ansätzen zur Entwicklung von Benutzungsschnittstellen werden Aufgabenmodelle eingesetzt (Szwillus 2011), um die Tätigkeiten der Benutzer vor Einführung eines Systems (Ist-Modell) und danach (Soll-Modell) zu dokumentieren. Diese Modelle sind in realistischen Fällen große Graphen mit einer unterliegenden Hierarchie. Da außerdem typischerweise verschiedene Rollen zu berücksichtigen sind, bietet sich eine kollaborative Bearbeitung dieser Modelle auf einem Multi-Touch-Tisch an. In einem ersten Schritt soll hier eine Konzeption für eine derartige Modellierungsumgebung vorgestellt werden, die für den Multi-Touch-Tisch (MTT) am Institut für Informatik der Universität Paderborn entwickelt werden soll.

Der verwendete MTT "Mister T" zeichnet sich durch ein 1,60 x 0,90 Meter großes, hochauflösendes Display aus. Die mit Hilfe des LLP-Prinzips (Laser-Light-Plane) implementierte und auf optischem Tracking basierende Berührungserkennung erlaubt eine zuverlässige und schnelle Identifikation vieler Touchpunkte, sodass die simultane Benutzung durch mehrere Benutzer problemlos möglich ist. Tangibles (physikalisch begreifbare Objekte) können durch ein eigens realisiertes Framework von unserem System erkannt und mobile Endgeräte (Smartphones, Tablets) als erweiterte Interaktionsartefakte mit dem System gekoppelt werden.

## 2 Interaktionen der Aufgabenmodellierung

Die konzeptionellen Vorüberlegungen stützen sich ab auf den Pseudo-Standard für die Aufgabenmodellierung, nämlich den CTT-Ansatz von Paternò (Paternò 2000). Demnach ist ein Aufgabenmodell strukturell eine Hierarchie, in der Aufgaben in Unteraufgaben dekomponiert werden, ergänzt um strukturell anders geartete Kanten zwischen benachbarten Knoten zur Spezifikation der zeitlichen Beziehungen (sog. temporalen Relationen) zwischen diesen. Nach diesem Ansatz entsprechen Aufgabenmodelle dem Konzept der „Knoten-Linie“-Diagrammen, für die in anderen Studien bereits Interaktionsmuster vorgeschlagen wurden (z.B. Frisch et al. 2009) und an denen wir uns orientiert haben.



Abbildung 1: Knoteninteraktionen und -darstellungen

So kann ein neuer Knoten durch eine Hold-Geste (Abb. 1 a) direkt am MTT oder mit einem Smartphone durch eine Stempel-Geste des Gerätes auf das Display des MTTs erzeugt werden. Anders als in der graphischen Entwicklungsumgebung CTTE (Paternò 2013) sollen – wie etwa auch bei K-MADe (Baron et al. 2006) – mehr Aufgabeneigenschaften direkt am Knoten sichtbar sein (Abb. 1 b), statt dafür erst einen Dialog öffnen zu müssen; dies betrifft den Hinweis auf Vor- und Nachbedingungen (Pfeilspitzen) und angebundene Aufgabenobjekte (Würfel). Auch sollte in den Knoten selbst direkt das Ein- und Ausklappen des darunterliegenden Unterbaumes anwählbar sein (+/-).

Knoteneigenschaften können zum einen durch Berühren mit dem Finger über ein Pie-Menü (Baron et al. 2006) am Knoten selbst, zum anderen durch Berühren mit einem mobilen Endgerät direkt auf dessen Display verändert werden (Abb. 1 c). Letzteres dient der Verlagerung der Texteingabe, wodurch Bildschirmplatz (für eventuelle virtuelle Tastaturen) auf dem MTT gespart und Interaktionsflüsse (vor allem in kooperativen Bearbeitungen) weniger gestört würden. Für die Knoteneigenschaften schlagen wir eine „Preisschild“-Metapher vor, die das „Anhaften“ dieser Eigenschaften stärker verdeutlichen soll (Abb. 1d).

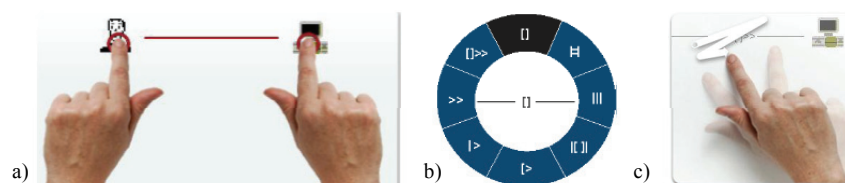


Abbildung 2: Gesten beim Arbeiten mit Kanten und globales Undo/Redo

Kanten werden durch das gleichzeitige Berühren zweier Knoten zwischen diesen erzeugt (Abb. 2 a). Anschließend wird die temporale Relation bei Kanten zwischen Kindknoten der-

selben Oberaufgabe wiederum über ein Pie-Menü festgelegt (Abb. 2 b). Die Reihenfolge der Relation ergibt sich dabei aus der visuellen Anordnung der Knoten (von links nach rechts). Eine Erase-Geste (Abb. 2 c) dient dem Entfernen nicht gewünschter Elemente und kann auf Knoten und Kanten angewandt werden. Des Weiteren kann über die etablierte Drag-Geste entweder das gesamte Diagramm (bei Initiieren der Geste auf dem Hintergrund) oder ein einzelner Knoten (bei Initiierung direkt auf dem Knoten) verschoben werden, wobei auch der gesamte Unterbaum darunter entsprechend verschoben wird, um die relative Anordnung zu erhalten. Da wir von potentiell sehr großen Diagrammen ausgehen, wird auch die bekannte 2-Finger-Pinch-Zoom-Geste als Möglichkeit der Vergrößerung bzw. Verkleinerung von Diagrammen implementiert.



Abbildung 3: Undo/  
Redo

Eine Besonderheit stellt die von uns vorgeschlagene, kombinierte Undo-Redo-Geste dar (Abb. 3). Sie wird als 3-Finger-Hold und Tap der verbleibenden Finger realisiert. Während die 3 Finger halten, fordert mehrfaches Tippen des Daumens (links) aufeinander folgende Undo-Operationen an, während ein Wippen nach rechts zum kleinen Finger eine Redo-Operation auslöst. Somit kann durch diese intuitive Wippgeste in der Historie schnell zurück- und vorgesprungen werden.

Um weitere Hilfe zum Umgang mit großen Diagrammen anzubieten, adaptieren wir die Off-Screen-Visualisierung (Frisch et al. 2009) in den Bereich der Aufgabenmodelle (Abb. 4). Hierbei werden am Rand des Bildschirms jeweils Indikatoren angebracht, die andeuten, was in der entsprechenden Richtung außerhalb des Bildschirmausschnitts liegt.

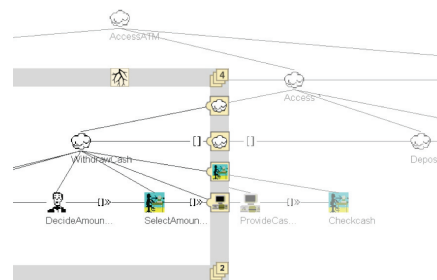


Abbildung 4: Off-Screen-Visualisierung

### 3 Kooperative Aspekte der Aufgabenmodellierung

CTT bietet die Möglichkeit kooperative Aufgabenmodelle zu beschreiben, diese beinhalten verschiedene Benutzerrollen. Beim Bearbeiten dieser kooperativen Modelle entstehen mehrere Aufgabenbäume parallel; diese werden in einer übergeordneten Aufgabenhierarchie zur Steuerung der Kooperation miteinander in Beziehung gesetzt. Aus Platzgründen beschränken wir uns in diesem Beitrag auf die Skizzierung der Interaktionen und gehen nicht auf die entstehenden prozessualen Fragen ein. Für die Bearbeitung der beteiligten Aufgabenmodelle im Team sehen wir die Möglichkeit vor, Diagrammteile zumindest zeitweise auseinander zu reißen (*tear*), dann gemäß der im Aufgabenmodell beteiligten Benutzerrollen arbeitsteilig im

Team durch verschiedene Entwickler bearbeiten zu lassen und anschließend wieder zusammenzuführen (*merge*).

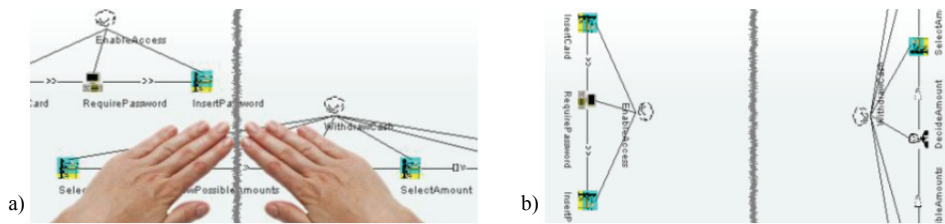


Abbildung 5: Tear-Operation

Dabei kann die tear-Operation (Abb. 5 a) derart eingestellt werden, dass sie die getrennten Teile direkt so dreht, dass bis zu vier Bearbeiter die Diagramme in der richtigen Ausrichtung vor sich haben (Abb. 5 b); die merge-Operation arbeitet entsprechend. Über diese reinen Graphikmanipulationen hinaus kann auch eine fachliche Trennung der Aufgabenbäume in die verschiedenen Rollen und/oder den übergeordneten Rollenmanager erfolgen.

Die hier beschriebenen Konzepte wurden im Rahmen einer Masterarbeit (Eggers 2013) entwickelt, dort sind auch weitere Details nachzulesen. Vieles wurde bereits experimentell erprobt und wird in den nächsten Monaten in einer kompletten Aufgabenmodellierungsumgebung für den MTT "Mister T" implementiert.

### Literaturverzeichnis

- Baron, M.; Lucquiaud, V.; Autard, D.; Scapin, D. L. (2006): K-MADE: un environnement pour le noyau du modèle de description de l'activité. In: Jean-Marc Robert und David Bertrand (Hg.): Proceedings of the 18th International Conference of the Association Francophone d'Interaction Homme-Machine. New York, ACM, S. 287–288.
- Eggers, Jens (2013): Entwicklung eines Konzeptes zur Erstellung von Aufgabenmodellen an einem interaktiven Display. Masterarbeit. Universität Paderborn, Paderborn. Institut für Informatik.
- Frisch, Mathias; Heydekorn, Jens; Dachzelt, Raimund (2009): Diagram Editing on interactive displays using multi-touch and pengestures. In: ITS. Proceedings of the ACM International Conference in Interactive Tabletops and Surfaces. ITS'09. Calgary, AB, Canada, 23.-25.11.2009, S. 149–156.
- Paternò, Fabio (2000): Model-based design and evaluation of interactive applications. London: Springer (Applied computing).
- Paternò, Fabio (Hg.) (2013): ConcurTaskTreesEnvironment. HIIS Laboratory, Istituto di Scienza e Tecnologia dell'Informazione. Online verfügbar unter <http://giove.cnuce.cnr.it/ctte.html>, zuletzt geprüft am 1.3.13.
- Szwillus, Gerd (2011): Task Models in the Context of User Interface Development. In: Heinrich Hussmann, Gerrit Meixner und Detlef Zuehlke (Hg.): Model-driven development of advanced user interfaces. Berlin: Springer, S. 277–303.

### Kontaktinformationen

Prof. Dr. Gerd Szwillus

Universität Paderborn, Fakultät für Elektrotechnik, Informatik und Mathematik  
Institut für Informatik, Fürstenallee 11, 33102 Paderborn