

Assistierte Informationsanzeige in Smart Meeting Rooms

Dissertation
zur Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

der Fakultät für Informatik und Elektrotechnik
der Universität Rostock



vorgelegt von
Christian Eichner
geboren am 05.01.1987 in Rostock
wohnhaft in Rostock
Rostock, 14. Dezember 2016

Betreuer

- ▷ Prof. Dr.-Ing. habil. Heidrun Schumann
Universität Rostock, Deutschland

Gutachter

- ▷ Prof. Dr.-Ing. habil. Heidrun Schumann
Universität Rostock, Deutschland
- ▷ Prof. Dr. Sebastian Boring
Universität Kopenhagen, Dänemark
- ▷ Prof. Dr. Oliver Staadt
Universität Rostock, Deutschland

Datum der Verteidigung

- ▷ 20.07.2017

Schlagwörter / Keywords

information presentation, user assistance, smart meeting rooms, multi display environments

Klassifikation / Classification (ACM CCS 2012)

*Human-centered computing → Information visualization, Human-centered computing
→ Ubiquitous and mobile computing theory, concepts and paradigms, Human-centered
computing → Visualization application domains*

Copyright © 2016 by Christian Eichner

Kurzfassung Das Anzeigen von visuellen Inhalten ist ein wichtiges Mittel, um Präsentationen und Diskussionen zu unterstützen. Um unterschiedliche Sichten auf Daten oder Inhalte von verschiedenen Nutzern gleichzeitig anzuzeigen, bietet sich die Nutzung von Multi-Display-Umgebungen (MDE) an. Smart Meeting Rooms (SMRs) stellen so eine Umgebung bereit und ermöglichen darüber hinaus auch neue Assistenzansätze, welche die Nutzer beim Gestalten und Anpassen der Informationsanzeige vielfältig unterstützen können. Allerdings adressieren diese Assistenzansätze für SMRs bisher kaum das Anzeigen von Informationen für Präsentationen und Diskussionen.

Ziel dieser Dissertation ist es deshalb, neuartige Konzepte für eine assistierte Informationsanzeige in Smart Meeting Rooms bereitzustellen. Dafür wird zunächst eine umfassende Modellierung erarbeitet, welche als Basis für die entwickelten Lösungsansätze dient. Basierend auf diesen Modellen wird das nutzergesteuerte Austauschen von Inhalten zwischen mehreren persönlichen Geräten und das automatische Verteilen und Anordnen von mehreren Views auf unterschiedlichen Displays adressiert. Darüber hinaus wird ein Ansatz zur interaktiven, bedarfsgerechten Erzeugung von neuen Darstellungen und zu ihrer spontanen Integration in eine bestehende Anzeige vorgestellt. Weiterhin werden Hilfsmittel eingeführt, um auch die manuelle Nutzerinteraktion zum Anpassen der Anzeige zu vereinfachen und die Suche nach semantisch passenden Inhalten zu unterstützen. Abschließend wird eine situationsangepasste Assistenz bereitgestellt, welche die entwickelten Ansätze automatisch auf die aktuelle Situation abstimmt.

Abstract The display of visual content is an important means to support presentations and discussions. For this, utilizing multi-display-environments (MDE) seems promising in order to show different aspects of data simultaneously and to combine pieces of content from various users. Smart Meeting Rooms (SMRs) provide such an environment but also allow for new assistance approaches, which can support users designing and adapting the display of information. However, current approaches for SMRs barely focus on the display of information with regard to presentations and discussions.

Therefore, the goal of this thesis is to provide new solutions for an assisted information presentation in smart meeting rooms. For this, comprehensive models are introduced which serve as a basis for further approaches. Based on these models, the exchange of content between multiple personal devices and the automatic distribution and arrangement of multiple views on different displays are addressed. Furthermore, a new way is introduced that allows users to interactively generate new views on demand and to seamlessly enhance an existing presentation with these additional views. Moreover, further assistance is introduced in order to simplify the required manual interaction to change the display of information and to support the search for semantically related content. Finally, an assistance approach also addresses different situations during a meeting by automatically adapting the developed means according to the current situation.

Danksagung

An dieser Stelle möchte ich mich bei allen bedanken, die mich beim Anfertigen meiner Dissertation unterstützt haben. An erster Stelle danke ich Prof. Heidrun Schumann für die viele zielorientierte und konstruktive Kritik an meiner Arbeit. Ihr großer Einsatz bei meiner Betreuung hat auf mich sehr motivierend gewirkt. Ebenso möchte ich mich bei Prof. Sebastian Boring und Prof. Oliver Staadt für die bereitwillige Übernahme der Begutachtung dieser Arbeit bedanken.

Darüber hinaus danke ich Dr.-Ing. habil. Christian Tominski, Martin Nyolt, Dr.-Ing. Stefan Gladisch, Dr.-Ing. Axel Radloff und Dr.-Ing. Hans-Jörg Schulz für viele inspirierende Diskussionen und fachliche Unterstützung.

Nicht zuletzt danke ich meinen Eltern und meinen Freunden Arne und Arianne, die mich während der nicht immer einfachen Zeit moralisch unterstützt haben.

Inhaltsverzeichnis

1. Einleitung	13
1.1. Motivation und Zielstellung	13
1.2. Problembeschreibung	15
1.3. Ergebnisse	16
1.3.1. Modellentwicklung	16
1.3.2. Nutzerinterface	17
1.3.3. Assistenz	19
1.3.4. Publikationen	20
1.4. Aufbau der Arbeit	21
2. Grundlagen und Stand der Forschung	23
2.1. Smart Environments	23
2.1.1. Smart Meeting Rooms	23
2.1.2. Software-Komponenten für Smart Meeting Rooms	24
2.1.3. Szenario des Smart Meeting Rooms in Rostock	26
2.1.4. Informationsanzeige in Smart Meeting Rooms	28
2.2. Informationsanzeige in Multi-Display Environments	34
3. Modellentwicklung	39
3.1. Präsentationsgraph	40
3.1.1. Problembeschreibung	40
3.1.2. Modellbeschreibung	41
3.1.3. Ergebnisse	43
3.2. Semantischer Graph	44
3.2.1. Problembeschreibung	45
3.2.2. Modellbeschreibung	46
3.2.3. Ergebnisse	48
3.3. Interaktionsgraph	49
3.3.1. Problembeschreibung	49
3.3.2. Modellbeschreibung	51
3.3.3. Ergebnisse	52

4. Assistierte Präsentation	55
4.1. Nutzer-Interface zur Beschreibung der Präsentation	55
4.2. Layout-Berechnung zur Anzeige der Präsentation	62
4.3. Spontan generierte Views	71
4.3.1. Exploration von Bilddaten	79
4.3.2. Visuelle Datenexploration von Volumendaten	81
4.3.3. Web-basiertes Bereitstellen neuer Inhalte	90
5. Assistierte Diskussion	95
5.1. Meta-Analyse und Wiederherstellen von Interaktionen	96
5.1.1. Visualisierung des Interaktionsgraphen	97
5.1.2. Analyse des Interaktionsgraphen	100
5.1.3. Undo & Redo	103
5.2. Unterstützte Suche nach semantisch passenden Inhalten	106
5.2.1. Visualisierung des Semantischen Graphen	108
5.2.2. Assistenz bei der Suche nach Inhalten	113
6. Integration in den Smart Meeting Room	125
6.1. Szenarien und Situationen	125
6.1.1. Situations- und Intentionserkennung	125
6.1.2. Untersuchte Szenarien und zugehörige Anforderungen	127
6.2. Situationsangepasste Assistenz	129
6.2.1. Automatische Interface-Anpassungen	130
6.2.2. Automatische Layout-Anpassungen	132
6.2.3. Automatisches Einblenden von Semantischen Informationen	135
7. Zusammenfassung und Ausblick	139
7.1. Zusammenfassung	139
7.2. Ausblick	142
A. Layout-Berechnung	147
Literaturverzeichnis	151
Publikationen und Vorträge	167
Thesen	171

Abbildungsverzeichnis

2.1. Systemkomponenten in Smart Meeting Rooms	25
2.2. Foto vom Smart Lab in Rostock	26
2.3. Schematische Darstellung vom Smart Lab in Rostock	27
2.4. Entscheidungsbaum für Adaptionsmechanismen von Thiede	30
2.5. Layout-Anpassungen von Radloff	31
2.6. Literaturübersicht für Display Ecologies von Chung et al.	37
4.1. Entworfenes Nutzer-Interface für die persönlichen Geräte	57
4.2. Illustration der View-Verteilung und Layout-Berechnung	64
4.3. Workflow für die Integration spontan generierter Views	72
4.4. Verknüpfen von Tools zur Generierung von Views	76
4.5. Illustration der Integration von spontan generierten Views	78
4.6. Generierung von Views durch das Lupen-Tool	80
4.7. Beispiele für Views mit DVR im FeVis-Tool	83
4.8. Integration von Views des FeVis-Tools mit DVR in den SMR	84
4.9. Beispiel für eine View mit Feature-Darstellung im 3D-Raum	85
4.10. Integration von Views mit Feature-Visualisierung	85
4.11. Beispiel für View mit 3D-Features aus mehreren Zeitpunkten	86
4.12. Integration von Views mit zusätzlichen Features in den SMR	87
4.13. Beispiel für View mit zeitl. Feature-Übersicht im FeVis-Tool	88
4.14. Beispiel-View mit zusätzl. Feature-Übersicht im 3D-Raum	88
4.15. Integration von Views mit Übersichtsdarstellungen in den SMR	89
4.16. Foto einer Feature-basierten Analyse im SMR von Rostock	90
4.17. Beispiele für Web-basierte Inhalte	92
4.18. Integration Web-basierter Inhalte in den SMR	93
5.1. Visualisierung des Interaktionsgraphen	99
5.2. Anwendung von Filtern auf dem Interaktionsgraph	102
5.3. Illustration einer globalen Undo/Redo-Operation	104
5.4. Illustration einer selektiven Redo-Operation	105
5.5. Basisvisualisierung des Semantischen Graphen	109
5.6. Semantic Substrate-Darstellung des Semantischen Graphen	110
5.7. Scatterplot-Darstellung des Semantischen Graphen	112
5.8. Visualisierung semantischer Informationen im Interface	114
5.9. Empfehlung semantisch passender Views auf den Displays	116

Abbildungsverzeichnis

5.10. Einblenden der semantischen Nachbarschaft auf den Displays	117
5.11. Balkendiagramm von den benötigten Zeiten für die Navigation (Studie) .	120
5.12. Balkendiagramm von den benötigten Zeiten für die Anpassung (Studie) .	121
6.1. Situationsbedingte Konfiguration des Nutzerinterfaces	131
6.2. Situationsbedingte Layout-Anpassung: Ausgangslage	132
6.3. Nutzer-bezogene Layout-Anpassung am Beispiel	133
6.4. Display-bezogene Layout-Anpassung am Beispiel	134
6.5. Nutzer- und Display-bezogene Layout-Anpassung	135

Tabellenverzeichnis

4.1. Zeiten für die View-Verteilung und Layout-Berechnung	70
---	----

1. Einleitung

1.1. Motivation und Zielstellung

Smarte Umgebungen basieren auf der Idee, dass mehrere in der Umgebung verteilte und miteinander vernetzte Geräte dazu verwendet werden, um Nutzer gemeinsam beim Erfüllen von alltäglichen Arbeiten zu unterstützen [EK05; Aar06; MuS15]. Smarte Umgebungen sind häufig auch Multi-Display Umgebungen. Das bedeutet, dass das Geräteensemble über mehrere visuelle Ausgabegeräte wie z.B. Beamer, Computermonitore oder große Fernsehgeräte verfügt, die zur Anzeige von grafischen Repräsentationen genutzt werden können. Darüber hinaus ermöglichen es installierte Sensoren, wie z.B. Kameras, Mikrofone, oder Fußbodensensoren, das Geschehen in smarten Umgebungen zu erfassen. Der Geräteverbund einer Smarten Umgebung ist typischerweise nicht statisch, sondern erlaubt in der Regel auch das spontane Integrieren von persönlichen Geräten wie z.B. eigenen Laptops oder Tablets.

Ziel von Smarten Umgebungen ist es, das Zusammenwirken der verteilten Komponenten so zu gestalten, dass den Nutzern beim Erfüllen ihrer Aufgaben durch automatische Konfiguration und Berechnungen assistiert wird. Hierzu wird von Smarten Systemen das Geschehen in der Umgebung erfasst und automatisch ausgewertet [YN10]. Basierend auf den gewonnenen Erkenntnissen wird anschließend die Umgebung selbstständig so konfiguriert, dass den Nutzern Arbeit abgenommen wird, oder der Aufwand zum Erreichen von Zielen reduziert wird.

Aktuelle Forschung konzentriert sich darauf Modelle und Algorithmen zu entwickeln, um das Geräteensemble in unterschiedlichen Verwendungsszenarien besser nutzen zu können und die hierfür notwendige Assistenz bereitzustellen.

Ein Beispiel für smarte Umgebungen sind Smart Meeting Rooms (SMR). Diese stellen das Referenzszenario für das Graduiertenkolleg MuSAMA (Multimodal Smart Appliance Ensembles for Mobile Applications) dar. Das GRK MuSAMA konzentriert sich darauf, Wissensexploration und Wissensintegration in Smart Meeting Rooms zu stärken und somit den Informationsaustausch innerhalb von Teams zu verbessern [MuS15]. Die vorliegende Dissertation entstand im Rahmen von MuSAMA und adressiert den Teilbereich der visuellen Informationsdarstellung unter Nutzung der vorhandenen Multi-Display-Umgebung.

Der Informationsaustausch in Smart Meeting Rooms kann verschiedene Formen annehmen, wie z.B. die einer Präsentation in Vorlesungs- oder Unterrichtssituationen oder die

1. Einleitung

einer freien Diskussion mit mehreren aktiven, sich spontan abwechselnden Beteiligten. Ein wichtiges Mittel für den Informationsaustausch ist die Verwendung von visuellen Repräsentationen: Mithilfe von Stichpunktfolien, Diagrammen, Übersichtsgrafiken und verschiedenen Datenvisualisierungen lassen sich Konzepte, Hypothesen und Argumente anschaulich vermitteln.

Prinzipiell haben Smart Meeting Rooms somit das Potential, den Informationsaustausch durch visuelle Inhalte zu unterstützen:

- Als Multi-Display-Umgebungen ermöglichen sie es, mehrere Inhalte gleichzeitig auf verschiedenen Anzeigegeräten anzuzeigen und für die Betrachtung durch mehrere Nutzer gut sichtbar darzustellen.
- Die bestehende Gerätevernetzung ermöglicht einen Austausch von Inhalten zwischen unterschiedlichen Geräten und die Integration von persönlichen Geräten eröffnet die Option, Inhalte, die von verschiedenen Nutzern bereit gestellten werden, miteinander zu kombinieren.
- Die installierten Sensoren erlauben es zudem, die aktuelle Nutzungssituation des Raums zu erkennen und ggf. auf Veränderungen mit einer unterstützenden Nutzerassistenz zu reagieren.

Diese Eigenschaften des Smart Meeting Rooms bieten eine ideale Basis, um die Informationspräsentation bei wechselnden Inhalten, die von unterschiedlichen Nutzern und Geräten bereitgestellt werden, zu unterstützen. Allerdings fokussieren heutige Arbeiten zu Assistenzsystemen in SMR hauptsächlich technische Aspekte, welche bspw. die Verwaltung eines dynamischen, heterogenen Geräteensembles oder die Erkennung der aktuellen Situation aufgrund der ausgewerteten Sensordaten betreffen [FBR15; YN10]. Dagegen wurde die Entwicklung von Assistenzsystemen zum Sammeln, Verteilen und Anzeigen von visuellen Informationen bisher nur unzureichend in den aktuellen Forschungsarbeiten zu SMRs thematisiert.

Andererseits nutzen bisherige Ansätze, welche die Informationsdarstellungen in Multi-Display Umgebungen unterstützen, noch nicht das Potential von Smart Meeting Rooms aus (vgl. [Chu+15]). Die vorliegende Dissertation ordnet sich in dieses Spannungsfeld ein.

Ziel der vorliegenden Dissertation ist es deshalb Ansätze zu entwickeln, die eine assistierte Informationspräsentation in Smart Meeting Rooms ermöglichen. Assistenz heißt hierbei, dass den Nutzern manueller Konfigurationsaufwand abgenommen wird, wenn es darum geht, Inhalte für Präsentationen bereit zu stellen und auf Inhalte von verschiedenen persönlichen Geräten zuzugreifen. Nutzern soll geholfen werden, mehrere dieser Inhalte gemeinsam auf den Displays einer Multi-Display Umgebung zur Unterstützung von Vorträgen anzuordnen, passende Inhalte für die Ergänzung der Inhaltsanzeige zu suchen und schnell zwischen verschiedenen Inhaltszusammenstellungen zu wechseln. Hierdurch soll es mehreren Nutzern möglich sein, gemeinsam dynamische Inhaltszusammenstellungen zu generieren und in einer Multi-Display-Umgebung anzuzeigen.

Anwendungsszenarien hierfür beinhalten das gemeinsame Vorbereiten eines Vortrags, das Abhalten eines Projekt-Meetings oder die Expertendiskussion über ein Fachthema. In jedem dieser Fälle soll es möglich sein, dass verschiedene Teilnehmer von ihren persönlichen Geräten aus Inhalte bereitstellen, die anschließend gemeinsam verwaltet und angezeigt werden. Dabei soll auch die spontane Visualisierung von Daten innerhalb einer Präsentation unterstützt werden. Außerdem sollen sich auch im Nachhinein Vortrags- und Diskussionsverläufe abrufen, modifizieren und ggf. mit weiteren Inhalten ergänzen lassen.

Der Schwerpunkt dieser Arbeit liegt auf der Entwicklung von Ansätzen, die solche Szenarien unterstützen, d.h. auf der Entwicklung von Methoden, die eine assistierte Präsentation und Diskussion in SMR gewährleisten.

1.2. Problembeschreibung

Ausgangspunkt ist ein Szenario in dem mehrere Nutzer ihre persönlichen Geräte dazu verwenden, um Inhalte aus verschiedenen Quellen zu kombinieren und um Inhaltszusammenstellungen für die Anzeige in einer Multi-Display-Umgebung zu generieren. Für eine assistierte Informationspräsentation in diesem Nutzungsszenario ist es erforderlich[YN10; Chu+15], dass

- das Auffinden von Inhalten unterstützt,
- das Zusammenstellen der Inhalte ermöglicht,
- die Verwaltung der Inhalte vorgenommen,
- die Anzeige der Inhalte auf den Displays unterstützt und
- bei der Protokollierung der Meetings geholfen wird.

Allerdings adressieren heutige Systeme nur eine gewisse Teilmenge dieser Forderungen indem sie einerseits zwar Lösungen für Teilprobleme bieten, die Übertragbarkeit dieser Ansätze andererseits aber durch verschiedene Annahmen einschränken, wie z.B. die Ausrichtung auf einen speziellen Anwendungsfall, einen festen Datensatz, festgelegte Analyseziele oder eine bestimmte Gerätekonfiguration [Chu+15]. Ziel der vorliegenden Dissertation ist es deshalb neue Ansätze zu entwickeln, welche eine gleichzeitige und allgemeine Unterstützung aller dieser Fragestellungen ermöglicht. Hierzu müssen folgende Teilprobleme gelöst werden:

1. Modelle müssen entwickelt werden, um die Suche und Verwaltung von Inhalten sowie die Protokollierung von Meetings zu ermöglichen.
2. Nutzerschnittstellen (UI) müssen bereitgestellt werden, um einen Zugriff auf die Modelle zu ermöglichen. Beim Zusammenstellen der Inhalte müssen hierbei nicht

1. Einleitung

nur vordefinierte, sondern auch während einer Besprechung spontan bzw. bedarfsgerecht generierte Informationsrepräsentationen eingebunden werden können, um so z.B. auch auf sich dynamisch ändernde Schwerpunkte im Diskussionsverlauf reagieren zu können.

3. Assistenz muss entwickelt werden, um beim Auffinden, Zusammenstellen und Anzeigen von Inhalten zu unterstützen und hierfür die SMR-eigenen Intentions- und Situationserkennung zu verwenden.

1.3. Ergebnisse

Es wurden neuartige Konzepte erarbeitet, die das dynamische Erzeugen, Anzeigen und Anpassen von Informationsdarstellungen für eine Multi-Display-Umgebungen durch mehrere Nutzer ermöglichen und durch automatische Assistenz unterstützen. Diese werden im Folgenden zusammengefasst.

1.3.1. Modellentwicklung

Als Grundlage für die entwickelten Konzepte wurde eine umfangreiche Modellbildung vorgenommen. Dazu wurden 3 Modelle entworfen:

- Der Präsentationsgraph für eine räumliche und zeitliche Strukturierung von anzuzeigenden Darstellungen.
- Der semantische Graph, um die themenbezogenen Verbindungen zwischen den Darstellungen zu beschreiben.
- Der Interaktionsgraph, um die erfolgten Interaktionen der Nutzer mit dem System zu beschreiben.

Das Modell für die räumliche und zeitliche Strukturierung der Inhalte dient dazu, das Zusammenstellen der anzuzeigenden Informationen und ihren Austausch über den Präsentationsverlauf hinweg zu beschreiben. Das als “Präsentationsgraph” bezeichnete Modell definiert Vorgaben für die Verteilung und Anordnung von mehreren Darstellungen auf multiplen Displays. Diese Vorgaben werden von Assistenzsystemen des Smart Meeting Rooms berücksichtigt, um die Anzeige automatisch an das flexible Geräteensemble anzupassen. Gleichzeitig ermöglicht die entwickelte Beschreibung aber auch spontane Anpassungen durch verschiedenen Nutzer während einer Besprechung. Diese werden somit in die Lage versetzt, die assistierte Inhaltsdarstellung zu steuern.

Daneben, wird ein Modell zur semantischen Strukturierung der Inhalte verwendet. Dieser “semantische Graph” erlaubt das schnelle Auffinden und inhaltsbasierte Verknüpfen von Darstellungen. Er wird genutzt, um durch eine Nutzerassistenz passende Inhalte für

die Anzeige zu empfehlen, einen nach Themen geordneten Überblick über die gesamte Inhaltsmenge zu geben und Verknüpfungen zwischen Inhalten zu visualisieren.

Um Besprechungen zu protokollieren, wird als drittes Modell der “Interaktionsgraph” eingeführt. Dieser wird automatisch generiert indem während einer Besprechung Nutzerinteraktionen zur Anpassung der Informationsanzeige aufgezeichnet werden. Er dient als Grundlage, um den Verlauf eines Treffens aufzuzeichnen und erfolgte Anpassungen einzelner Nutzer nachvollziehbar zu machen.

Die Bereitstellung dieser Modelle ist ein erstes wichtiges Ergebnis der vorliegenden Dissertation. Die Modelle wurden in 2 Publikationen beschrieben [Eic+15; ENS15]. Sie sind die Grundlage, um das Zusammenstellen und Anzeigen einer assistierten Präsentation in dem Geräte-Ensemble eines Smart Meeting Rooms zu ermöglichen.

1.3.2. Nutzerinterface

Für die Steuerung der Informationsdarstellung wurde ein Nutzerinterface (UI) entwickelt, das auf allen persönlichen Geräten der Nutzer aufgerufen werden kann und jedem Nutzer Zugriff auf die gemeinsam generierte Präsentation bzw. Diskussion gibt.

Hierzu stellt das Nutzerinterface folgende Funktionen bereit:

1. Es erlaubt Nutzern eigene Inhalte für eine Präsentation bereitzustellen. Dies ist die Grundlage dafür, dass eine Informationsdarstellung generiert werden kann, die Inhalte von verschiedenen Nutzern enthält.
2. Das UI bietet den Nutzern einen Zugang zu den eingeführten Modellbeschreibungen, welche die bereitgestellten Inhalte strukturieren (vgl. Abschnitt 1.3.1). Hierfür werden verschiedene Visualisierungen der Modelle verwendet:
 - Zur Koordinierung einer Präsentation dient eine Visualisierung des Präsentationsgraphen. Diese Visualisierung erlaubt es Benutzern, sich einen Überblick über den zeitlichen Ablauf einer Präsentation und den dabei angezeigten Inhalten zu verschaffen. Nutzer können mit dieser Visualisierung interagieren, um die Verwendung von Inhalten zu steuern, dabei die Anzeige von Informationsdarstellungen zu priorisieren, Darstellungen in räumliche Beziehung zueinander zu setzen und die zeitliche Abfolge von Darstellungen für eine Präsentation zu planen.
 - Eine themenbasierte Suche nach verwandten Inhalten wird durch Visualisierungen des semantischen Graphen unterstützt. Einerseits werden hierbei einzelne semantische Verknüpfungen zwischen ausgewählten Darstellungen durch Visual Links kommuniziert, um das Ergänzen einer Informationsdarstellung durch passende Inhalte zu unterstützen. Andererseits, erlauben alternative Darstellungen des semantischen Graphen eine Exploration der gesamten verfügbaren Inhaltsmenge mit Bezug auf verschiedene Themengebiete.

1. Einleitung

te oder Attribute. Hierdurch können sich Nutzer einen Überblick von den durch verschiedene Nutzer bereitgestellten Inhalten und von den Zusammenhängen zwischen ihnen verschaffen.

- Ein Abrufen früherer Zustände einer Präsentation wird durch eine Visualisierung des Interaktionsgraphen unterstützt. Basierend auf dieser Ansicht lassen sich einzelne Nutzeraktionen rückgängig machen oder komplette frühere Zustände wiederherstellen.
3. Während einer Besprechung können spontan erzeugte Darstellungen in eine Präsentation aufgenommen werden. Hierzu bietet das UI die Möglichkeit, bilderzeugende Programme mit der Präsentation zu verknüpfen. Auf diese Weise kann je nach Bedarf eine Informationsanzeige um alternative Datenvisualisierungen, Detailansichten oder zusätzliche Ansichten ergänzt werden, um auf Fragen, wechselnde Diskussionspunkte oder Analyseziele eingehen zu können.

Die Entwicklung dieser UIs ist ein weiteres wichtiges Ergebnis der vorliegenden Dissertation. Es stellt einen Funktionsumfang bereit, der von heutigen Systemen noch nicht angeboten werden kann. Dazu gehören insbesondere die unter 2. und 3. beschriebenen Ansätze:

- Die Visualisierung des Präsentationsgraphen erlaubt es, mehreren Nutzern gemeinsam und spontan, die gleichzeitige Verwendung von mehreren Darstellungen in einer Multi-Display-Umgebung zu beschreiben. Die Visualisierung fokussiert sich hierbei darauf, die Nutzervorgaben für die räumliche und zeitliche Verwendung von Darstellungen zu verdeutlichen, ohne dabei gerätspezifische Vorgaben (wie z.B. welche Darstellung auf welchem Display angezeigt werden soll) zu verwenden. Hierdurch kann die Informationsanzeige durch Nutzervorgaben gestaltet werden, ohne dass Nutzer dabei die konkret vorhandene Display-Infrastruktur (oder deren Rekonfiguration durch ein sich ad-hoc anpassendes Geräteensemble) überblicken müssen. Außerdem können auf diese Weise einmalig geplante Präsentationen wiederverwendet und in unterschiedlichen Multi-Display-Umgebungen (z.B. mit anderer Display-Anzahl und -Größe) angezeigt werden. Der Präsentationsgraph, seine Visualisierung und die Interaktionsmöglichkeiten mit der Visualisierung wurden in einem Journal-Bericht publiziert [Eic+15].
- Die Visualisierung des Interaktionsgraphen und die hierauf aufbauende Undo/Redo-Mechanismus erlauben ein schnelles Umkonfigurieren der Inhaltsanzeige. Die Besonderheit hierbei ist, dass Aktionsfolgen (welche die Informationsdarstellung beeinflusst haben) differenziert nach Nutzern, Displays und Inhalten angezeigt und bearbeitet werden können. Auf diese Weise können Anpassungen der Informationsdarstellung schnell identifiziert und rückgängig gemacht bzw. wiederhergestellt werden. Außerdem wird es hierdurch auch möglich, schnell zwischen komplett unterschiedlichen Präsentationszuständen wechseln zu können. Dieser Ansatz wurde in Zusammenarbeit mit dem Potsdamer Institut für Klima-

folgenforschung (PIK) im Anwendungshintergrund einer Podiumsdiskussion über Klimadaten publiziert [Eic+15].

- Die spontane Erzeugung und Integration von neuen Darstellungen in eine Präsentation erlaubt es, einen geplanten Präsentationsverlauf durch verschiedenste neue und ungeplante Inhalte zu ergänzen. Für das Einbinden von alternativen Datenvisualisierungen wurde der Ansatz am Beispiel einer Feature-basierten Analyse von Simulationsdaten biochemischer Reaktionsnetzwerke umgesetzt und veröffentlicht [Eic+13; Eic+14]. Zudem wurde ein Lupen-Tool für das Universitätsprojekt “Studium Optimum” entwickelt. Das Tool erlaubt es Nutzern, interaktiv Regionen von existierenden Darstellungen auszuwählen und vergrößert als neue, separate Darstellungen in die Präsentation einzubinden.

Als weiteres Anwendungsbeispiel für spontan erzeugte Inhalte wurde eine Umsetzung für den Webbrowser entwickelt [Eic+15]. Hierdurch können interaktive Webseiteninhalte als neue Inhalte in die Präsentation aufgenommen werden. Auf diese Weise können insbesondere auch online-verfügbare Visualisierungstools verwendet werden.

1.3.3. Assistenz

Über das UI können Nutzer spezifizieren, welche Darstellungen, wann im Präsentationsverlauf erscheinen sollen, wie die Anzeige der Darstellungen hierbei priorisiert werden soll, welche Zusammenhänge zwischen den Inhalten bestehen und welche Repräsentationen während der Präsentation generiert werden sollen. Die Informationsanzeige auf den unterschiedlichen Displays soll sich jedoch automatisch an die Geräteumgebung und die Situation in der Umgebung anpassen. Hierfür wurden Ansätze entwickelt, um folgende Funktionen durch Assistenz bereitzustellen:

1. Die Verteilung und räumliche Anordnung der Darstellungen auf den verfügbaren Displays des Raums wird (unter Berücksichtigung der durch die Nutzer spezifizierten Vorgaben) automatisch vorgenommen.
2. Die Suche nach relevanten/interessanten Inhalten wird durch Empfehlungen von passenden Inhalten unterstützt.
3. Die Informationsanzeige wird durch situationsabhängiges Priorisieren von Darstellungen angepasst.

Für die Anzeige der Inhalte wurde ein Layout-Algorithmus entwickelt, der die Verteilung und räumliche Anordnung der visuellen Inhalte auf den verfügbaren Displays automatisch übernimmt [Eic+15]. Hierbei werden Geräteeigenschaften (z.B. Displaygröße), die Situation im Raum (z.B. Sichtbarkeit der Displays basierend auf den Positionen der Nutzer) und insbesondere die Nutzervorgaben auf Basis des Präsentationsgraphen berücksichtigt.

1. Einleitung

Für die Unterstützung der Suche nach passenden Inhalten werden durch die Nutzer bereitgestellte Inhalte automatisch ausgewertet, um mögliche Darstellungen (im Folgenden auch “Views” genannt) und semantische Beziehungen zwischen den Darstellungen zu extrahieren. Diese Beziehungen werden in den semantischen Graph eingebunden und dienen dazu, zu einer aktuellen Informationsanzeige passende Darstellungen zu ermitteln. Diese werden anschließend als Navigationsempfehlungen auf den Displays des Raums eingeblendet, um Nutzern ein spontanes Abrufen der zusätzlichen Informationen zu ermöglichen.

Um die Informationsdarstellung automatisch an die Situation im Raum anzupassen, wird die Priorisierung der Inhalte automatisch geändert. Hierfür werden z.B. beim Wechsel des Redners die zum erkannten Nutzer gehörenden Darstellungen stärker betont als die Inhalte von anderen Nutzern.

Als Grundlage zur Anwendung dieser Techniken dient die durch die Situationserkennung eines Smart Meeting Rooms ermittelte Situation und Intentionen der Nutzer. Die grundlegenden Ideen hierfür wurden durch einen Konferenzbeitrag veröffentlicht [ENS15]. Diese Publikation thematisiert auch die praktische Integration der persönlichen Geräte in das bestehende Geräteensemble eines Smart Meeting Rooms.

Die Entwicklung dieser Assistenzsysteme ist ein weiteres wichtiges Ergebnis dieser Arbeit. Hiermit werden die Möglichkeiten, die Smart Meeting Rooms durch ihre besondere Geräte- und Software-Infrastruktur bereitstellen, für die Informationsanzeige in einer Multi-Display-Umgebung genutzt. Diese Kombination wird von bisherigen Ansätzen noch nicht unterstützt.

1.3.4. Publikationen

Die Kombination der entwickelten Modelle, die Steuerung der Umgebung über das Nutzerinterface durch mehrere Nutzer und die hierfür entworfene Assistenz leisten einen wichtigen Beitrag, um die Informationsanzeige für Multi-Display-Umgebungen weiterzuentwickeln.

Die Ergebnisse dieser Dissertation wurden in 5 Artikeln veröffentlicht; 3 Journal-Publikationen und 2 Beiträge auf internationalen Konferenzen.

[Eic+13] Christian Eichner, Arne Bittig, Heidrun Schumann and Christian Tomin-ski. Feature-Based Visual Analytics for Studying Simulations of Dynamic Bi-Stable Spatial Systems. In *Proceedings of the EuroVis Workshop on Visual Analytics (EuroVA)*. Eurographics Association, 2013.

[Eic+14] Christian Eichner, Arne Bittig, Heidrun Schumann and Christian Tomin-ski. Analyzing Simulations of Biochemical Systems with Feature-Based Visual Analytics. *Computers & Graphics*, 38(0) : 18 - 26, 2014.

- [Eic+15] Christian Eichner, Thomas Nocke, Hans-Jörg Schulz and Heidrun Schumann. Interactive Presentation of Geo-Spatial Climate Data in Multi-Display Environments. *ISPRS International Journal of Geo-Information*, 4(2) : 493, 2015.
- [ENS15] Christian Eichner, Martin Nyolt and Heidrun Schumann. A Novel Infrastructure for Supporting Display Ecologies. In *Proceedings of the 11th International Symposium on Visual Computing (ISVC)*, Las Vegas, NV, USA, December 2015.
- [Eic+16] Christian Eichner, Stefan Gladisch, Heidrun Schumann and Christian Tominiski. Direct Visual Editing of Node Attributes in Graphs. *Informatics*, 3(4) : 17, 2016.

1.4. Aufbau der Arbeit

Die Arbeit ist in 8 Kapitel unterteilt. Nach der Einführung und Motivation in Kapitel 1, werden in Kapitel 2 die Grundlagen und der Stand der Forschung zusammengefasst. Basierend auf diesen Betrachtungen werden in den folgenden Kapiteln die entwickelten Lösungsansätze vorgestellt.

Kapitel 3 beschreibt die grundlegende Modellbildung zur Organisation der visuellen Inhalte und die Protokollierung der Nutzeraktionen. Die notwendige zeitliche und räumliche Strukturierung wird als Präsentationsgraph (Abschnitt 3.1), die themenbezogene Strukturierung als semantischer Graph (Abschnitt 3.2) und die Interaktionsfolgen als Interaktionsgraph beschrieben (Abschnitt 3.3).

Kapitel 4 - 6 zeigen dann, wie auf Basis dieser Modelle eine assistierte Präsentation und Diskussion erreicht werden kann.

Kapitel 4 stellt die Konzepte für die assistierte Präsentation vor. Hierbei werden das Nutzer-Interface (Abschnitt 4.1), der zur automatischen Verteilung und Anzeige von Darstellungen verwendete Layout-Algorithmus (Abschnitt 4.2) und die Ansätze zur bedarfsgerechten Ergänzung der Anzeige durch spontan generierte Visualisierungen (Abschnitt 4.3) vorgestellt.

Kapitel 5 befasst sich mit der assistierte Diskussion. Hierfür werden protokollierte Interaktionsfolgen visualisiert und für einen entwickelten Undo/Redo-Mechanismus genutzt (Abschnitt 5.1). Die Visualisierung der semantischen Eigenschaften und Beziehungen der Inhalte, sowie deren Nutzung bei der Suche nach Inhalten wird in Abschnitt 5.2 thematisiert.

Kapitel 6 beschreibt die automatische Konfiguration, die zur Reaktion auf Situationen und Szenarien entwickelt wurde, welche durch die Situations- und Intentionserkennung in einem SMRs erkannt wurden.

1. Einleitung

In Kapitel 7 werden abschließend die Ergebnisse der Dissertation noch einmal kurz zusammengefasst und ein Ausblick auf zukünftige Arbeiten gegeben.

2. Grundlagen und Stand der Forschung

In diesem Kapitel wird der aktuelle Stand der Forschung beschrieben. Hierzu wird zunächst in Abschnitt 2.1 auf Smart Environments und im Speziellen auf Smart Meeting Rooms (SMR) eingegangen, welche die Forschungsumgebung von MuSAMA bilden. Anschließend wird in Abschnitt 2.2 die Informationsanzeige in Multi-Display-Umgebungen behandelt.

2.1. Smart Environments

Smart Environments liegt die Idee zugrunde, dass Personen in Zukunft zunehmend von Informations- und Kommunikationstechnologie umgeben sind [Wei99] und diese täglich zur Erfüllung ihrer Aufgaben nutzen [Aar06; ANA10]. Unterschiedliche Arten von Geräten sollen sich hierfür möglichst unauffällig in die alltägliche Umgebung integrieren [Nor98]. Dies macht es für den Nutzer jedoch schwerer, die Vielzahl der unterschiedlichen Computer als solche zu erkennen, ihren Funktionsumfang zu erfassen und entsprechend zu nutzen. Aus diesem Grund ist es ein wichtiges Forschungsanliegen, die Bedienung dieser Umgebungen intuitiv zu gestalten [PST04]. In intelligenten Umgebungen [EK05] sollen durch Sensornetzwerke und Gerätevernetzung Informationen über den Zustand der Umgebung, die Nutzer und deren Intentionen gesammelt werden [BDR07]. Mithilfe der gewonnenen Informationen soll sich dann das Geräteensemble automatisch so konfigurieren, dass die Nutzer beim Erfüllen ihrer Aufgaben unterstützt werden und somit ihre Ziele einfacher erreichen können [CD07].

Je nach Anwendungshintergrund gibt es verschiedene Arten von Smart Environments wie z.B.: Smart Homes [Cha+08; Doh+10; RM13; Moz05; EK05; HK05], Smart Meeting Rooms [Wai+03; Aar06; JFW02; HK05], Smart Office oder Smart Class Rooms [EK05; FFH00]. Im Folgenden sollen Smart Meeting Rooms genauer betrachtet werden, da sie das Referenzszenario für das Graduiertenkolleg MuSAMA und damit auch für die vorliegende Dissertation darstellen.

2.1.1. Smart Meeting Rooms

Ziel von MuSAMA ist es, durch Assistenz das Arbeiten in Smarten Umgebungen, speziell in Smart Meeting Rooms, mit vielen verteilten Geräten zu unterstützen [MuS15].

2. Grundlagen und Stand der Forschung

Smart Meeting Rooms werden speziell dazu entworfen, das Abhalten von Besprechungen mit mehreren Beteiligten zu unterstützen. Diese Besprechungen werden in der Regel in Form von Präsentationen (z.B. Vorträgen) oder wissenschaftlichen Diskussionen abgehalten [EK05; Aeh+13; Bur+08]. Um dabei den Informationsaustausch zwischen den Nutzern zu fördern, werden visuelle Repräsentationen aus unterschiedlichen Quellen ausgetauscht und angezeigt.

Als Quelle zur Bereitstellung von visuellen Repräsentationen sollen die verschiedenen persönlichen Geräte der Nutzer dienen. Für solche *Bring your own device*-Szenarien ist es erforderlich, dass die persönlichen Geräte mit der Geräteinfrastruktur des Raums verbunden werden [Lei11]. Um den Konfigurationsaufwand hierbei so gering wie möglich zu halten, sind Smart Meeting Rooms deshalb typischerweise als ad-hoc Umgebungen ausgelegt [EK05]. Das bedeutet, dass das Geräteensemble nicht statisch ist, sondern durch neue Geräte spontan erweitert werden kann [Thi10]. Neben dem Bereitstellen von Inhalten, sollen die persönlichen Geräte auch dazu verwendet werden, um die installierten Geräte und insbesondere auch die Anzeige der Inhalte im Raum steuern zu können [SME15; Wig+09]. Für diese Anzeige von visuellen Inhalten verfügen Smart Meeting Rooms üblicherweise über mehrere unterschiedliche Displays oder Projektoren [RLS11; Wig+09; Chu+15; Wal+09], was sie auch zu *Multi-Display-Umgebungen* macht.

Zur Unterstützung dieser Aufgaben werden in Smart Meeting Rooms spezielle Software-Systeme eingesetzt, welche ein auf Präsentations- und Diskussionsszenarien abgestimmtes Assistenzverhalten anbieten [Krü11]. Beispiele für Assistenzverhalten sind etwa das automatische Steuern der Beleuchtungssituation, um Darstellungen auf Leinwänden besser sichtbar zu machen [BNK13; Dyr+11], oder das Einschalten von Projektoren (und Herunterfahren von dazugehörigen Leinwänden), um Anzeigefläche für einen dazugekommenen Nutzer zu schaffen [Dyr+11; BNK13]. Bezüglich der Aufgaben, die in dieser Dissertation thematisiert werden, wurden im Rahmen von MuSAMA bereits Ansätze entwickelt, die Assistenz beim Austauschen und Anzeigen der visuellen Darstellungen [Rad14; Thi10; Aeh+13] anbieten.

2.1.2. Software-Komponenten für Smart Meeting Rooms

Systeme für Smart Meeting Rooms bestehen aus verschiedenen, miteinander verknüpften Software-Komponenten, die jeweils unterschiedliche Teilaufgaben übernehmen und Dienste (z.B. für andere Komponenten) bereitstellen. Die Einordnung der Komponenten erfolgt in drei Bereiche (vgl. Abbildung 2.1) [YN10; JM05; CD07; Leh+10]. Auf der untersten Ebene, von Yu “Meeting Capture” genannt [YN10], steht die Aufnahme von Sensordaten, wie z.B. Video- oder Audiodaten durch Mikrophone oder Kameras, sowie das Aufzeichnen raumspezifischer Daten wie z.B. das Ermitteln von Bewegungsdaten durch Drucksensoren im Fußboden. Basierend auf diesen durch physikalische Messungen gewonnen Sensordaten, wird in einem Erkennungsschritt (“Meeting Recognition”) eine

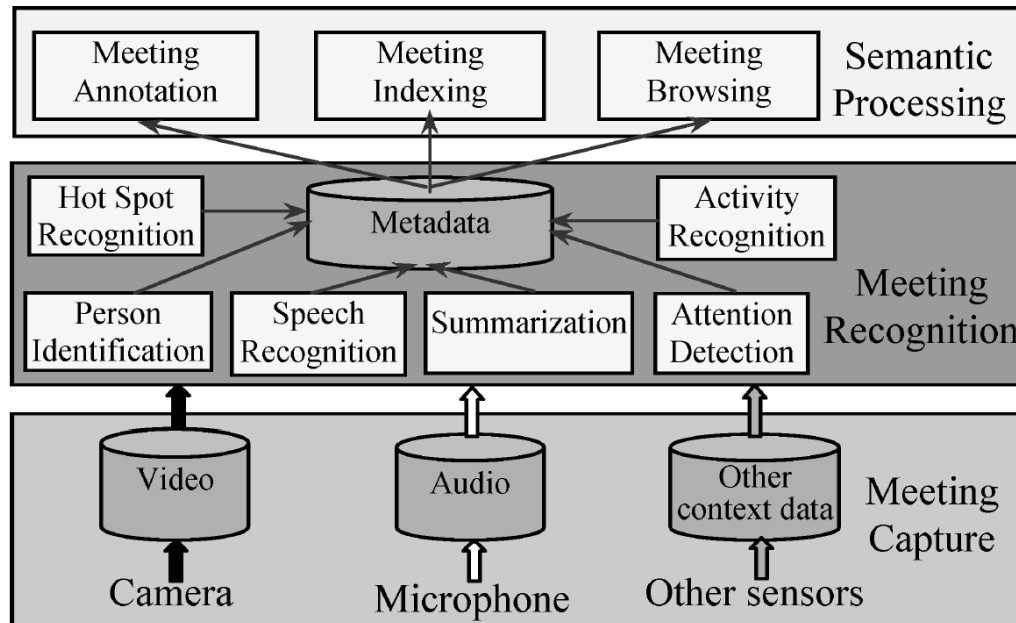


Abbildung 2.1.: Überblick und Einordnung der Komponenten eines Systems für Smart Meeting Rooms nach Yu (Bild aus [YN10]).

Abstraktion vorgenommen: Durch Dienste zur Aktivitätserkennung, Spracherkennung, Identifikation von Nutzern usw. werden Metadaten gewonnen, die z.B. die Position und Blickrichtung eines Nutzers, dessen Identität und Aktivität beschreiben. Die beiden unteren Ebenen übernehmen somit das Aufnehmen, Sammeln, Aggregieren und Übertragen von Daten, sowie das Ansteuern von erkannten Geräten (z.B. von Displays des SMR oder von persönlichen Geräten der Nutzer). Die obere Schicht der Architektur (“Semantic Processing”) steuert wie diese Daten verwendet werden, auf welche Weise die Geräte konfiguriert werden, um die Nutzer zu unterstützen, und wie die Nutzer selbst mit dem System interagieren können.

Die im Abschnitt 1.2 beschriebenen Herausforderungen adressieren vor allem die Suche und Verwaltung von Inhalten, die Berechnung der Anzeige sowie die Protokollierung von Meetings und sind somit in die Teilbereiche “Meeting Browsing” und “Meeting Indexing” des “Semantic Processing” einzuordnen (vgl. Abbildung 2.1).

Die Software-Komponenten eines SMR können auf unterschiedliche physikalische Geräte verteilt sein. Weil Smart Meeting Rooms ad-hoc Umgebungen sind, müssen auch die Software-Komponenten lose miteinander gekoppelt sein und der Datenaustausch muss über Netzwerkverbindungen erfolgen. Für die Verwaltung der Software-Infrastruktur und die Datenübertragung zwischen den Komponenten wird eine *Middleware* verwendet [You05; YN10; BK12]. Die Middleware ist eine zusätzliche Software-Ebene zwischen der Hardware und den (Software-)Anwendungen. Sie kapselt die Geräte-Hardware und

2. Grundlagen und Stand der Forschung

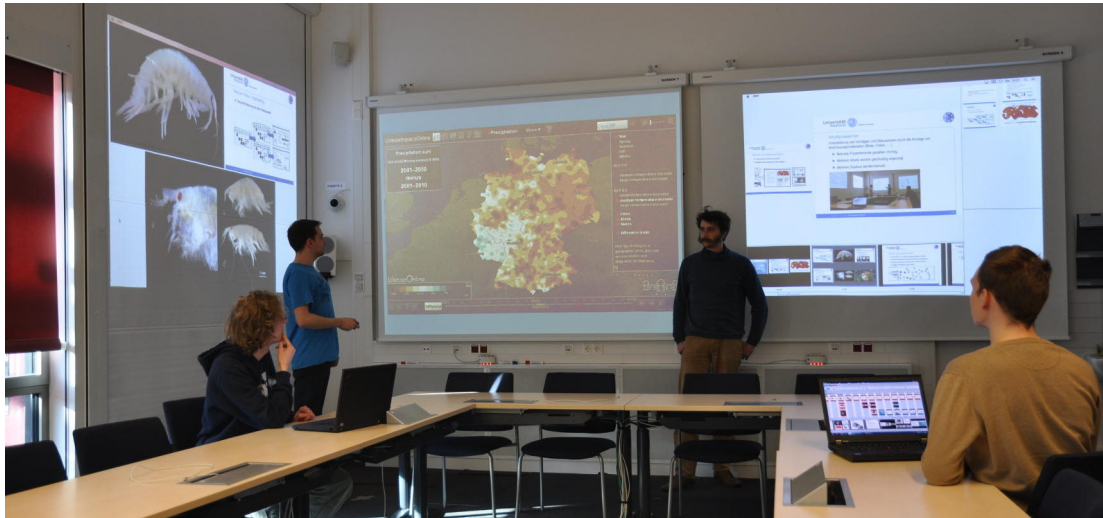


Abbildung 2.2.: Bild einer Präsentation im Smart Lab von Rostock. Zu sehen sind drei der sieben Leinwände die zur Anzeige von visuellen Inhalten genutzt werden.

Dienste des Raums in eigenständige Software-Komponenten und implementiert Protokolle, um Dienste zu suchen, Daten über verschiedene Netzwerkprotokolle zwischen den Software-Komponenten zu übertragen oder um Komponenten zur Laufzeit miteinander zu verbinden [BK12]. Aus Sicht der Anwendungen abstrahiert eine Middleware von der zugrunde liegenden Hardware. Auf diese Weise erlaubt es eine Middleware, schnell neue Software-Komponenten zu entwickeln [BRK10] und in eine bereits bestehende Infrastruktur einzubetten, um neue Funktionalitäten in einen Smart Meeting Room aufzunehmen.

Eine umfassende Übersicht über verteilte Systeme und die in ihnen verwendete Middleware findet sich unter [EBM05].

2.1.3. Szenario des Smart Meeting Rooms in Rostock

Als konkrete Arbeitsumgebung für das Graduiertenkolleg MuSAMA dient das Smart Lab an der Universität von Rostock. Es verfügt über mehrere fest installierte Geräte und eine komplexe Infrastruktur, die das spontane Integrieren von persönlichen Geräten in das bestehende Geräte-Ensemble ermöglicht. Ein Foto vom Smart Labs ist in Abbildung 2.2 zu finden.

Auf bis zu sieben Leinwänden können die Bilder von mehreren Projektoren angezeigt werden. Eine Video-Matrix Crossbar verbindet bis zu 9 VGA-Anschlüsse an den Tischen des Raums und bis zu 7 VGA-Signale von einem Server mit den Projektoren an der Decke des Raums. Um technische Inkompatibilitäten beim Verbinden zu vermeiden

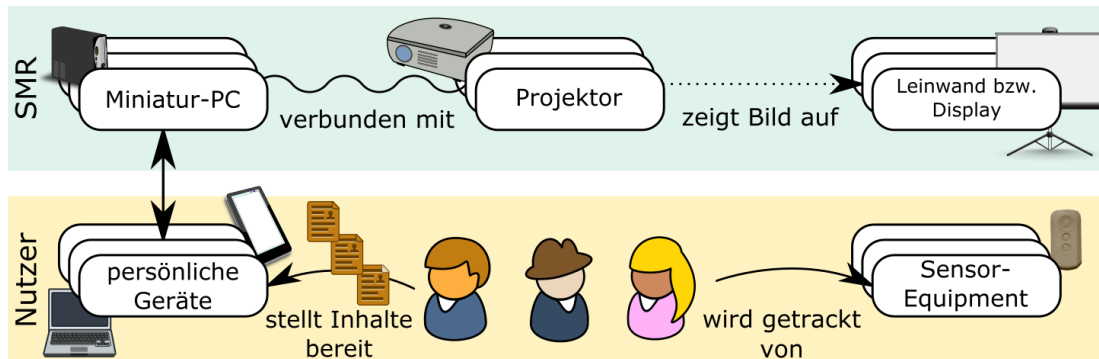


Abbildung 2.3.: Schematische Darstellung des Rostocker Smart Labs mit den persönlichen Geräten der Nutzer und der Sensor-Infrastruktur.

(z.B. auf Grund von verschiedenen Übertragungsschnittstellen wie VGA, HDM, DVI, DP), werden die persönlichen Geräte jedoch nicht direkt an die Projektoren des Raums angeschlossen. Stattdessen wurden drei Miniatur-PCs installiert, welche mit den Projektoren des Smart Labs verbunden sind. Software-seitig stellen die Miniatur-PCs eine Zugriffsschnittstelle bereit, die es ermöglicht, übertragene Bildinformationen auf jeder der Leinwände des Raums anzuzeigen. Die Miniatur-PCs sind hierfür über Ethernet an ein gemeinsames Netzwerk angeschlossen. Persönliche Geräte der Nutzer können sich zu diesem Netzwerk über Ethernet oder Wi-Fi verbinden, um Daten zu liefern oder zu empfangen. Neben einer Reihe von Kameras und Mikrofonen verfügt der Raum auch über Drucksensoren im Boden, die es ermöglichen, die Position von Personen zu bestimmen [LST13]. Das UbiSense-System [Ubi15] ermöglicht zudem eine Identifikation von Nutzern über tragbare Tags. Abbildung 2.3 zeigt eine schematische Übersicht der beschriebenen Geräte-Konfiguration. Im oberen Teil der Abbildung sind die fest installierten Geräte des Smart Meeting Rooms dargestellt. Der untere Teil zeigt den spontan in diese Umgebung integrierenden Nutzer, welcher seine persönlichen Geräten mitbringt und dessen Verhalten durch die Sensoren des SMRs aufgezeichnet wird.

Die Infrastruktur des Smart Labs wird durch eine Middleware [BK12] kontrolliert, welche verschiedene Paradigmen für das lose Zusammenschließen von Geräten bereitstellt und die Kontrolle über die Umgebung ermöglicht. Durch direkten Nachrichtenaustausch, Event-basierte Kommunikation und durch publish/subscribe-Mechanismen [BN12] wird eine flexible Kommunikation zwischen den Geräten, den verschiedenen Diensten und Anwendungen realisiert (und protokolliert). Installierte Softwarekomponenten verarbeiten Sensordaten und ermöglichen u.a. das Lokalisieren von Nutzern, das Erkennen von bestimmten Aktivitäten von Nutzern und ein Ableiten von einigen grundlegenden Nutzerabsichten [Krü+12; Krü+13]. Verschiedene Visualisierungen sind abrufbar, um den Zustand der installierten Geräte abzufragen und zu steuern.

2. Grundlagen und Stand der Forschung

Anwendungen werden im Rostocker Smart Lab von einem Web Server aus über Java Web Start gestartet. Das bedeutet, dass ein Nutzer über den Webbrowser auf seinem persönlichen Gerät Software-Komponenten des Raums starten kann. Diese Komponenten werden dann über die Middleware automatisch in die Infrastruktur des Raums integriert. Je nach gestarteter Anwendung können so z.B. Geräte im Raum ferngesteuert, Daten von und zu dem persönlichen Gerät übertragen oder die Interaktion auf dem Gerät mitgeschnitten werden.

2.1.4. Informationsanzeige in Smart Meeting Rooms

Mit steigender Display-Anzahl und Nutzern, welche die Informationsanzeige beeinflussen sollen, nehmen die Anforderungen an die Generierung, Verteilung und Anordnung der visuellen Darstellungen zu: So können einige Displays unter Umständen nicht für alle Personen gut einsehbar sein, weil sie z.B. durch andere Personen verdeckt werden, zu weit von einigen Nutzern entfernt sind, oder weil sich Displays nicht im Sichtbereich aller Nutzer befinden. Auch die Art des Geräts spielt hierbei eine Rolle: Anzeigegeräte mit kleinen Bildschirmen wie z.B. Smartphones oder Tablets bieten u.U. nicht die notwendige Anzeigefläche und Auflösung, um alle Details einer Darstellung gut erkennbar anzuzeigen.

Bei einer manuellen Verteilung von Informationsdarstellungen auf Ausgabegeräte müssten die Nutzer selber diese Faktoren berücksichtigen und ggf. ausgleichen. Insbesondere, weil die Anzahl und Auswahl der verwendeten Darstellungen sich häufig ändert, weil unterschiedliche Nutzer dies beeinflussen können und weil hierbei eine sich verändernde Display-Umgebung berücksichtigt werden muss, ist der hierfür notwendige Konfigurationsaufwand jedoch groß.

Für die Gestaltung der Informationsanzeige in Smart Meeting Rooms müssen somit folgende Probleme adressiert werden:

1. Die Darstellungen müssen an die heterogenen Displays der Umgebung angepasst werden.
2. Die wechselnden Darstellungen müssen auf die Displays verteilt und dort räumlich angeordnet werden.

Da hierfür manuelle Interaktion sehr aufwändig ist, müssen Ansätze entwickelt werden, welche dem Nutzer einen Teil dieses Konfigurationsaufwands abnehmen. Die o.g. Probleme stehen jedoch nicht im Fokus aktueller Forschung im Bereich von Smart Meeting Rooms, wurden aber speziell im Rahmen von MuSAMA adressiert.

Im Folgenden wird auf diese in MuSAMA entwickelten Ansätze eingegangen. Hierbei werden zunächst Ansätze besprochen, welche die Erzeugung einer Darstellung automatisch an das verwendete Ausgabegerät anpassen. Anschließend werden Ansätze für

Smart Meeting Rooms beschrieben, welche die Verteilung und Anordnung von Darstellungen auf mehreren Displays unterstützen.

Assistenz bei der Erzeugung von Darstellungen

Das Anpassen einer visuellen Darstellung an unterschiedliche Geräteeigenschaften ist Gegenstand von Forschungsfragen im Bereich der “Display Scalability” [TC05]. Insbesondere neuere Geräteklassen mit unterschiedlichen Displaygrößen, wie z.B. große, zusammengesetzte Display-Walls oder mobile Geräte mit kleinen Displays, stehen im Vordergrund aktueller Forschung [Pap+15; Mor12; JH13].

Eine allgemeingültige Lösung für unterschiedliche Geräte und Anwendungshintergründe ist hierbei jedoch nicht in Sicht. Für Smart Meeting Rooms wurden deshalb spezielle Assistenzlösungen entwickelt, welche die Informationsvisualisierung dynamisch an die Gegebenheiten anpasst.

In seiner Dissertation beschreibt Thiede wie die Darstellungen von Daten speziell auf das verwendete Ausgabegerät angepasst werden [Thi10]. Dazu werden eine Reihe von Adaptionismechanismen vorgeschlagen, die an unterschiedlichen Stellen der Visualisierung-Pipeline einsetzen: Durch Mechanismen die auf der Datenbasis arbeiten, können z.B. Datenwerte bei der Erzeugung der Visualisierung ausgeblendet oder zusammengefasst werden, um eine Anzeige auf kleinen Displays zu unterstützen. Mechanismen, die den Darstellungsraum betreffen, werden genutzt, um die Anzahl der grafischen Primitive zu reduzieren oder um beispielsweise Bildbereiche von Interesse vergrößert darzustellen. Durch ein von Radloff entwickeltes “redundantes Mapping” kann zudem das Mapping, also das Abbilden von Daten auf grafische Primitive, so beeinflusst werden, dass unterschiedliche Display-Größen adressiert werden [Rad+11; Rad14].

Die Verwendung der Adaptionismechanismen wird durch ein Assistenzsystem gesteuert, das die Geräteeigenschaften der verwendeten Displays (z.B. Auflösung und Größe der Displays) auswertet. Abbildung 2.4 zeigt hierfür den von Thiede entworfenen Entscheidungsbaum. Für jede Alternative in diesem Baum wurden entsprechende Ansätze entwickelt.

Die Besonderheit bei der Adaption der Darstellung ist die on-the-fly Erzeugung der Visualisierung: Die Darstellungen werden nicht vorberechnet, sondern dann erzeugt, wenn bestimmte Daten auf einem ausgewählten Display visualisiert werden sollen. Der Ansatz von Thiede entwirft hierfür eine verteilte Service-Architektur, bei der verschiedene Dienste die unterschiedlichen Adaptionsschritte übernehmen können.

Mit seiner Arbeiten leistet Thiede einen wichtigen Beitrag zur assistierten Informationsvisualisierung, indem die Generierung einer Darstellung sich automatisch an die zur Anzeige verwendete Geräte-Hardware anpasst. Im Folgenden werden Ansätze besprochen die automatisch entscheiden, welche Displays zur Anzeige einer Darstellung verwendet werden.

2. Grundlagen und Stand der Forschung

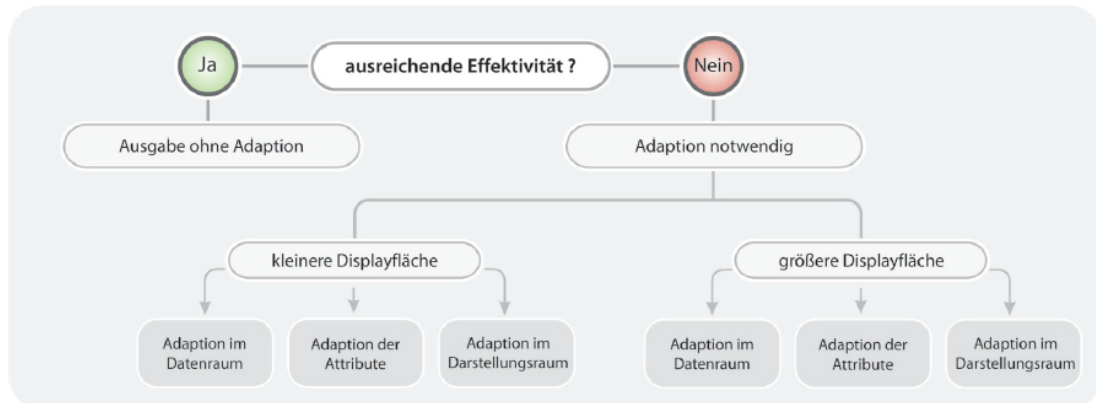


Abbildung 2.4.: Schematische Darstellung des von Thiede entworfenen Entscheidungsbaum zur Anwendung der Adaptionsmechanismen bei der angepassten Informationsvisualisierung. Abbildung aus [Thi10].

Assistenz bei der Inhaltsverteilungen

Um das Problem der unterschiedlichen Views und deren Verteilung auf die Displays eines Smart Meeting Rooms zu lösen, hat Heider ein Assistenz-System entworfen, das selbstständig eine Zuordnung von Darstellungen zu Displays berechnet [GH07; HK07]. Diese Zuordnung wird verwendet, um die Darstellungen ohne notwendige manuelle Interaktion anzuzeigen. Die durch Sensoren ermittelten Positionen und Blickrichtungen von Nutzern werden hierbei dazu verwendet, die Sichtbarkeit einer Darstellung auf einem Display zu bewerten.

Je nach verwendeter visueller Repräsentation, genutztem Display und Nutzer kann die Sichtbarkeit eines einzelnen visuellen Inhalts variieren. Diese Sichtbarkeit wird formal durch eine Qualitätsfunktion beschrieben. Um die Sichtbarkeit für eine Menge von Darstellungen und für mehrere Nutzer zu bewerten, werden die einzelnen Qualitätswerte aufsummiert. Das Ergebnis ist eine Qualitätsfunktion, die für eine Menge von Nutzern und für eine konkrete Zuordnung von Darstellungen auf Displays, die allgemeine Sichtbarkeit aller Darstellungen bewertet.

Analog wird auch eine weitere Qualitätsfunktion definiert, welche die Neuordnung von Darstellungen bei Veränderung der Inhaltsmenge, der Anzahl der Displays oder der Nutzer(-positionen) bewertet: Die zeitliche Qualität ist für eine Zuordnung hoch, wenn diese neue Zuordnung sich nur wenig von der zuletzt verwendeten Zuordnung unterscheidet und somit ein Umverteilen von bereits verwendeten Darstellungen vermeidet.

Der Assistenz-Algorithmus generiert die möglichen unterschiedlichen Zuordnungen und ermittelt diejenige Zuordnung mit der besten Gesamtqualität. Diese beste Zuordnung wird verwendet, um die visuellen Darstellungen automatisch auf die Displays zu verteilen.

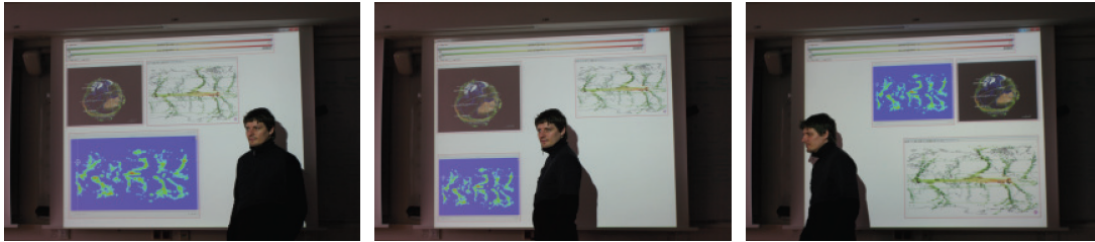


Abbildung 2.5.: Die Abbildungen zeigen eine automatische Anpassung der View-Anordnung, wenn der Präsentierende seine Position verändert. Die View-Layouts werden so berechnet, dass Verdeckungen vermieden werden. (Bild aus [Rad14]).

Der Ansatz von Heider beschränkt sich darauf, Zuordnungen zu berechnen, bei denen pro Display maximal eine Darstellung angezeigt wird. Die Anzahl der möglichen injektiven Abbildungen von der Menge der Darstellungen in die Menge der Displays ist $d!/(d-i)!$, wobei d die Anzahl der Displays und i die Anzahl der Darstellungen ist. Für das Smart Lab in Rostock mit 7 Displays kommen somit beispielsweise maximal 5040 verschiedene Abbildungen in Frage. Mit Algorithmen, welche jede Abbildung generieren und anschließend testen, sind diese Berechnungen selbst für bis zu 10 Displays in maximal einer Sekunde möglich.

Im Vergleich zur manuellen Anordnung der Darstellungen schneidet das Assistenz-System von Heider in einer Nutzerstudie [HK07; HK08] gut ab: Die vom System berechneten Zuordnungen entsprechen zu 95% den von den Nutzern gewünschten Verteilungen und die Zufriedenheit mit dem System ist im Durchschnitt höher als bei der manuellen Anordnung [HK07]. In weiterführenden Arbeiten wird auch die Anzeige von mehreren visuellen Darstellungen auf einem einzelnen Display durch eine assistierte Inhaltsanordnung unterstützt.

Assistenz bei der Inhaltsanordnung

In seiner Dissertation erweitert Radloff den Ansatz von Heider zum *Smart View Management System* [Rad14; Rad+15; RLS11]. Das System verteilt automatisch die Informationsdarstellungen über so bezeichnet *Views* auf die Displays und erlaubt es dabei, auch mehrere Darstellungen pro Display anzuzeigen. Hierfür werden alle Views automatisch in JPEG2000 kodiert, um sie besser skalieren und somit auf eine durch die Ausgabe-Displays geforderte Größe anpassen zu können. Für jedes Display wird separat ein Layout berechnet, das beschreibt, an welcher Stelle auf dem Display die visualisierten Inhalte platziert werden. Die Layouts werden so bestimmt, dass Inhalte möglichst gut sichtbar bleiben. Dazu werden einerseits überschneidungsfreie Layouts generiert, so dass Views sich nicht überdecken. Andererseits werden die Views so platziert, dass

2. Grundlagen und Stand der Forschung

Verdeckungen durch Personen, die vor den Displays stehen, vermieden werden [RFS12]. Abbildung 2.5 zeigt wie 3 Views auf einem einzelnen Display platziert werden. Die Anordnung der Darstellungen passt sich hierbei automatisch an, wenn eine Person sich vor dem Display entlang bewegt.

Um mehrere Views auf mehreren Displays anzuzeigen, wird, ähnlich wie bei Heider, eine Zuordnung von Views auf Displays berechnet. Hierbei werden einzelne Views auch automatisch vervielfacht und auf mehreren Displays angezeigt, um sicherzustellen, dass jeder Nutzer jede Darstellung sehen kann [Rad+11]. Anschließend wird für jedes Display ein Layout berechnet. Die Layouts werden grundsätzlich nach der gleichen Qualitätsfunktion wie bei Heider bewertet. Für die Beschreibung der Sichtbarkeit von Views berücksichtigt Radloff jedoch zusätzlich noch die menschliche Wahrnehmungsfähigkeit, um z.B. zu bewerten, ob bestimmte Inhalte ausreichend groß dargestellt sind. Für die Gruppierung von Views können interaktiv so bezeichnete “View Packages” definiert werden. Darstellungen aus einem View-Package werden, wenn möglich, auf dem gleichen Display platziert oder bei einer großen Anzahl von Views ggf. auf benachbarten Displays verteilt [Rad14].

Bei der Berechnung der Inhaltsverteilung und Anordnung wird versucht, eine Zuordnung und dazugehörige Layouts zu erzeugen, die eine möglichst hohe Gesamtqualität aufweisen. Durch die getroffenen Erweiterungen ist der mögliche Lösungsraum hierfür jedoch deutlich größer als bei Heider: Alleine die erweiterten Zuordnungsmöglichkeiten (ohne das Vervielfältigen von Darstellungen) ergeben d^i mögliche Abbildungen von Darstellungen auf Displays, wobei d die Anzahl der Displays und i die Anzahl der Darstellungen ist. Bei 7 Darstellungen für 7 Displays sind dies beispielsweise mehr als 8×10^5 Abbildungen, für die jeweils 7 Layouts berechnet werden müssen. Bei jeder Layout-Berechnung kommen wiederum grundsätzlich verschiedene Arten der Darstellungsanordnung (bspw. horizontal nebeneinander oder vertikal übereinander) und unterschiedliche Skalierungen der Darstellungen in Frage, was den möglichen Lösungsraum weiter vergrößert.

Eine effiziente Suche innerhalb dieser großen Lösungsmenge ist aufgrund der definierten Qualitätsfunktion jedoch schwierig: Werden beispielsweise zu viele Darstellungen auf einem einzelnen gut sichtbaren Display platziert, müssen diese jeweils kleiner dargestellt werden und sind somit in der Summe wieder schlechter wahrnehmbar. Dies ist auch der Fall, wenn Darstellungen mit ungünstigen (z.B. mit sehr unterschiedlichen) Seitenverhältnissen auf einem Display angeordnet werden: Um eine Überschneidung der Darstellungen zu vermeiden, müssen diese kleiner skaliert werden, was u.U. zu Layouts mit großen ungenutzten Anzeigeflächen führen kann.

Radloff berechnet die Inhaltsverteilung und Anordnung, indem er einen genetischen Algorithmus verwendet: Ausgehend von vielen wahrscheinlich guten Ausgangslösungen, kann durch leichtes Abändern eine geeignete Zuordnung von Views auf Displays und die dazugehörigen Layouts berechnet werden [Rad14].

Der Vorteil dieser Vorgehensweise ist, dass eine relativ gute initiale Standardlösung auch für große Inhaltsmengen in wenigen Sekunden gefunden werden kann. Dies ermöglicht

es dem Nutzer, schnell die Inhalte anzuzeigen und bietet einen Ausgangspunkt für anschließende manuelle Anpassungen. So können beispielsweise Nutzer nachträglich per Drag and Drop die ermittelten Positionen von Darstellungen in den Layouts anpassen und durch weitere Darstellungen schnell ergänzen [Rad+12]. Durch den genetischen Algorithmus wird auch bei nachträglichen Anpassungen eine gewisse zeitliche Kontinuität der Inhaltsanordnung erreicht, weil immer die aktuelle Inhaltsanordnung als Ausgangspunkt für eine Neuberechnung dient.

Nachteilig ist, dass die ermittelte (und verwendete) Lösung mit hoher Wahrscheinlichkeit nicht die optimale Verteilung der Views garantieren kann. Die ermittelte Lösung kann insbesondere dann weit von der besten Lösung entfernt sein, wenn weitere Vorgaben zur Anordnung der Inhalte gemacht werden, die nicht in den initialen Zuständen des genetischen Algorithmus enthalten sind. Die ist z.B. der Fall, wenn die Menge der angezeigten Views großteils ausgetauscht wird (was eine komplette Neuberechnung erfordert) oder wenn unregelmäßige Layouts eine gute Lösung wären.

Zusammenfassung

Die in diesem Abschnitt betrachteten Assistenzsysteme für Smart Meeting Rooms bieten Ansätze, um eine Informationsvisualisierung mit mehreren visuellen Darstellungen an eine konkrete Multi-Display-Umgebung und an eine bestimmte Raumsituation anzupassen. Der Nutzer kann bei den entwickelten Ansätzen die Menge der darzustellenden Inhalte bearbeiten [Rad14], manuell die Generierung der Darstellung beeinflussen [Thi10] und nachträglich die Inhaltsverteilung und Anordnung verändern [Rad+15]. Den Einfluss, den der Nutzer auf die Parametrisierung der Assistenzsysteme nehmen kann, ist jedoch begrenzt und nicht im Voraus planbar. Dies wäre jedoch erforderlich, um eine Informationspräsentation an gerade besprochene Diskussionsschwerpunkte anzupassen oder Vorträge vorzubereiten.

Bezüglich der in Abschnitt 1.2 beschriebenen Aufgaben bietet die aktuelle Assistenz keine Unterstützung zum Auffinden benötigter Inhalte und auch keine Möglichkeit die Inhalte von mehreren Nutzern zu koordinieren.

Die Verwaltung von Views innerhalb von “View Packages”, wie von Radloff vorgeschlagen [Rad14], ist zwar geeignet, um eine einzelne Darstellungsmengen zu strukturieren, für Präsentations- und Diskussionsszenarien, bei denen angezeigte Inhaltsmengen häufig ausgetauscht und gleiche Darstellungen über den zeitlichen Verlauf unterschiedlich verwendet werden, ist diese Beschreibung jedoch nicht mehr ausreichend. Hier müssen weitere Koordinationsmechanismen bereitgestellt werden.

Die Anzeige von Views auf den Displays wird durch die beschriebenen Konzepte für das automatische Verteilen und Anordnen dagegen bereits unterstützt. Allerdings sind hierbei die für die Berechnung verwendeten Ansätze ungeeignet, um nutzergetriebene Vorgaben in die Layout-Berechnung zu integrieren. Weil beispielsweise der genetische Algorithmus von Radloff bzgl. der verwendeten Qualitätsfunktion keine optimale Lösung für

2. Grundlagen und Stand der Forschung

die Darstellungsverteilung ermittelt, könnten Nutzervorgaben, welche zu sehr von den vordefinierten Standardlösungen abweichen, (zumindest teilweise) durch den Algorithmus ignoriert werden. Eine Erhöhung der erlaubten Berechnungszeit zur Generierung einer besseren Inhaltsverteilung ist zudem aufgrund des großen Lösungsraums nicht sehr vielversprechend. Auch hierfür müssen neue Lösungen gefunden werden.

Zusammenfassend lässt sich also sagen, dass die nutzergesteuerte Informationspräsentation bisher nicht gezielt im Fokus vorangegangener Forschung für Smart Meeting Rooms stand. Es sind somit weitere Untersuchungen notwendig, um insbesondere die Verteilung und das Anordnen von Views besser zu unterstützen. Hierfür ist es erforderlich, neue Ansätze zu entwickeln, die es Nutzern erlauben, schnell Inhalte zu teilen und deren Verwendung zu beschreiben. Diese Modellierung kann dann die Anwendung der Assistenzsysteme parametrisieren, um Nutzervorgaben bei der Anzeige, Verteilung und Anordnung zu berücksichtigen. Hierzu muss die formale Definition von geeigneten Verteilungen erweitert und eine neue Strategie zur Berechnung der Lösungen entwickelt werden.

2.2. Informationsanzeige in Multi-Display Environments

Während im Bereich der smarten Umgebung stärker an Ansätzen gearbeitet wird, welche hauptsächlich die technischen Aspekte bei der assistierten Gestaltung einer Informationsanzeige beleuchten, konzentrieren sich andere Arbeiten im Bereich der Multi-Display-Umgebungen darauf, wie Nutzer gezielt mehrere Visualisierungen (auf mehreren Displays) zur Erfüllung ihrer Aufgaben einsetzen können.

Bei diesen Ansätze stehen die Techniken im Vordergrund, die es Nutzern erlauben zu bestimmen, welche Darstellungen, zu welchem Zweck, wo angezeigt werden sollen, wie mehrere Darstellungen gemeinsam kombiniert werden können, um eine bestimmte Aufgabe (z.B. die Analyse von Daten) zu unterstützen und welche Koordinations- und Interaktionsmechanismen dabei eingesetzt werden, um die Informationsanzeige zu gestalten.

Ein Ausgangspunkt ist hierbei die Weiterentwicklung von bekannten Konzepten für die Verwendung von mehreren Darstellungen (bzw. "Views") in Single-Display-Umgebung und die Adaption dieser Konzepte für Multi-Display-Umgebungen. Dieses Vorgehen führte zur Entwicklung von View-Management-Systemen und zu Präsentationssystemen für Multi-Display-Umgebungen.

View Management-Systeme sind eine Weiterentwicklung aus den klassischen Window Management Systemen, wie sie z.B. durch Betriebssysteme (Linux X-Server, Windows) bereitgestellt werden. Der Fokus dieser Ansätze liegt darauf Nutzern zu ermöglichen, Views dynamisch zu definieren [TMC04; WL07; YW14], sie (manuell) auf die Displays zu verteilen [BB06; Bie+08; BB04] und die Darstellungen interaktiv auf einem Display anzuordnen [Wal+11a; Wal+11b; Bie+08; Wig+09;

Joh+02]. Die Ansätze werden um Display-übergreifende Interaktion [Rad+12; Bie+08; Joh+02] und die Fähigkeit zum Teilen von Ansichten über Systemgrenzen hinweg [Wal11; TMC04] erweitert. Mit Hilfe dieser Systeme können mehrere Nutzer gemeinsam die Menge der zu verwendenden Darstellungen, ihre Verteilung und Anordnung auf den Displays zum Zeitpunkt ihrer Verwendung (z.B. während einer Besprechung) interaktiv anpassen.

Präsentationssysteme für Multi-Display-Umgebungen haben das Ziel, die Anzeige der Darstellungen in der Multi-Display-Umgebung im Voraus planbar zu machen. In Vorbereitung einer Präsentation können hierfür beispielsweise mehrere Folienfolgen für unterschiedliche Displays durch die Nutzer vorbereitet und dann während einer Besprechung zur Unterstützung eines Vortrags abgerufen werden [LBT08; Liu+05; Spi+12; Zha+04].

Beide Ansätze ermöglichen es grundsätzlich, mehrere Displays für die Informationsanzeige zu nutzen.

Die einzelnen Displays werden jedoch weiterhin als separate Anzeigeräume betrachtet, welche jeweils zur Anzeige einer (durch den Nutzer) festgelegte, meist statischen Auswahl an Darstellungen genutzt werden. Der Zugriff auf die Displays ist zudem bei den meisten Ansätzen nur von bestimmten Geräten aus (und damit nur für ausgewählte Nutzer) möglich. Außerdem wird typischerweise angenommen, dass einzelne Darstellungen unabhängig voneinander sind. Zusammenhänge zwischen mehreren Darstellungen werden von den Ansätzen deshalb i.d.R. nicht berücksichtigt.

Diese Einschränkungen erschweren es, diese Systeme für komplexe Aufgaben einzusetzen, bei denen mehrere Nutzer aktiv Einfluss auf die Gestaltung der Darstellungsanordnung auf mehreren Displays nehmen und dabei mehrere unterschiedliche Darstellungen in Kombination nutzen müssen. Dies trifft beispielsweise auf Szenarien zu, bei denen unterschiedliche Schritte einer gemeinsamen Datenanalyse, jeweils durch andere Visualisierungen gesteuert und unterstützt werden sollen.

Unter dem Begriff der *Display Ecology* wird deshalb eine weiterentwickelte Verwendung einer Multi-Display-Umgebung zusammengefasst [Chu+15; Wig+09; Chu+14]. Nach Chung et al. müssen folgende grundlegenden Herausforderungen adressiert werden, um eine Multi-Display-Umgebung zu einer Display Ecology weiterzuentwickeln [Chu+15]:

- C1:** Die einzelnen Displays müssen zu einem gemeinsamen und durch alle Nutzer verwendbaren Anzeigeraum verbunden werden.
- C2:** Die Nutzer müssen in die Lage versetzt werden, Daten und Darstellungen auszutauschen, um die Verteilung von Informationen gemeinsam koordinieren zu können.
- C3:** Es muss Nutzern möglich sein, einzelne Ansichten, auch über Displaygrenzen hinweg, miteinander zu verbinden.
- C4:** Einzelne Displays und Darstellungen müssen vom Nutzer, je nach Bedarf, hinzugefügt bzw. wieder entfernt werden können.

2. Grundlagen und Stand der Forschung

Durch diese Punkte soll ein gemeinschaftliches Zusammenarbeiten unterstützt werden, bei dem neben Präsentationsaufgaben insbesondere auch eine Datenauswertung (z.B. mit Hilfe von Visual Analytics) vorgenommen werden kann. Hierfür soll es auch möglich sein, einzelnen Displays unterschiedliche Rollen zuzuweisen und so speziell dazu zu nutzen, um Teilaufgaben in einem größeren, zusammengesetzten Analyseprozess zu übernehmen [Chu+15].

Bei dem Entwurf von Systemen für Display-Ecologies sind verschiedene Vorgehensweisen möglich, um die Herausforderungen C1-C4 zu adressieren. Nach Chung et al. sind Design-Entscheidungen für folgende Bereiche zu treffen [Chu+15]:

- **Display Composition:** Beschreibt die Art und Weise, wie einzelne Displays zur Informationsdarstellung ausgesucht werden, und wie mehrere Displays zusammen genutzt werden, um multiple Darstellungen anzuzeigen.
- **Information Coordination/Transfer:** Beschreibt, wie Nutzer Informationen zu den Displays übertragen, auf die Displays verteilen und auf ihnen anordnen.
- **Information Connection:** Beschreibt die Möglichkeiten von Nutzern über mehrere Displays verteilte Darstellungen miteinander zu verknüpfen und zusammenzuführen.
- **Display Membership:** Betrachtet Aspekte wie mit dem ad-hoc Hinzufügen und Entfernen von Display umgegangen wird.

Eine umfassende Übersicht aktueller Lösungsansätze ist in der Abbildung 2.6 zu sehen. Die aus [Chu+15] entnommene Tabelle ordnet hierbei die Literatur bzgl. der Design-Entscheidungen ein.

Aus dem Übersichtsartikel wird ersichtlich, dass es nur wenige Ansätze gibt, welche den Nutzer durch automatische Konfiguration unterstützen. Software-Systemen, welche dafür entwickelt wurde neuartige Formen des kollaborativen Arbeitens zu unterstützen, fehlt zudem oft die Fähigkeit mit dynamisch zusammengestellten Geräte-Ensembles umzugehen und hierbei Assistenz bzw. Situationserkennung zu integrieren.

Andererseits adressieren die Systeme für SMRs (vgl. Abschnitt 2.1) schon teilweise die Herausforderungen, indem sie die Displays des Raums und die persönlichen Geräte der Nutzer verbinden (C1), es erlauben, das Geräteensemble durch Hinzufügen bzw. Entfernen von Geräten anzupassen (C4) und grundlegende Werkzeuge bereitstellen, um den Informationsaustausch zwischen den Geräten vorzunehmen (C2).

Eine semantische Koordinierung der Darstellungsanzeige und die Verwendung der vorhandenen Assistenzmöglichkeiten, um eine vom Nutzer verfolgten Aufgabe zu unterstützen (wie es die Systeme für Multi-Display-Umgebungen vorsehen), bleibt jedoch offen. Insbesondere das hierfür notwendige, dynamische Herstellen von Verbinden zwischen den Darstellungen (C3) von verschiedenen Nutzern und das gezielte Kombinieren von mehreren Darstellungen auf unterschiedlichen Displays, ist eine weitestgehend unbehandelte Problemstellung.

2.2. Informationsanzeige in Multi-Display Environments

Design Considerations	Sub-category	Related Visual Analysis Tasks	Target Challenges
Display Composition	Display Composition	<ul style="list-style-type: none"> Remote Selection / Filter: Remotely isolating a subset of data items for analysis using dynamic query slider [1], [2], [3] Search: Finding related information [2] Physical Navigation: [1], [3] 	Semi-Collaborative Opportunistic Analysis
	Data View: Single Continuous	<ul style="list-style-type: none"> Pan and Zoom: Physically navigate the increased visualization space from multiple displays [1], [3], [10], [11] 	Scalability
	Data View: Multiple Coordinated	<ul style="list-style-type: none"> Compare: Juxtaposition of visualization views [12], [13] Complement: Analyze multiple views of the same dataset [14], [15] Split: Decomposing a visualization view / entity into multiple views / entities on multiple displays [16] 	Heterogeneous Data
	Data View: Navigation Metaphor	<ul style="list-style-type: none"> Focus + Context [18], [20] Overview + Detail [14], [22], [24] Navigate: Traverse the visualization space [14] 	Scalability
	Data View Semantic Substrates	<ul style="list-style-type: none"> Organize: Managing data views [27], [4], [7] Navigate: Traverse the visualization space [27], [4], [7] 	Heterogeneous Data Sensemaking
Information Coordination/ Transfer	Synchronize Surrogate Nominal Physical	<ul style="list-style-type: none"> Publish: Broadcasting updates to a group of devices [31], [22], [16] Share: Exporting views and meta content such as annotations, bookmarks to other displays [33], [35] 	Sensemaking Heterogeneous Data
Information Connection	Overview	<ul style="list-style-type: none"> Merge: Aggregating multiple views [36], [38], [39] 	Sensemaking Provenance
	Explicit	<ul style="list-style-type: none"> Annotate / Bookmark: Manually pointing important information on different displays and devices/Add metadata to visualization elements [41] Highlighting: Selecting related visual entities [7], [41] Navigate/Guide: Using Cross Display Links [41] 	Sensemaking
	Implicit	<ul style="list-style-type: none"> Cluster/Filter: Creating semantic structure [4], [7] Organize: Spatially organize documents based on different displays [27], [4], [7] Sort: Ordering of data items/ entities [12], [3] 	Sensemaking
Display membership	Pre-designed	N/A	N/A
	Ad-hoc	N/A	Opportunistic Analysis

Abbildung 2.6.: Überblick über die, nach den Design-Entscheidungen von Chung et al. geordnete, Literatur zu Display Ecologies (Tabelle aus [Chu+15]).

In der vorliegenden Dissertation werden diese Probleme adressiert. Hierfür wird zunächst eine Modellbeschreibung benötigt, mit der sich die Zusammenhänge zwischen den Darstellungen beschreiben lassen (C3) und die als Grundlage für das Einbinden von Assistenz aus Smart Meeting Rooms dienen kann. Die hierfür entworfenen Modellbeschreibungen werden im folgenden Kapitel beschrieben.

3. Modellentwicklung

In dieser Dissertation werden Szenarien adressiert, bei denen wechselnde Themen und Interessen den Verlauf von Präsentationen und Diskussionen ändern können und es somit erforderlich wird, spontan andere Daten oder Dokumente von unterschiedlichen Nutzern und auf wechselnden Displays zu zeigen. Hierfür soll eine umfassende Nutzerunterstützung gewährleistet werden.

Im Folgenden werden zusammenfassend die Informationen als **Inhalte** bezeichnet. Inhalte können beispielsweise Vorträge in Form von PDF-Dateien oder Abbildungen in Form von JPEG-Dateien sein. Die gerenderte grafische Repräsentation eines konkreten Inhalts zur Anzeige auf einem Display wird, in Anlehnung an die Arbeiten von Radloff, als **View** bezeichnet [Rad14].

Für die Anzeige und das Arbeiten mit Views soll Assistenz bereitgestellt werden.

Die in dieser Dissertation entworfenen Grundkonzepte für die Nutzerassistenz sind hierbei:

- Die **assistierte Präsentation** von spontan bereit gestellten Inhalten, indem die Verteilung und Anordnung der Views auf den Displays automatisch berechnet und aktualisiert wird.
- Für die **assistierte Diskussion** wird die Interaktion unterstützt. Hierbei werden, für einen Undo/Redo-Mechanismus, durch die Nutzer vorgenommenen Anpassungen aufgezeichnet und strukturiert. Basierend auf diesen Aufzeichnungen können Nutzer, in Hinblick auf wechselnde Diskussionsschwerpunkte, schnell zwischen verschiedenen View-Zusammenstellungen hin- und herwechseln. Darüber hinaus wird die Suche durch das Empfehlen von semantisch passenden Inhalten unterstützt.

Für diese Assistenzansätze sind umfassende Modellbeschreibungen notwendig, welche *i)* die zeitliche und räumliche Verwendung, sowie die Zusammenhänge zwischen Views modellieren, *ii)* die thematische Beziehung zwischen Inhalten beschreiben und *iii)* die interaktive Anpassungen der Informationsanzeige protokollieren. Die hierfür entwickelten Modellbeschreibungen sind der Präsentationsgraph, der Semantische Graph und der Interaktionsgraph.

In diesem Kapitel werden die einzelnen Modelle nacheinander in den Abschnitten 3.1, 3.2 und 3.3 vorgestellt.

3.1. Präsentationsgraph

Der Präsentationsgraph dient als Grundlage für die assistierte Präsentation der Inhalte (vgl. Kapitel 4). Er wird von den Nutzern erstellt, beinhaltet die anzuzeigenden Views und beschreibt Vorgaben für die räumliche Verteilung und die zeitliche Ersetzung von Views auf den Displays des Raums.

3.1.1. Problembeschreibung

Bisherige Ansätze für Smart Meeting Rooms erlauben es, Views automatisch auf die Displays zu verteilen und auf den verfügbaren Anzeigeflächen anzuordnen (vgl. Abschnitt 2.1.4). Nutzervorgaben zur Gestaltung der Informationsanzeige, werden hierbei jedoch im Allgemeinen nicht berücksichtigt.

Systeme für Multi-Display-Umgebungen erlauben es dagegen, diese Nutzervorgaben für die Kombination und zeitliche Verwendung von mehreren Views auszuwerten (vgl. Abschnitt 2.2). Allerdings nehmen diese Ansätze meist auch an, dass die verwendete Display-Konfiguration statisch festgelegt ist und dass die Menge der verwendeten Views und das Layout auf den Displays nur selten angepasst werden. Dies kann bei dem in der vorliegenden Dissertation adressierten Diskussionsszenario nicht vorausgesetzt werden. Zum einen werden hier, je nach wechselndem Themenschwerpunkt in der Diskussion, Informationen durch verschiedene Nutzer aufgerufen, in den Vordergrund gerückt oder ausgeblendet. Andererseits erfordert auch das Hinzufügen bzw. Entfernen von einzelnen Displays eine Neuordnung der Views. Dies ist ein wesentlicher Bestandteil, um (für die nutzergesteuerte Informationsanzeige) eine Multi-Display-Umgebung im Sinne einer Display-Ecology verwenden zu können (vgl. Anforderungen C1 und C4 für Display Ecologies in Abschnitt 2.2).

Ziel dieser Arbeit ist es, die Neuordnung von Informationsdarstellungen durch eine geeignete Assistenz zu unterstützen und Nutzer hierbei zu entlasten.

Basierend auf vorangegangenen Arbeiten im Bereich der Smart Meetings Rooms kann angenommen werden, dass Assistenzsysteme die verfügbaren Displays und ihre Eigenschaften automatisch ermitteln können und, dass dabei auch das spontane Hinzufügen und Entfernen von Displays berücksichtigt wird.

Zur Realisierung des in dieser Arbeit angestrebten Assistenzansatzes ist deshalb ein Modell erforderlich, das zwar Nutzervorgaben beschreibt, aber keine gerätebezogenen Informationen enthält, da diese Funktionalität vom Smart Meeting Room bereitgestellt wird.

Darüber hinaus soll das Modell nicht nur dazu dienen die Informationsanzeige zu steuern, sondern soll es auch ermöglichen, die zeitliche Verwendung von Inhalten in Vorbereitung einer Präsentation zu planen und dazu Beiträge unterschiedlicher Nutzer zu koordinieren und aufeinander abzustimmen.

Beim Entwurf der Modellbeschreibung werden dementsprechend folgende Anforderungen berücksichtigt:

- A1:** Es müssen Inhalte bestimmt werden können, die zeitgleich für die Informationsanzeige verwendet werden sollen.
- A2:** Zusammenhänge zwischen den Views müssen beschrieben werden,
 - A2.1** um Vorgaben für die zeitliche Anzeige und Ersetzung von Views beschreiben zu können,
 - A2.2** um Vorgaben für die relative, räumliche Positionierung von mehreren, gleichzeitig angezeigten Views zu ermitteln und
 - A2.3** um inhaltliche Zusammenhänge zu beschreiben.
- A3:** Es muss möglich sein, einzelne Views zu priorisieren, um ihnen beispielsweise mehr Fläche des Anzeigeraums zuweisen zu können.

Im folgenden Abschnitt wird die anhand dieser Anforderungen entworfene Modellbeschreibung vorgestellt. Die Generierung des Modells und das Assistenzsystem, das diese Vorgaben für eine konkrete Display-Umgebung auswertet und die Verteilung der Views automatisch vornimmt, wird in Kapitel 4 beschrieben.

3.1.2. Modellbeschreibung

Das entwickelte und als *Präsentationsgraph* bezeichnete Modell, ist ein attributierter Graph $G_P = (V_P, E_P, f_P)$. V_P ist die Menge der Knoten, $E_P = V_P \times V_P$ bezeichnet die Kanten, welche die Verbindungen zwischen den Knoten beschreiben und die Funktion f_P beschreibt die Relevanz-Werte der Knoten.

- **Knoten:** Jeder Knoten verweist auf einen einzelnen Inhalt, der für die Informationsanzeige verwendet werden soll indem zu ihm ein View erzeugt und auf den Displays angezeigt wird. Für eine zeitliche Ordnung der Knoten, wird die Knotenmenge in mehrere Schichten bzw. Zeitschritte aufgeteilt (Anforderung A1). Eine Schicht enthält hierbei jeweils diejenigen Knoten $V_i \subseteq V_P$, die zu einem bestimmten Zeitpunkt i gehören. Die nach i sortierten Knotenmengen V_i beschreiben somit den geplanten zeitlichen Verlauf einer Präsentation. Um eine Präsentation in einer Multi-Display-Umgebung anzuzeigen, werden die Inhalte der Schichten nacheinander zur View-Erzeugung verwendet. Hierbei ersetzen jeweils die Views der aktuellen Schicht die Views der vorangegangenen Schicht auf den Displays des Raums.
- **Kanten:** Die Kanten beschreiben die Relationen der Knoten im Kontext einer Präsentation. Hierbei werden drei Arten von Kanten eingeführt:

3. Modellentwicklung

- **Zeitliche Kanten** verbinden Knoten aus verschiedenen Zeitschritten miteinander. Während die Knotenschichten beschreiben *welche* Inhalte zu bestimmten Zeitpunkten angezeigt werden sollen, beschreibt diese Art von Kanten *wie* die Inhalte von aufeinanderfolgenden Zeitschritten über den zeitlichen Verlauf der Präsentation miteinander verbunden sind.

Sie definieren, wie sich bei einem Wechsel von einem Zeitschritt zum nächsten die neuen Layouts für die View-Anordnung an den alten Layouts des vorangegangenen Zeitpunkts orientieren sollen (Anforderung A2.1).

Eine zeitliche Kante $e = (v_1, v_2)$ kennzeichnet, dass der Inhalt von v_1 durch den Inhalt von v_2 ersetzt werden soll. Daher wird angenommen, dass die zu v_1 und v_2 erzeugten Views an der ungefähr gleichen Stelle auf den Displays positioniert werden sollen.

- **Räumliche Kanten** verbinden Knoten derselben Knotenschicht. Sie definieren Vorgaben für die räumliche Anordnung der Views in einer Multi-Display-Umgebung (Anforderung A2.2). Diese Kanten verbinden Knoten, deren Views räumlich nahe beieinander auf den Displays positioniert werden sollen.

Die Knotenschichten modellieren zusammen mit den zeitlichen und räumlichen Kanten welche Inhalte angezeigt werden und welche Vorgaben zur räumlichen Anordnung und zeitlichen Ersetzung der hierfür verwendeten Views berücksichtigt werden sollen. Um darüber hinaus im Kontext einer Präsentation miteinander in Beziehung stehende Inhalte zu kennzeichnen (Anforderung A2.3), werden gruppierende Kanten verwendet.

- **Gruppierende Kanten** definieren Verbindungen zwischen Inhalten die aufgrund des Themengebiets bestehen. Diese können von Nutzern definiert werden, um semantische Zusammenhänge zwischen einzelnen Inhalten für eine geplante Präsentation zu kennzeichnen. Auf diese Weise können Nutzer auf Hintergrundinformationen verweisen und Verbindungen zwischen verschiedenen Stellen einer Präsentation herstellen.
- **Relevanzwerte** werden als Attribute den Knoten zugewiesen: $f_P : V_P \rightarrow \mathbb{R}$. Der Relevanzwert gewichtet die Bedeutung eines einzelnen Inhalts gegenüber allen anderen Inhalten des gleichen Zeitschritts. Ist der Relevanzwert eines Knotens im Vergleich zu den anderen Knoten hoch, wird für die View des Knoteninhalts mehr Platz auf den Displays eingeräumt. Beim Erstellen eines Knotens wird der Relevanzwert auf einen Standardwert gesetzt. Durch nachträgliches Ändern des Werts können einzelne Inhalte bei der Informationsanzeige priorisiert werden (Anforderung A3).

Der Präsentationsgraph ist dafür ausgelegt, durch mehrere Nutzer gleichzeitig editiert und spontan während einer Präsentation erweitert zu werden. Hierfür können Nutzer

eigene Inhalte als neue Knoten in eine bestehende Präsentation einbringen oder unterschiedliche (Teile von) Präsentationen zu einem gemeinsamen Graphen verbinden.

Eine Besonderheit ist hierbei, dass Knoten nicht ausschließlich auf abgespeicherte Inhalte (wie z.B. PDF-Dokumente oder Bilder) verweisen müssen, sondern auch Tools zur interaktiven Erzeugung von neuen Views mit der Präsentation verlinken können. Diese *spontan generierten Views* können bei Bedarf während der Präsentation, durch Bedienung der Tools, angepasst werden und fügen sich automatisch an der modellierten Stelle in den Präsentationsgraph ein. Dieser Ansatz erlaubt es, Tools (z.B. zur Daten-Visualisierung, zur Integration von interaktiven Web-Inhalten oder zur Bearbeitung der Anzeige (wie z.B. Lupen)) direkt für die Gestaltung der Informationsanzeige, und damit zur Unterstützung einer Präsentation bzw. Diskussion, zu nutzen.

Auf diese Weise können spontan neue Anforderungen an die Visualisierung der Inhalte bedient werden, indem z.B. neue Sichten auf die Daten erzeugt oder zusätzliche Details eingeblendet werden. Um die Tools mit der restlichen Präsentation zu verknüpfen, werden gruppierende Kanten verwendet. Eine genauere Beschreibung wie spontan generierte Views für eine assistierte Präsentation genutzt werden, wird in Abschnitt 4.3 gegeben.

3.1.3. Ergebnisse

Mit den Knoten des Präsentationsgraphen kann eine Auswahl an Inhalte definiert werden, die für die Informationsanzeige in einer Multi-Display-Umgebung verwendet werden sollen. Insbesondere die Möglichkeit, hierbei auch Tools zur spontanen Generierung von neuen Views mit der Präsentation zu verlinken, ist ein neuartiger Beitrag zur Modellierung von Multi-Display-Präsentationen.

Durch die Unterteilung der Knoten in mehrere zeitlich geordnete Schichten, sowie durch die zeitlichen und räumlichen Kanten und die Relevanzwerte beschreibt der Präsentationsgraph die räumliche Verteilung und zeitliche Verwendung von Inhalten.

Die zeitliche Strukturierung erlaubt es, den Ablauf einer Präsentation zu planen und Beiträge unterschiedlicher Nutzer geordnet zu kombinieren.

Die räumliche Modellierung durch Relevanzverhältnisse und räumliche Kanten ist robust gegen (spontane) Änderungen der Knotenmenge. Das bedeutet, dass ein Hinzufügen bzw. Entfernen eines Knotens zwar die Verhältnisse zwischen den Relevanzwerten der Views beeinflusst, die resultierende Beschreibung jedoch weiterhin (ohne zusätzliche Nutzeranpassungen am Modell) eine gültige Beschreibung für eine räumliche View-Verteilung ist.

Zudem ist die Beschreibung geräteunabhängig, wodurch keine Anpassungen des Modells an eine konkrete Display-Umgebung bzw. bei einer Veränderung der Display-Zusammenstellung notwendig werden.

Der Präsentationsgraph bietet somit die Möglichkeit, um Nutzervorgaben für eine assistierte View-Verteilung in einer Multi-Display-Umgebung zu beschreiben (vgl. Kapi-

3. Modellentwicklung

tel 4). Auf diese Weise werden die Anforderungen zur Schaffung einer Display Ecology für die Informationsanzeige durch den Präsentationsgraph adressiert (vgl. Abschnitt 2.2). Insbesondere erlaubt es die räumliche Modellierung, die Anzeigeflächen auf unterschiedlichen Geräten flexibel zu kombinieren (C1), um ausgewählte Informationen von verschiedenen Nutzern gemeinsam anzuzeigen (C2) und hierbei auch auf Veränderungen in der Display-Umgebung (C4) reagieren zu können.

Der Präsentationsgraph modelliert bereits räumliche und zeitliche Verbindungen bei der Verwendung von Inhalten und adressiert damit auch teilweise die Anforderung C3 an eine Display Ecology. Verknüpfungen zwischen Inhalten können jedoch nicht nur aufgrund ihrer gemeinsamen Verwendung zur Gestaltung einer Informationsanzeige bestehen, sondern auch basierend auf thematischen bzw. semantischen Zusammenhängen. Zur Modellierung dieser semantischen Beziehungen ist eine weitere, vom Präsentationsgraph unabhängige, Beschreibung notwendig.

3.2. Semantischer Graph

Um Zusammenhänge zwischen Inhalten zu kommunizieren, ist eine Modellbeschreibung notwendig, die es erlaubt, thematische Beziehungen zu beschreiben.

Als typische Ansätze für die Wissensrepräsentation werden hierbei semantische Netze verwendet. Diese sind einerseits dafür ausgelegt vom Mensch verstanden und andererseits auch von IT-Systemen verarbeitet werden zu können [Rei10; Kha15].

Semantische Netze beschreiben und verknüpfen Objekte miteinander. Basierend auf diesen Informationen können IT-Systeme einen Sinn für thematische Entfernung, den Kontext und Ähnlichkeit entwickeln. Mit zunehmend formalerer Repräsentation des Wissens und Anspruch an die automatische Auswertbarkeit, können semantische Netze hierbei grundlegend in Assoziative Netzwerke, Themennetzwerke bzw. Faktennetzwerke und Ontologien unterteilt werden [Rei10].

- Assoziative Netzwerke beschreiben Inhalte durch mehrere Schlagworte. Diese sog. Tags können automatisch z.B. durch Dokumentanalyse gewonnen werden. Durch Clustern von Inhalten nach Tags werden Zusammenhänge und Themengruppen automatisch abgeleitet. Auf diese Weise sind sie gut dafür geeignet, um verschiedene Wissensbestände zu vereinen. Außerdem ist typischerweise nur ein geringer manueller Aufwand zum Erstellen dieser Art von Netzen notwendig. Andererseits ist ggf. die Qualität der gewonnenen semantischen Beziehung weniger gut als bei anderen Ansätzen.
- Themennetzwerke erweitern und formalisieren diese Modellierung weiter und erlauben es, explizite Beziehungen und anspruchsvollere Konstrukte (wie z.B. Querverbindungen und hierarchische Ordnungen) zu beschreiben. Sie ermöglichen es Nutzern, auf diese Weise genauer die Stellung bzw. Rolle eines Inhalts bezüglich

des Themengebiets zu beschreiben und somit eine höhere Qualität der semantischen Beschreibung zu erreichen.

- Ontologien zielen darauf ab, weiterführende Schlussfolgerungen und Zusammenhänge automatisch aus einer großen Wissensbasis abzuleiten und ggf. dem Nutzer den hierbei ablaufenden Prozess der Wissensgewinnung zu erklären. Sie werden hauptsächlich von Expertensystemen verwendet, um komplexe Anfragen oder Suchanfragen zu bearbeiten. Um die hierfür notwendige Verarbeitbarkeit der Daten zu gewährleisten, wird eine (im Vergleich zu anderen Arten von semantischen Netzwerken) komplexere und formalere Datenrepräsentation gewählt. Eine umfassende Übersicht von Ontologie-basierten Systemen findet sich in [DWL15].

Für eine geeignete semantische Modellierung ist eine Abwägung zwischen dem manuellen Aufwand zur Verwaltung der Struktur, den Eigenschaften, die es erlauben Inhalte präzise bezüglich eines Themengebiets einzuordnen und der Möglichkeit die Informationen automatisch auszuwerten notwendig. Die Wahl eines geeigneten Ansatzes zur Beschreibung von semantischen Verbindungen orientiert sich somit an der vorliegenden Problemstellung.

3.2.1. Problembeschreibung

Das in dieser Dissertation verfolgte Anwendungsszenario, sieht vor, dass verschiedene Nutzer spontan verschiedene Arten von Inhalten für eine Präsentation miteinander teilen und für diese semantische Informationen bereitstellen und ggf. ergänzen. Hieraus ergeben sich folgende Anforderungen an die semantische Modellierung:

- B1:** Die Art der Beschreibung muss für jeden Inhalte individuell festlegbar sein. Die Inhalte können nicht nach einem allgemeingültigen Muster beschrieben werden, da heterogene Arten von Inhalten jeweils eigene Beschreibungen benötigen, verschiedene Nutzer uneinheitliche Beschreibungen bereitstellen oder die Verfügbarkeit von semantischen Informationen von Inhalt zu Inhalt variieren kann.
- B2:** Mehrere Beschreibungen aus verschiedenen Quellen müssen miteinander kombiniert werden können, um die Wissensbestände der unterschiedlichen Nutzer zu vereinen und die verfügbaren Inhalte über ein einzelnes semantisches Netzwerk zu verbinden.
- B3:** Spontanes (manuelles) Anpassen und Ergänzung dieser bereitgestellten Beschreibung während der Präsentation bzw. Diskussion muss unterstützt werden. Hierfür ist es wichtig, dass die Art der Modellierung schnell von den Nutzern erfasst und bearbeitet werden kann.
- B4:** Ähnlichkeiten zwischen Inhalten müssen automatisch durch Auswertung der semantischen Informationen gewonnen werden können, um eine automatische Assistenz (z.B. zum Empfehlen von "passenden Inhalten") bereitstellen zu können.

3. Modellentwicklung

Bekannte Ansätze zu semantischen Netzwerken unterstützen gezielt einzelne dieser Anforderungen [Rei10]. In Hinblick auf Präsentationszwecke erlauben Assoziative Netze, z.B. zum Beschreiben von Bildersammlungen [Hu+14] oder zum Ordnen von automatisch gewonnenen Informationen über Präsentationsmaterialien [WS11; KOS09; HTK08], eine sehr freie Inhaltsbeschreibung (Anforderung B1). Außerdem existieren verschiedene Ansätze, um Verbindungen zwischen den Inhalten automatisch zu extrahieren [Hu+14] (Anforderung B4). Ontologie-basierte Systeme verfolgen hierfür noch flexiblere Ansätze um möglichst geeignete Verbindungen zwischen den Inhalten automatisch herzustellen, indem beispielsweise der Kontext einer Suchanfrage berücksichtigt wird [Den+13]. Hierfür verwenden sie vergleichbar formale, nicht spontan erweiterbare Beschreibungen. Wegen den Anforderungen B1-B3 sind diese Ansätze (im Vergleich mit den anderen Modellierungsarten) deshalb weniger gut geeignet.

Themennetzwerke, die durch mehrere Nutzer bearbeitet werden [Ber+10] oder die automatisch Beziehungen zwischen verschiedenen Inhalten herstellen können [Spi+12], erlauben es, die Inhalte gezielt für eine Präsentation oder Diskussion zu ordnen (Anforderung B2 und B3). Die Modellbeschreibungen werden jedoch meist für einen einzelnen Vortrag bzw. ein spezielles Themengebiet erstellt und adressieren deshalb weniger die Kombination von Informationen zu verschiedenen Themen.

Eine Lösung, welche alle Anforderungen bzgl. der verfolgten assistierte Präsentation und Diskussion vollständig abdeckt, ist noch offen. Im Rahmen dieser Dissertation stehen in erster Linie das (manuelle) Bearbeiten, die Verwendung und die hierfür notwendige Kombination der semantischen Informationen mit den anderen Modellbeschreibungen im Vordergrund. Aus diesem Grund soll im Rahmen dieser Arbeit keine komplett neue Art eines semantischen Netzwerks entwickelt werden. Stattdessen wird für die semantische Modellierung auf bestehende Ansätze aufgesetzt und diese entsprechend erweitert, um die Anforderungen B1-B4 zu adressieren.

3.2.2. Modellbeschreibung

Zur Modellierung der semantischen Informationen über die Inhalte und ihre Verbindungen zueinander wurde der sog. *Semantische Graph* entworfen. Der Graph $G_S = (V_S, E_S, A_S)$ besteht aus Knoten V_S , Kanten E_S und Knoten-Attributen A_S .

- **Knoten:** Ein Knoten repräsentiert einen durch einen Nutzer bereitgestellten Inhalt. Jeder Inhalt ist (unabhängig von seiner Verwendung in der Präsentation) genau einmal im semantischen Graph enthalten. Hierfür wird die Knotenmenge automatisch aktualisiert, wenn neue Inhalte verfügbar werden.
- **Kanten:** Kanten zwischen den Knoten modellieren explizite Beziehungen zwischen den Inhalten. Sie werden dazu genutzt, um eine feste Strukturierung der Inhalte zu beschreiben und so Themen beispielsweise hierarchisch zu gliedern oder wichtige, nutzerdefinierte Beziehungen zwischen zwei Inhalten zu beschreiben.

Die Knoten und Kanten geben an, *welche* Inhalte beschrieben werden, bzw. welche Inhalte explizit miteinander verbunden sind. Es wird jedoch noch eine Modellierung benötigt, welche die Eigenschaften der Inhalte charakterisiert. Hierfür werden die Attribute verwendet.

- **Attribute:** Jedem Knoten wird eine Menge von Attribut-Wert-Paaren zugewiesen $A_S : V_S \rightarrow Pot(N \times W)$, wobei N die Menge der Attributnamen, W die Werte dieser Attribute und Pot die Potenzmenge bezeichnet. Über die Attribute können Meta-Daten über die Inhalte (z.B. Auflösung eines Bildes, Größe oder Inhaltsquelle), semantische Eigenschaften (z.B. extrahierte Tags) und die Rolle eines Inhalts bzgl. des Themengebiets (z.B. Übersichtsfolie oder Detailerläuterung) angegeben werden.

Durch das Beschreiben der Inhaltseigenschaften über Attributmengen sind die Beschreibungen der Knoteneigenschaften voneinander unabhängig und können (individuell) beliebig erweitert werden (vgl. Anforderung B1).

Zur Vereinigung von unterschiedlichen Beschreibungen (Anforderung B2), werden die Knoten-, Kanten- und Attributmengen von zwei Semantischen Graphen vereinigt. Im Fall der Kanten können auf diese Weise verschiedene nutzerdefinierte Strukturen miteinander kombiniert und im Fall der Attribute die Inhaltsbeschreibung von mehreren Personen zusammengeführt werden.

Analog hierzu lassen sich auch manuell neue Beziehungen und Inhaltseigenschaften durch Ändern der Attributmenge und der Kantenverbindungen erstellen (Anforderung B3). Ein speziell hierfür entworfenes Nutzerinterface und neuartige Ansätze zum visuellen Editieren von Knotenattributen werden in den Abschnitte 4.1 und Abschnitt 5.2.1 vorgestellt.

Um zu einem gegebenen Inhalt automatisch semantisch verwandte Inhalte ermitteln zu können (Anforderung B4), ist noch eine Berechnungsvorschrift notwendig, die einerseits die Struktur (d.h. basierend auf den Kantenverbindungen) und andererseits die Inhaltseigenschaften (und somit die Attribute) berücksichtigt. Hierfür werden folgende Ansätze genutzt:

- **Struktur-bezogen:** Anhand der Graphstruktur werden Inhalte als verwandt angesehen, wenn sie sich in k -Nachbarschaft befinden. Ein Knoten befindet sich in k -Nachbarschaft zu einem anderen Knoten, wenn der kürzeste Pfad im Graphen über maximal k Kanten führt. Der Wert von k kann vom Nutzer bestimmt werden, um die Definition der semantischen Nachbarschaft einzuschränken oder zu erweitern.
- **Attribut-bezogen:** Ein Abgleich der Knotenattribute wird vorgenommen, bzgl. dem Inhalte ähnlich zueinander sind, wenn ihre Attributwerte wahlweise in irgendeinem, in mehreren oder in einem speziellen vom Nutzer (über den Attributnamen) bestimmten Attribut übereinstimmen.

3. Modellentwicklung

Die Struktur-basierte Definition erlaubt es, in einem semantischen Netz mit heterogenen Inhalten Verbindungen zwischen verschiedenen Arten von Inhalten (z.B. zwischen einem Text und dem Bild einer Karte) herzustellen. Durch Auswerten der Kantenverbindungen kann hierbei insbesondere auf das Hintergrundwissen der Nutzer über die eigenen Inhalte und über deren Organisation herangezogen werden.

Die Attribut-basierte Definition ermöglicht es dagegen, zu einem gegebenen Inhalt insbesondere andere Inhalte mit einem ähnlichen Typ zu ermitteln. Außerdem ist diese Art der Definition dazu geeignet, um Verbindungen zwischen den (teilweise separaten) Wissensbeständen von unterschiedlichen Nutzern automatisch (d.h. ohne manuelles Hinzufügen von Kanten) herzustellen.

Durch die Freiheiten bei der Anzahl der übereinstimmenden Attribute und bei der maximalen Pfadlänge, ist es den Nutzern möglich zu bestimmen, wie strikt hierbei die Definition von semantischer Ähnlichkeit ausgelegt werden soll.

3.2.3. Ergebnisse

Die entwickelte Graphbeschreibung erlaubt es Nutzern, Informationen über bereitgestellte Inhalte miteinander zu teilen und Inhalte themenbasiert miteinander zu verknüpfen. Das gewählte Modell ist hierbei flexibel genug, um unterschiedliche Arten der Beschreibungen aufzunehmen, zu kombinieren und somit zum semantischen Ordnen der Inhaltsmenge zu verwenden.

Basierend auf diesen Informationen werden, durch Auswertung der Inhaltseigenschaften und der explizit modellierte Verbindungen, Zusammenhänge zwischen Inhalten automatisch ermittelt.

Für die Darstellung von Inhalten auf den Displays werden diese Verknüpfungen genutzt, um auch die zu den Inhalten generierten Views semantisch zu verknüpfen oder die Informationsdarstellung durch thematisch passende Views zu ergänzen. Auf diese Weise wird Anforderung C3 für Display Ecologies adressiert (vgl. Abschnitt 2.2).

Die durch den Semantischen Graphen modellierten Beziehungen und Eigenschaften werden zum einen dazu genutzt, um Nutzern einen geordneten Überblick über die Menge der verfügbaren Inhalte zu geben (vgl. Abschnitt 5.2.1) und zum anderen, um ggf. auch automatisch passende Inhalte bei einer Diskussion und Präsentation empfehlen zu können (vgl. Abschnitt 5.2.2).

Der Präsentationsgraph erlaubt es, eine flexibel anpassbare Anzeige von Inhalten zu beschreiben und der Semantische Graph ermöglicht es, hierfür weitere thematisch passende Inhalte bereitzustellen. Um auch die Anpassungen selbst zu protokollieren, wird noch ein drittes Modell benötigt, das verschiedene Konfigurationen der Informationsanzeige beschreibt und diese schnell abrufbar macht.

3.3. Interaktionsgraph

Das Kommunizieren der Herkunft (engl. provenance) von Informationen ist ein wichtiges Mittel, um die Ergebnisse von Visualisierungs- und Analyse-Prozessen zu erklären und somit für den Nutzer nachvollziehbar und reproduzierbar zu machen. In einem aktuellen Übersichtsartikel geben Ragan et al. eine umfassende Zusammenfassung über die verschiedenen Arten von Provenance, den Zweck ihrer Verwendung und die hierfür genutzten Mittel [Rag+16].

Um eine assistierte Diskussion zu ermöglichen, wird in der vorliegenden Dissertation Provenance insbesondere in Hinsicht auf den Arbeitsablauf zum (spontanen) Erstellen und Anpassen der Informationsanzeige betrachtet. Bezüglich der von Ragan et al. klassifizierten Einsatzzwecke von Provenance sollen hierfür die folgenden Aktivitäten unterstützt werden:

- *Replicate*: Während einer Diskussion soll es möglich sein, zuvor angezeigte Informationen erneut abzurufen und ggf. als Ausgangspunkt für neue, alternative Argumentationslinien zu nutzen.
- *Action Recovery*: Einzelne ungewollte oder unpassende Änderungen in der Informationsanzeige sollen (durch einen Undo/Redo-Mechanismus) einfach zurückgenommen, aber auch wiederhergestellt werden können.
- *Collaborative Communication/Presentation*: Die Nutzer sollen während und nach einer Diskussion dazu in die Lage versetzt werden, sich einen Überblick vom Ablauf der Diskussion zu verschaffen. Hierzu sollen jeweils die einzelnen (durch Interaktion vorgenommenen) Anpassungen der Informationsanzeige kommuniziert werden.

Zur Unterstützung dieser Aktivitäten werden typischerweise Interaktionshistorien aufgezeichnet, welche als Grundlage für Visualisierungen und Undo/Redo-Mechanismen dienen [Rag+16]. Die konkrete Modellierung dieser Historien orientiert sich hierbei an den Anforderungen des jeweiligen Anwendungshintergrunds.

3.3.1. Problembeschreibung

Die in dieser Dissertation betrachteten Diskussionsszenarien gehen davon aus, dass die Informationsanzeige durch mehrere Nutzer gestaltet wird indem mehrere verschiedene Inhalte gleichzeitig auf multiplen Displays angezeigt werden. Eine Anpassung der Informationsanzeige (z.B. durch Hinzufügen oder Austauschen eines Inhalts) bewirkt, dass (automatisch) die Zusammenstellung, Verteilung und Anordnung der Views auf den Displays aktualisiert wird. Der Grund für eine Aktualisierung ist hierbei nicht immer sofort für jeden Nutzer ersichtlich. Dies kann insbesondere der Fall sein, wenn mehrere Displays betroffen sind, unterschiedliche Stellen im Präsentationsverlauf geändert

3. Modellentwicklung

werden oder Anpassungen durch verschiedene Nutzer mithilfe der persönlichen Geräte vorgenommen werden.

Eine Interaktionshistorie, welche über ihre Visualisierung den Ursprung und die Art dieser Aktualisierungen vermittelt und als Grundlage für einen Undo/Redo-Mechanismus dienen soll, muss in Hinsicht auf folgende Anforderungen entworfen werden:

D1: Eine einzelne Interaktion muss durch die Modellierung bezüglich dreier Aspekte in den Diskussionsverlauf einzuordnen sein. Ein *personeller* Bezug ist notwendig, um den ausführenden Nutzer identifizieren zu können, über den *display-bezogenen* Aspekt muss das betroffene Display bestimmbar sein und der von einer Interaktion betroffene Inhalt muss über einen *objekt-bezogenen* Aspekt der Modellierung ermittelt werden können.

Erfüllt eine Modellierung der Interaktionshistorie diese Anforderung, kann bereits eine der drei verfolgten Aktivitäten (*Collaborative Communication/Presentation*) über eine Visualisierung der Historie adressiert werden.

D2: Für einen Undo/Redo-Mechanismus muss es, basierend aus der Beschreibung der Historie, zudem möglich sein, die Änderungen der Informationsanzeige umzukehren bzw. wiederherzustellen. Bezüglich der verfolgten Aktivitäten *Replicate* und *Action Recovery* kann diese Anforderung weiter unterteilt werden:

D2.1 Ein sequentielles Undo/Redo muss unterstützt werden, um nacheinander die zuletzt vorgenommenen Änderungen rückgängig zu machen und somit zu früheren Konfigurationszuständen der Informationsanzeige wechseln zu können (vgl. Aktivität *Replicate*).

D2.2 Ein selektives Undo/Redo ist notwendig, um nicht nur die zuletzt ausgeführten, sondern auch einzelne Aktionen innerhalb der Historie rückgängig machen zu können (vgl. Aktivität *Action Recovery*). Ein selektives Undo/Redo kann auf diese Weise auch zu Konfigurationszuständen führen, die im vorherigen Verlauf der Diskussion noch nicht aufgerufen wurden.

Bisherige Ansätze, welche Interaktionsaufzeichnungen visualisieren, adressieren häufig zwei der drei Aspekte aus Anforderung D1 [MP12; VMP11; BLI08; Hee+08]. Beispielsweise decken Martins et al. mit ihrem Interaktionsmodell den personellen und den Objektbezug ab, konzentriert sich aber auf Single-Display-Umgebungen, weshalb der Display-Bezug nicht in der Modellierung berücksichtigt wird [MP12]. Die in den bestehenden Arbeiten verfolgten Ansätze könnten jedoch meist relativ einfach um den jeweils fehlenden Aspekt erweitert werden.

Für den vorliegenden Fall besteht dagegen die besondere Herausforderung für die Interaktionsmodellierung in der Kombination der Anforderungen D1 und D2. Bisherige Ansätze, welche Undo/Redo betrachten, adressieren hierfür bereits personelles [Che+09;

YGW04; RG99], display-bezogenes¹ [Sei+12; PK94] oder objektbasiertes [Che+12b; ZI97] Undo/Redo. Für Umgebungen mit großen Displays haben Seifried et al. ein Undo/Redo entwickelt, das personelle und display-bezogene Aspekte kombiniert. Hier können Änderungen sequentiell für einzelne Anzeigebereiche oder bezüglich des Nutzers rückgängig gemacht werden. Auf diese Weise wird bereits (mit Einschränkungen auf 2 der 3 Aspekte) Anforderung D2.1 behandelt. Arbeiten zum selektiven Undo/Redo (vgl. Anforderung D2.2) konzentrieren sich dagegen eher auf das Editieren von Dokumenten und finden i.d.R. keine Anwendung in Multi-Display-Umgebungen [YM15; Sun00; Che+12b; YGW04].

Zusammenfassend ist festzustellen, dass für die Gestaltung der Informationsanzeige, kein existierender Ansatz bekannt ist, der ein sequentielles und selektives Undo/Redo (D2) separat für alle drei Aspekten aus D1 ermöglicht.

Ein Hauptgrund hierfür ist, dass ein solcher Undo/Redo-Mechanismus zu neuen (vorher nicht erstellten) Zuständen der Informationsanzeige führen kann, die aufgrund mangelnder Assistenz oft nicht automatisch beschrieben, ermittelt und angezeigt werden können.

Da in der vorliegenden Arbeit diese Problemstellung durch den entwickelten Assistenzansatz zur automatischen Verteilung und Anordnung von Views in einer Multi-Display-Umgebung gelöst wird, wurde eine hierauf basierende Modellierung der Interaktionshistorie entworfen.

3.3.2. Modellbeschreibung

Als Grundlage für die automatische Informationsanzeige dient der in Abschnitt 3.1.2 beschriebene Präsentationsgraph. Einzelne **Interaktionen**, welche die Informationsanzeige bearbeiten, lassen sich somit auf Änderungen am Präsentationsgraph zurückführen.

Die **Zustände** der Informationsanzeige sind dementsprechend unterschiedliche Versionen des Präsentationsgraphen (die entstehen wenn der Graph durch Interaktion verändert wird).

Um diesen Zusammenhang zu modellieren, wird der sogenannte *Interaktionsgraph* eingeführt. Der Graph $G_I = (V_I, E_I)$ besteht aus den Knoten V_I und den Kanten E_I .

- **Knoten:** Ein Knoten stellt eine durch Nutzerinteraktion hervorgerufene Änderung des Präsentationsgraphen dar. Modelliert wird diese Interaktion als ein Tupel $(u, d, c, a) \in V_I$, welches den ausführenden Nutzer u , das betroffene Display d und den Inhalt c referenziert. Auf diese Weise lassen sich aller Interaktionen nach den drei Aspekten aus Anforderung D1 einordnen.

Der Wert a beschreibt die Art der Änderung am Präsentationsgraph. Gemäß der Definition des Präsentationsgraphen (vgl. Abschnitt 3.1.2) wird hierbei das Hin-

¹Ein Undo/Redo, welches den Display-Bezug berücksichtigt, wird in den referenzierten Arbeiten als “regional undo/redo” bezeichnet.

3. Modellentwicklung

zufügen und Löschen von Knoten oder Kanten, das Einordnen von Knoten in andere Zeitschritte oder das Ändern von Relevanzwerten vermerkt.

Die Menge der Knoten des Interaktionsgraphen beinhaltet alle während einer Diskussion aufgezeichneten Interaktionen. Außerdem wird ein besonderer Wurzelknoten eingefügt, der den Startzustand des Präsentationsgraphen bei Beginn der Interaktionsaufzeichnung beinhaltet.

Um aus den aufgezeichneten Interaktionen einen bestimmten Konfigurationszustand des Präsentationsgraphen ableiten zu können, muss die Interaktionsreihenfolge modelliert werden. Hierzu dienen die Kanten des Graphen.

- **Kanten:** Eine Kante $(v_1, v_2) \in V_I \times V_I$ verbindet zwei Knoten, wenn die Interaktion v_2 nach der Interaktion v_1 angewendet wurde.

Basierend auf den Kantenverbindungen können so die Zustände der Informationsanzeige abgeleitet werden. Hierbei definiert ein Pfad im Interaktionsgraphen, der beim Wurzelknoten beginnt und an einem beliebigen Knoten endet, einen Konfigurationszustand der Anzeige. Dieser Konfigurationszustand entspricht demjenigen Präsentationsgraph, der entsteht, wenn man (entlang dieses Pfads im Interaktionsgraphen) alle in den Pfadknoten $v_p = (u, d, c, a)$ enthaltenden Änderungen a auf den Präsentationsgraphen des Wurzelknotens anwendet.

Die Version des Präsentationsgraphen, die aktuell dazu verwendet wird, um die Informationsanzeige in einer Multi-Display-Umgebung zu steuern, ist ein besonderer Pfad im Interaktionsgraphen. Dieser Pfad wird in den folgenden Teilen dieser Arbeit als **aktiver Pfad** bezeichnet.

In einem Undo-Schritt werden Knoten aus dem aktiven Pfad entfernt. Dadurch hat die in dem entfernten Knoten enthaltende Änderung keinen Einfluss mehr auf die aktuell angezeigte Informationsdarstellung. Um für einen Redo-Schritt die Knoten wieder in den aktiven Pfad aufnehmen zu können, werden die entfernten Knoten nicht komplett aus dem Graphen gelöscht. Stattdessen werden sie in (nicht aktive) Seitenpfade ausgelagert.

3.3.3. Ergebnisse

Die gewählte Modellierung des Interaktionsgraphen erlaubt unterschiedliche Zustände der Informationsanzeige zu beschreiben. Hierzu werden verschiedene Versionen des Präsentationsgraphen in Beziehung zueinander gesetzt. Die zur Anzeige der Informationen verwendete Version des Präsentationsgraphen kann anhand des Interaktionsgraphen in andere Versionen des Präsentationsgraphen überführt werden, um einzelne aufgezeichnete Interaktionen rückgängig zu machen oder wiederherzustellen.

Jeder (bei der Wurzel beginnende) Pfad im Interaktionsgraph beschreibt hierbei eine bestimmte Version des Präsentationsgraphen und jeder Präsentationsgraph definiert wiederum eine gültige Verwendung der Inhalte für die Informationsdarstellung. Hierdurch

3.3. Interaktionsgraph

kann der aktueller Zustand der Informationsanzeige relativ frei über einen Undo/Redo-Mechanismus geändert werden, weil einzelne Interaktionen an (fast) beliebigen Stellen des aktiven Pfads eingefügt oder aus ihm entfernt werden können.

Der gewählte Modellierungsansatz für den Interaktionsgraph erweitert auf diese Weise bisherige Ansätze, indem er es erlaubt, neue (anzeigbare) Zustände zu generieren, die in der bisherigen Diskussion noch nicht (manuell) von Nutzern erstellt wurden.

Insbesondere wird hierdurch ein selektives Undo/Redo möglich (vgl. Anforderung D2.2) und (als Sonderfall eines solchen selektiven Undo/Redo) ein separat nach Displays, Personen und Inhalten differenzierendes sequentielles Undo/Redo unterstützt (Anforderungskombination D1 und D2.1).

Bezogen auf die zu unterstützenden Aktivitäten bedeutet das, dass ein *Replicate* nicht immer die gesamte Informationsanzeige betreffen muss. Stattdessen kann beispielsweise die Informationsdarstellung auf einem einzelnen, ausgewählten Display zurückgesetzt werden, während gleichzeitig die Zustände aller anderen Displays beibehalten werden. Für ein *Action recovery* können auf diese Weise nicht nur die letzte oder eine begrenzte Auswahl von Aktionen, sondern beliebige Änderungen zurückgenommen oder wiederhergestellt werden. In Kombination mit dem personellen Bezug können so beispielsweise alle Änderungen einer einzelnen Person zurückgenommen und durch Anpassungen einer anderen Person ersetzt werden.

Des Weiteren legt der Interaktionsgraph die Grundlage für eine, nach Displays, Inhalten und Nutzern differenzierende interaktive Visualisierung. Auf diese Weise wird auch die dritte Aktivität (*Collaborative Communication/ Presentation*) adressiert.

Die Visualisierung des Interaktionsgraphen und der entwickelte Undo/Redo-Mechanismus ermöglichen eine assistierte Interaktion. Die Ansätze hierfür und insbesondere die für ein personelles, display- und objekt-bezogenes Undo/Redo notwendigen Umformungen des Interaktionsgraphen, werden genauer in Abschnitt 5.1 behandelt.

Zunächst wird jedoch auf die entwickelten Ansätze für die assistierte Präsentation eingegangen.

4. Assistierte Präsentation

In diesem Kapitel werden die entwickelten Konzepte zur assistierten Präsentation vorgestellt. Der Ansatz sieht vor, dass die Informationsanzeige in einer Multi-Display-Umgebung interaktiv durch die Nutzer gestaltet wird und durch eine automatische Assistenz auf den Displays angezeigt wird.

In Abschnitt 4.1 wird zunächst auf das hierfür entwickelte grafische Interface eingegangen. Dieses wird auf den persönlichen Geräten der Nutzer aufgerufen und dient als individuelle Interaktionsschnittstelle, die es jedem Nutzer erlaubt, die Informationsanzeige im Raum zu konfigurieren. Hierfür ermöglicht das Interface, Inhalte von verschiedenen Quellen für eine Präsentation bereitzustellen. Es bietet weiterhin Zugang zu den im vorangegangenen Kapitel vorgestellten Modellen und erlaubt es somit, die Informationsanzeige zu kontrollieren. Insbesondere der nach den Nutzervorgaben modellierte Präsentationsgraph dient hierbei dazu, die Nutzervorgaben zur gewünschten Verteilung und Anordnung von Views auf die multiplen Displays zu beschreiben.

In einem nächsten Schritt werden diese Vorgaben dazu genutzt, um einen automatischen Layout-Algorithmus zu parametrisieren. Dieser Algorithmus berechnet automatisch eine passende View-Verteilung und -Anordnung auf den Displays und unterstützt damit den Nutzer beim Anzeigen der Präsentation in einer Multi-Display-Umgebung. Der Ansatz wird in Abschnitt 4.2 vorgestellt.

Um die Informationsanzeige bei Bedarf durch weitere und insbesondere interaktiv anpassbare Views ergänzen zu können, wurde ein Ansatz zur Integration von spontan generierten Views entwickelt. Dieser erlaubt es, interaktive Visualisierungs-Tools direkt mit der Präsentation zu verknüpfen und die durch diese Tools erzeugten Views automatisch in die bestehende Informationsanzeige einzubinden. Dieser Ansatz wurde für verschiedene Anwendungshintergründe umgesetzt und wird in Abschnitt 4.3 vorgestellt.

4.1. Nutzer-Interface zur Beschreibung der Präsentation

Das Interface erlaubt einen Zugriff auf die in Kapitel 3 eingeführten Modelle. Es wird auf den persönlichen Geräten der Nutzer angezeigt und ermöglicht es den Nutzern, die Informationsanzeige interaktiv zu gestalten. Hierfür sind drei grundlegenden Aufgaben zu unterstützen:

T1: Das **Zusammentragen von Inhalten**, welche für die Gestaltung der Informationsanzeige bereitstehen sollen, muss ermöglicht werden. Hierfür ist es erforder-

4. Assistierte Präsentation

lich, dass der einzelne Nutzer (lokal gespeicherte) Inhalte in die Multi-Display-Umgebung einbringen kann.

Damit anschließend auch andere Nutzer diese freigegebenen Inhalte für das Erstellen einer Präsentation verwenden können, müssen die bereitgestellten Inhalte gesammelt und für alle Nutzer verfügbar gemacht werden.

T2: Für die **Gestaltung der Informationsanzeige** muss die Verwendung der zusammengetragenen Inhalte und die Anzeige der entsprechenden Views auf den Displays der Multi-Display-Umgebung beschrieben werden. Die hierfür entworfene Modellierungsgrundlage bietet der Präsentationsgraph (vgl. Abschnitt 3.1). Entsprechend müssen die Knoten, die Kantenverbindungen und die Attribute des bestehenden Präsentationsgraphen über das Interface vermittelt und durch Interaktion mit dem Interface von den Nutzern einfach angepasst werden können.

T3: Das **Vortragen/Moderieren** der geplanten Präsentation bzw. der Diskussionsbeiträge durch die Nutzer muss unterstützt werden. Insbesondere muss hierbei von den Nutzern gesteuert werden können, welche Teile der erstellten Präsentation auf den Displays der Umgebung gleichzeitig bzw. nacheinander angezeigt werden und welche Inhalte ggf. zusätzlich eingeblendet werden sollen.

Im Folgenden wird zunächst der Aufbau des entworfenen Nutzer-Interfaces und die hierfür verwendete visuelle Repräsentation der Inhalte, des Präsentationsgraphen sowie des zeitlichen Verlaufs vorgestellt. Anschließend werden jeweils die interaktiv nutzbaren Funktionalitäten zum Bereitstellen von Inhalten, zum Editieren des Präsentationsgraphen und zur Anzeige der Präsentation beschrieben.

Abbildung 4.1 zeigt das entworfene Interface, mit einer durch mehrere Nutzer erstellten Präsentation. Der Aufbau des Interfaces orientiert sich an den drei zu unterstützenden Aufgaben und unterteilt das Interface in drei vertikal übereinander angeordnete Bereiche:

Content Pool: Der Content Pool befindet sich im oberen Teil des Interfaces und dient zum Sammeln der von mehreren Nutzern bereitgestellten Inhalte.

Neue Inhalte können zu dem Content Pool hinzugefügt werden indem ein Nutzer (lokal auf seinem persönlichen Gerät gespeicherte) Dateien per einfachem Drag & Drop in die Region des Content Pools verschiebt. Die Datei wird daraufhin automatisch ausgewertet, um die darstellbaren Inhalte in der Datei zu ermitteln. Es werden beispielsweise die einzelnen Folien einer Präsentation und die enthaltenden Bilder aus einem PDF extrahiert. Die so gewonnenen Inhalte werden anschließend automatisch an alle persönlichen Geräte verteilt und erscheinen in dem Content Pool der anderen Nutzer. Dort können die Inhalte nun von jedem Nutzer betrachtet werden.

Die Inhalte im Content Pool werden horizontal nebeneinander angeordnet und (zusätzlich zum Dateinamen) über kleine Vorschaubilder symbolisiert, um einzelne Inhalte schneller identifizieren zu können. Für eine genauere Betrachtung kann ein

4.1. Nutzer-Interface zur Beschreibung der Präsentation

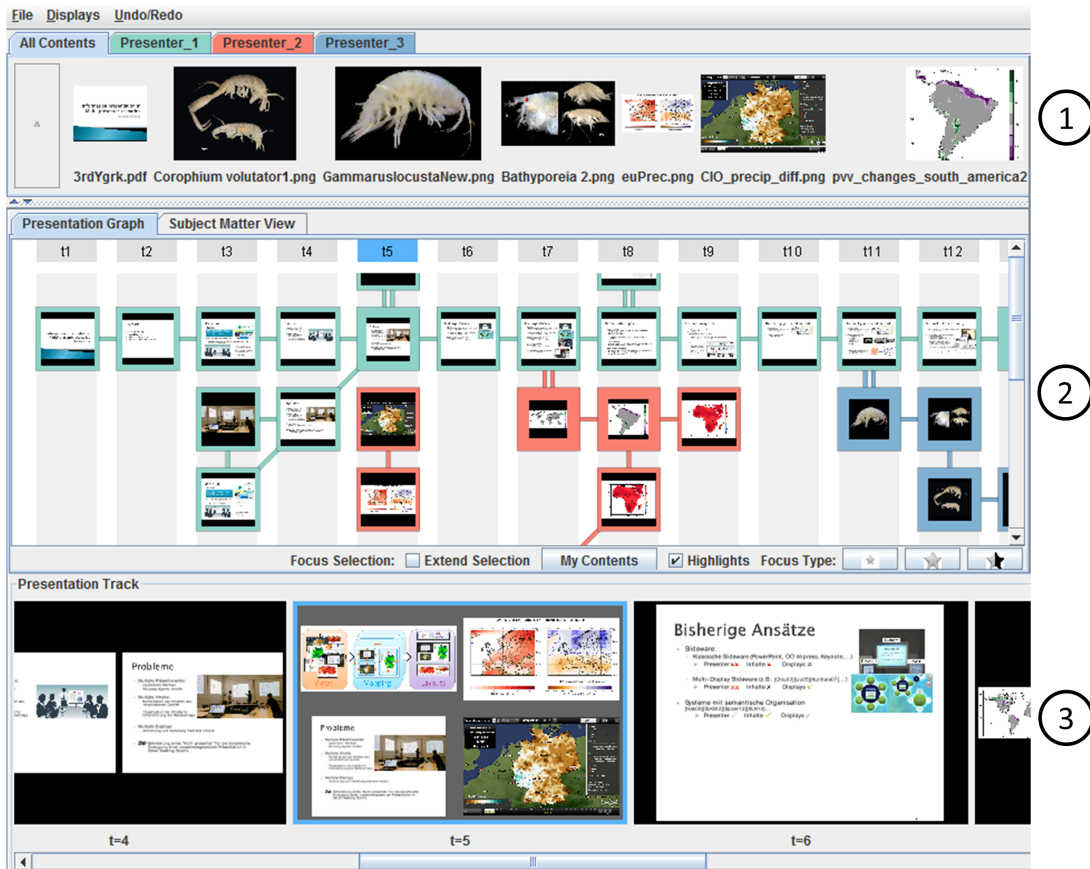


Abbildung 4.1.: Nutzer-Interface mit dem Content Pool (1) im oberen Teil, dem Präsentationsgraphen in der Mitte (2) und der Layer Bar im unteren Bereich (3). Der Content Pool enthält mehrere PDFs und PNG-Bilder, die durch 3 Nutzer bereitgestellt und in Tabs eingeordnet wurden. Der Präsentationsgraph beschreibt die angezeigten Inhalte und die Struktur der Präsentation. Die durch die Nutzer erstellten Teile der Präsentation werden hierbei rot, blau und grün gekennzeichnet. Die Layer Bar zeigt die Inhalte des zeitlichen Ausschnitts von t_4 bis t_6 . Der aktuell auf den Displays angezeigte Zeitschritt t_5 wird durch eine blaue Umrandung hervorgehoben.

Inhalt anschließend im Vollbildformat angezeigt werden.

Der Content Pool kann eine potenziell sehr große und heterogene Inhaltsmenge umfassen. Um einen Nutzer beim Verschaffen eines Überblicks und bei der Suche nach Inhalten zu unterstützen, gruppiert der Content Pool die Inhalte. Hierbei werden zwei Ansätze verfolgt:

4. Assistierte Präsentation

- **Nutzer-bezogen:** Die Inhalte werden in Tabs am oberen Rand des Content Pool gruppiert, wobei ein Tab jeweils nur die Inhalte eines bestimmten Nutzers enthält. Der Nutzernamen wird hierbei als Tab-Beschriftung verwendet. Zusätzlich existiert auch ein Tab, der alle gesammelten Inhalte beinhaltet.
- **Datei-bezogen:** Hierbei werden mehrere Inhalte, die aus der gleichen Datei stammen, zusammengefasst. Dies können beispielsweise mehrere Seiten eines PDFs oder mehrere Bilder von einer einzelnen Seite sein. Eine Gruppe wird, ähnlich wie andere Inhalte, über ein einzelnes Symbolbild im Content Pool dargestellt. Nutzer können die verschiedenen Inhaltsgruppen, ähnlich wie eine Ordnerstruktur, durchsuchen. Hierfür lassen sich die Inhalte einer Gruppe sichtbar machen (und anschließend wieder ausblenden), indem der Nutzer auf das Symbolbild einer Gruppe klickt.

Der Content Pool unterstützt somit das Zusammentragen von Inhalten und stellt zudem eine geordnete Übersicht von allen verfügbaren Inhalten bereit (vgl. Aufgabe T1).

Diese können nun dazu verwendet werden, um den Präsentationsgraph zu erstellen.

Darstellung des Präsentationsgraphen: In der Mitte des Interfaces wird der Präsentationsgraph als Node-Link-Diagramm angezeigt.

- Die Knoten des Präsentationsgraphen werden als Rechtecke mit kleinen Vorschaubildern der referenzierten Inhalte dargestellt.
Der Relevanzwert eines Knotens wird über die Größe des Vorschaubilds innerhalb der rechteckigen Knotenkontur vermittelt. Ist das Vorschaubild im Verhältnis zu den anderen Knoten groß, so ist auch der Relevanzwert dieses Knotens vergleichsweise groß (und der zu dem Knoten generierte View erhält mehr Platz auf den Displays des Raums).
Die Zeitschritte der Präsentation werden aufsteigend von links nach rechts auf der horizontalen Achse der Zeichnung abgebildet. Das bedeutet, dass Knoten, die zeitgleich auf den Displays angezeigt werden sollen, vertikal übereinander angeordnet werden. Eine Beschriftung der Zeitschritte wird am oberen Rand der Zeichnung angezeigt, um das Wiederfinden eines bestimmten Zeitschritts zu unterstützen.
- Die Kanten des Präsentationsgraphen werden als gerade Linien zwischen den Knoten eingezeichnet. Durch das horizontale Aneinanderreihen der Zeitschritte, verlaufen zeitliche Kanten somit in horizontaler und räumliche Kanten in vertikaler Richtung. Die gruppierenden Kanten werden als doppelte Linie dargestellt, um sie von den anderen Kanten unterscheidbar zu machen.

Das Editieren des Präsentationsgraphen wird durch Interaktion mit dem Node-Link-Diagramm ermöglicht (vgl. Aufgabe T2).

Um neue Inhalte in eine Präsentation aufzunehmen, können Nutzer die Inhalte

4.1. Nutzer-Interface zur Beschreibung der Präsentation

per Drag & Drop aus dem Content Pool in den Präsentationsgraph verschieben. Hierdurch wird ein neuer Knoten im Präsentationsgraph erstellt, der auf den jeweiligen Inhalt im Content Pool verweist. Wird anstelle eines einzelnen Inhalts eine Inhaltsgruppe in die Darstellung gezogen, werden automatisch Knoten für alle Inhalte in der Gruppe angelegt. Hierdurch ist es möglich, größere Präsentationsteile (z.B. aus einem anderen Vortrag) schnell in eine neue Präsentation zu übernehmen.

Die Zuordnung von einem Knoten zu einem Zeitschritt kann vom Nutzer geändert werden, indem der Knoten innerhalb der Darstellung des Präsentationsgraphen an eine neue (unbelegte) Position verschoben wird. Anhand der neuen horizontalen Position in der Darstellung wird der aktualisierte Zeitschritt ermittelt und im Modell des Präsentationsgraphen für den Knoten geändert.

Die Kantenverbindungen zwischen den Knoten können auf ähnliche Weise erstellt werden. Hierfür wird ein ausgewählter Knoten (statt auf eine freie Stelle) auf einen anderen Knoten in der Darstellung gezogen, um die Knoten miteinander zu verbinden. Durch Wiederholung dieser Geste kann die so eingefügte Kante wieder entfernt werden.

Die Relevanzwerte von angewählten Knoten lassen sich mit einer Scroll-Geste (z.B. per Mausrad) verändern.

Die Knotenpositionen in der Darstellung werden nach dem Verschieben automatisch so korrigiert, dass sich Knoten nicht überlappen und sich die Knoten immer in der Mitte des neuen Zeitschritts befinden.

Dies hat den Vorteil, dass Knoten aus verschiedenen Zeitpunkten in der Zeichnung immer klar räumlich voneinander getrennt sind und innerhalb eines Zeitschritts die Vorschauansichten der einzelnen Knoten, durch Vermeiden von Überschneidungen, erkennbar bleiben.

Weiterhin müssen hierdurch die verwendeten Interaktionsgesten weniger präzise ausgeführt werden (um bspw. Knoten per Drag & Drop zu platzieren) und es werden keine Tastatureingaben benötigt.

Auf diese Weise lässt sich das entwickelte Interface zum Editieren des Präsentationsgraphen nicht nur über Laptops und Computer mit Cursor-Interaktion bedienen, sondern kann auch von anderen Geräten mit Touch-Interaktion (wie z.B. Tablets) verwendet werden.

Um denjenigen Nutzer identifizieren zu können, der einen Knoten oder eine Kante erstellt hat, werden die Knotenumrandungen und die Kanten eingefärbt. Jeder Nutzer erhält hierbei automatisch eine eigene Farbe, die mit der Einfärbung des personenbezogenen Tabs im Content Pool übereinstimmt. Auf diese Weise können Nutzer abschätzen, an welcher Stelle im Vortragsverlauf, welcher Nutzer, welchen Beitrag zu einer geplanten Präsentation geleistet hat und diesen ggf. vortragen möchte.

4. Assistierte Präsentation

Der sichtbare Ausschnitt des Präsentationsgraphen lässt sich außerdem verschieben und in der Größe verändern. Hierdurch können sich Nutzer individuell einen Überblick vom Präsentationsgraphen verschaffen und einzelne Details können bei Bedarf durch reinzoomen sichtbar gemacht werden.

Bei Änderungen am Präsentationsgraph wird das Node-Link-Diagramm bei allen Nutzern automatisch aktualisiert. Der über das Interface angezeigte Ausschnitt des Präsentationsgraphen wird jedoch von jedem Nutzer individuell eingestellt. Dies erlaubt es verschiedenen Nutzern, gleichzeitig und unabhängig an unterschiedlichen Teilen einer Präsentation zu arbeiten.

Layer Bar: Die Layer Bar im unteren Teil des Interfaces ist als Moderationsansicht entworfen worden (vgl. Aufgabe T3). Sie unterstützt Nutzer beim Vortragen indem sie eine Zusammenfassung der aktuellen und der in den folgenden Zeitschritten angezeigten Inhalte gibt.

Hierzu werden Vorschauansichten der verwendeten Inhalte angezeigt und, nach Zeitschritten gruppiert, nebeneinander dargestellt. Im Vergleich zum Präsentationsgraph werden die Inhalte hierbei jedoch relativ groß (und ohne Kantenverbindungen) dargestellt, um einzelne Inhalte besser identifizieren (und ggf. Stichpunkte oder Überschriften lesen) zu können.

Die Darstellung der Layer-Bar zeigt einen Ausschnitt des zeitlichen Verlaufs, der für unterschiedliche Präsentationssituationen und individuelle Moderationsstile angepasst werden kann.

Hierzu kann die Anzahl der gleichzeitig dargestellten Zeitschritte, durch den Nutzer frei eingestellt werden. Dies erlaubt es beispielsweise, für Vorträge mit vielen Inhalten pro Zeitschritt, die Anzahl der gleichzeitig in der Layer Bar gezeigten Zeitschritte zu reduzieren und dafür die einzelnen Inhalte besser sichtbar zu machen. Dagegen können Nutzer, die eher weniger Inhalte gleichzeitig verwenden, einen größeren zeitlichen Ausschnitt aus dem Vortrag einblenden, indem sie die Anzahl der angezeigten Zeitschritte erhöhen.

Der in der Layer Bar gezeigte zeitliche Ausschnitt kann unterschiedliche Stellen der Präsentation zeigen. Für die Unterstützung eines Vortrags kann er so konfiguriert werden, dass er automatisch immer dem aktuell auf den Displays angezeigten Zeitschritt folgt. Alternativ können Nutzer den Ausschnitt auch frei verschieben, um sich einen Überblick über den zeitlichen Verlauf der geplanten Präsentation zu verschaffen.

Externe Tools zur interaktiven Generierung von neuen Views können außerdem per Drag & Drop über die Layer Bar mit der Präsentation verknüpft werden. Auf diese Weise wird das Tool mit allen Knoten, die diesen Inhalt verwenden, verlinkt (vgl. Abschnitt 4.3).

Die vorgestellten drei Elemente des Nutzer-Interfaces können vom Nutzer in der Größe verändert oder komplett ausgeblendet werden. Hierdurch kann das Interface für die

4.1. Nutzer-Interface zur Beschreibung der Präsentation

Aufgaben T1-T3 individuell angepasst werden.

In der Vorbereitungsphase einer Präsentation ist beispielsweise eher das Sammeln der Inhalte im Content Pool (T1) und das Erstellen des Präsentationsgraphen (T2) über das Node-Link-Diagramm wichtig. Entsprechend kann für solche Situationen die Layer Bar ausgeblendet werden, um den anderen beiden Komponenten mehr Platz im Interface zu bieten. Steht dann das Moderieren (T3) einer Präsentation im Vordergrund, kann es wiederum wünschenswert sein, die Layer Bar als Moderationsansicht zu nutzen. Um hierfür die Inhalte des aktuellen Zeitschritts in der Layer Bar größer darstellen zu können, kann in solche Situationen wiederum der Content Pool oder der Präsentationsgraph (temporär) ausgeblendet werden.

Der aktuell auf den Displays des Raums angezeigte Zeitschritt im Präsentationsgraph kann interaktiv über das Interface festgelegt werden. Er wird im Präsentationsgraph bzw. in der Layer Bar durch eine blaue Umrandung gekennzeichnet (vgl. Abbildung 4.1). Für das Moderieren (T3) einer Präsentation können die im Präsentationsgraph vorbereiteten Zeitschritte durch inkrementieren des angezeigten Zeitschritts nacheinander abgerufen werden.

Eine Diskussion bzw. das Beantworten von Fragen erfordert jedoch auch, schnell direkt zu unterschiedlichen Stellen der Präsentation springen zu können. Um dies zu ermöglichen, kann der Nutzer den angezeigten Ausschnitt der Layer Bar manuell verschieben und somit schnell alle verwendeten Inhalte überfliegen. Hat er den gesuchten Zeitschritt gefunden, kann er ihn durch einfaches Anklicken zum aktuellen Zeitschritt machen und damit auf den Displays anzeigen. Ein ähnliches Vorgehen wird auch von der Darstellung des Präsentationsgraphen unterstützt. Hier kann der Nutzer zeitliche Sprünge durch einen Doppelklick auf die Beschriftung der Zeitschritte (bzw. auf eine Stelle innerhalb des Präsentationsgraphen) ausführen.

Zusammenfassend erlaubt das entwickelte Interface mit Hilfe des Content Pools, des Präsentationsgraphen und der Layer Bar das Zusammentragen von Inhalten aus verschiedenen Quellen (T1), das Kombinieren dieser Inhalte in einer Präsentation (T2) und legt die Grundlage für das interaktive Moderieren der Präsentation (T3).

Das Besondere hierbei ist, dass die Nutzer zum Ausführen dieser Aufgaben das Interface auf ihrem persönlichen Gerät nutzen können. Auf diese Weise können mehrere Nutzer die Inhaltssammlung individuell ergänzen, Vorträge gemeinsam interaktiv erstellen oder sich während des Moderierens eines Vortrags einfach abwechseln, ohne hierfür ein sonst übliches Präsentationsgerät übergeben zu müssen.

Um die interaktiv gestaltete Präsentationen auch auf den Displays der Multi-Display-Umgebung anzeigen zu können, ist hierfür ein Assistenzansatz entwickelt worden. Dieser wird im nächsten Abschnitt vorgestellt.

4.2. Layout-Berechnung zur Anzeige der Präsentation

Das entwickelte Interface erlaubt es den Nutzern, die Modelle zur Beschreibung einer Präsentation interaktiv zu erstellen. Der nächste Schritt besteht nun darin, diese Präsentation auf den Displays des Smart Meeting Rooms automatisch anzuzeigen. Hierfür müssen zwei Aufgaben ausgeführt werden. Erstens müssen die zu den Inhalten generierten Views auf die vorhandenen Displays verteilt werden und zweitens müssen die Views auf ihrem jeweiligen Displays angeordnet werden, um die genaue Position und die Größe der Views zu bestimmen.

Für diese grundlegenden Aufgaben ergeben sich allgemeine Anforderungen durch die Nutzung der Umgebung als Display Ecology (vgl. Abschnitt 2.2):

- Die Anzahl der anzuzeigenden Views verändert sich spontan durch das Hinzufügen oder Entfernen von Inhalten.
- Die Größe der zur Verfügung stehenden Anzeigefläche kann, durch das Hinzufügen oder Entfernen von Displays variieren.
- Die Verteilung und Anordnung von Views auf dieser Anzeigefläche wird durch die Nutzer angepasst.

Aus diesen Anforderungen folgt, dass die Verteilung und Anordnung der Views nicht statisch ist, sondern laufend dynamisch verändert werden muss, um auf wechselnde Inhalte, Änderungen in der Display-Zusammenstellung und auf die aktuellen Nutzervorgaben reagieren zu können.

Konkrete Anforderungen ergeben sich diesbezüglich durch die Nutzung der Umgebung für Präsentationen und Diskussionen. Hierbei müssen die durch den Präsentationsgraph modellierten Nutzervorgaben berücksichtigt werden. Konkret sind dies:

- L1:** Durch die Nutzer priorisierte Views müssen bei der Anzeige bevorzugt werden.
- L2:** Miteinander räumlich oder semantisch verbundene Views müssen in räumlicher Nähe zueinander platziert werden.
- L3:** Die modellierte zeitliche Ersetzung von Views muss berücksichtigt werden.

Bisherige Ansätze für die automatische Anzeige von Views in Multi-Display-Umgebungen adressieren bereits einige der allg. Anforderungen (vgl. Abschnitt 2.1.4). Beispielsweise erlaubt der Ansatz von Heider, automatisch mehrere Views auf die verfügbaren Displays zu verteilen und berücksichtigt hierbei die individuelle Eignung der einzelnen Displays. Die Arbeiten von Radloff erweitern diesen grundlegenden Ansatz, indem auch eine initiale View-Anordnung pro Displays automatisch berechnet wird, der Sichtbereich von Nutzern bei der Verteilung berücksichtigt wird und erste Ansätze für die räumliche Gruppierung von Views entwickelt wurden.

4.2. Layout-Berechnung zur Anzeige der Präsentation

Bezüglich der allgemeinen Anforderung, dass die Anzeige auch dynamischen anpassbar sein muss, konzentrieren sich die Ansätze hauptsächlich auf das Unterstützen von wechselnden Display-Konfigurationen und Nutzerpositionen. Nutzervorgaben zur Gestaltung der Anzeige (und insbesondere die Anforderungen L1-L3), müssen dagegen weitestgehend noch durch manuelles Verschieben und Skalieren von Views umgesetzt werden.

Im Rahmen dieser Dissertation wurde deshalb ein neuer Ansatz entwickelt, welcher die bestehenden Konzepte zur automatischen View-Verteilung und Anordnung erweitert und hierbei die allgemeinen Anforderungen an Display Ecologies auch umfassend für die nutzergesteuerte Anpassung umsetzt.

Der prinzipielle Lösungsansatz basiert hierbei auf einer ähnlichen Vorgehensweise wie bei den Arbeiten von Heider und Radloff:

- Eine **Qualitäts-Funktion** wird verwendet, um unterschiedliche View-Verteilungen und View-Anordnungen zu bewerten.
- Basierend auf dieser Funktion wird der Lösungsraum (systematisch) durchsucht. Jede Lösung beschreibt hierbei eine mögliche View-Verteilung und die Anordnung der Views. Sie besteht demzufolge aus zwei Komponenten, die in Abbildung 4.2 nochmal veranschaulicht werden:

View-Display-Zuordnung: Eine View-Display-Zuordnung ist eine Funktion, welche die Menge der anzuzeigenden Views auf die Menge der Displays abbildet und dabei für jede View angibt, auf welchem Display die View erscheinen soll. Diese Abbildung definiert die Verteilung aller Views auf die Displays.

Layouts: Ein Layout bestimmt die genaue Position und Größe der Views auf einem Display. Für jedes Display wird ein Layout erstellt, um die Anordnung aller Views zu bestimmen.

Durch Ermitteln der Lösung mit der höchsten Qualität, wird diejenige View-Verteilung und Anordnung bestimmt, die den Vorgaben der Nutzer am besten entspricht.

Im Rahmen dieser Arbeit wurde die bestehende Qualitätsfunktion umfassend überarbeitet, um die existierenden Teilaspekte bei der Bewertung (z.B. die Bewertung der Sichtbarkeit von Views oder die Eignung von ausgewählten Anzeigegeräten) weiterverwenden zu können, zusätzlich aber auch die interaktiv erstellten Nutzervorgaben (L1-L3) bei der Bewertung zu berücksichtigen.

Darüber hinaus wurde ein neuer Algorithmus entworfen, der die am besten bewertete Zuordnung und die Layouts zu interaktiven Frameraten berechnen kann. Dies ermöglicht es, die View-Verteilung und die Layouts laufend automatisch zu aktualisieren und auf diese Weise ein interaktives, nutzergetriebenes Anpassen der Inhaltsanzeige für Präsentationen und Diskussionen zu unterstützen.

4. Assistierte Präsentation

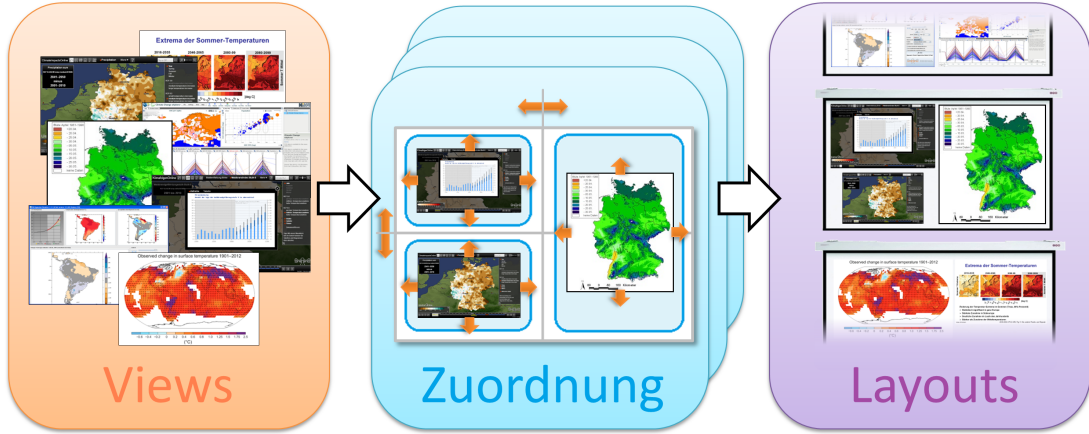


Abbildung 4.2.: Die Abbildung zeigt eine mögliche Verteilung und Anordnung von mehreren Views auf drei Displays. Durch eine Zuordnung von Views auf Displays wird bestimmt, welche Views auf welchen Displays angezeigt werden sollen. Anschließend werden die Views auf jedem Displays skaliert und positioniert, um die Layouts zu erzeugen. Die am besten bewertete Kombination aus Zuordnung und Layouts wird dazu verwendet, die Views im Smart Meeting Room anzuzeigen.

Die entwickelte Qualitätsfunktion bewertet eine Lösung basierend auf den erzeugten Layouts. Weil die View-Display-Zuordnung vorgibt, welche Views zusammen auf einem Display angezeigt werden und somit in demselben Layout erscheinen, wird hierdurch auch diese Verteilung der Views mitbewertet. Die Qualitätsfunktion hat die folgende Form:

$$Max : Q = \alpha \cdot Q_s + \beta \cdot Q_t + \gamma \cdot Q_d \quad (4.1)$$

Die zu maximierende Funktion Q ist eine gewichtete Summe aus der Anzeigequalität Q_d , der zeitlichen Qualität Q_t und der räumlichen Qualität Q_s , wobei α , β und γ die Gewichte sind.

- Die Anzeigequalität Q_d quantifiziert, wie gut die Views des aktuell angezeigten Zeitschritts V_i für die Nutzer sichtbar sind.

$$Q_d = \sum_{v \in V_i} d(v) \cdot \left(-r_v \cdot \left(\frac{s_v}{s_d} \right)^2 + 2r_v \cdot \left(\frac{s_v}{s_d} \right) \right) \quad (4.2)$$

Für eine einzelne View v bezeichnet hierbei s_v die Größe der View, r_v ihren Relevanzwert und s_d die Größe des Displays dem sie zugeordnet wurde.

4.2. Layout-Berechnung zur Anzeige der Präsentation

Für jedes Display, wird ein Sichtbarkeitsfaktor $d(v)$ verwendet, der bewertet, wie gut eine View auf seinem zugeordneten Display durch die Nutzer wahrnehmbar ist. Der Faktor berücksichtigt hierbei unter anderem die Display-Größe, der Display-Position im Raum und die Blickrichtung des Publikums. Die genaue Berechnung des Sichtbarkeitsfaktors bzgl. der Raumeigenschaften wird vereinfacht nach den von Radloff beschriebenen Formeln vorgenommen [Rad+11; Rad14] (siehe Anhang A).

Um auch die nutzerdefinierten Vorgaben zu berücksichtigen, fließen die Relevanzwerte r_v aus dem Präsentationsgraph in die Berechnung der Anzeigequalität mit ein. Hierbei wird angenommen, dass mit zunehmender View-Größe s_v auch die Sichtbarkeit von Views zunimmt. Bei mehreren Views auf demselben Display erhalten demzufolge diejenigen Views mit größeren Relevanzwerten mehr von dem verfügbaren Anzeigeraums s_d zugewiesen, als diejenigen Views mit einem geringeren Relevanzwert.

Durch das Verwenden einer quadratische Funktion für die Summanden von Q_d wird sichergestellt, dass beim Maximieren der Anzeigequalität der verfügbare Platz auf einem Display nicht ausschließlich für die Anzeige von Views mit hohen Relevanzwerten verwendet wird. Stattdessen wird hierdurch auch anderen Views (entsprechend ihren Relevanzwerten) eine bestimmte Mindestgröße eingeräumt. Durch höhere Bewertung von priorisierten Views, wird Anforderung L1 bei der Layout-Berechnung berücksichtigt.

- Die räumliche Qualität Q_s bewertet das Einhalten der Vorgaben zur räumlichen Nähe von Views (Anforderung L2). Diese Vorgaben basieren auf den räumlichen und gruppierenden Kanten $E_i \subseteq E_p$ des Präsentationsgraphen, welche die Knoten des aktuell angezeigten Zeitschritts V_i verbinden.

$$E_i = E_p \cap (V_i \times V_i)$$

$$Q_s = \sum_{(v_1, v_2) \in E_i} 1 - \frac{|p_{v_1} - p_{v_2}|}{s_d} \quad (4.3)$$

Für eine Kante (v_1, v_2) , welche die Views v_1 und v_2 verbindet, ist die Qualität hoch, wenn die Positionen p_{v_1} und p_{v_2} der Views in dem erzeugten Layout nahe beieinander liegen. Der Term s_d bezeichnet (analog zur Formel von Q_d) die Display-Größe und ist damit die maximal mögliche Entfernung für zwei Views in demselben Layout.

- Die zeitliche Qualität Q_t berücksichtigt die zeitlichen Kanten des Präsentationsgraphen. Hierbei werden diejenigen Kanten E_t verwendet, die Views aus dem aktuell angezeigten Zeitschritt V_i mit den Views aus dem zuvor gezeigten Zeitschritt V_{i-1} verbinden. Für die Kanten aus E_t werden die Einzelqualitäten nach

4. Assistierte Präsentation

der gleichen Formeln berechnet wie bei den räumlichen Kanten, um die Nutzervorgaben für die zeitliche Ersetzung von Views zu berücksichtigen (Anforderung L3). Darüber hinaus werden noch weitere, automatisch bestimmte Kanten E_q berücksichtigt. Der Einfluss einer Kante auf die zeitliche Qualität wird durch das Gewicht w bestimmt.

$$E_t = E_p \cap (V_i \times V_{i-1})$$

$$Q_t = \sum_{(v_1, v_2) \in E_t \cup E_q} \left(1 - \frac{|p_{v_1} - p_{v_2}|}{s_d} \right) \cdot w \quad (4.4)$$

Beim Anzeigen einer Präsentation kann es vorkommen, dass Zeitschritte im Präsentationsgraph von den Nutzern übersprungen werden. In solchen Fällen existieren meist nur wenige, durch die Nutzer manuell erstellte zeitliche Kanten zum zuvor angezeigten Zeitpunkt (d.h. E_t ist leer). Um trotzdem neue Layouts zu produzieren, die sich zeitlich an den vorangegangenen Layouts orientieren, werden in solchen Fällen Hilfskanten E_q automatisch berechnet und bei der Layout-Bewertung berücksichtigt.

Hierbei wird eine zusätzliche Kante (v_1, v_2) berücksichtigt, wenn der kürzeste Pfad im Präsentationsgraph zwischen den Knoten v_1 und v_2 eine bestimmte Anzahl an Zeitschritten nicht überschreitet. Soll beispielsweise laut den Kanten des Präsentationsgraphen die Darstellung v_1 durch v_x ersetzt werden und v_x wiederum durch v_2 ausgetauscht werden, so wird eine neue Kante (v_1, v_2) angenommen, die v_1 an eine ähnliche Stelle wie v_2 platziert. Auf diese Weise wird die räumliche Beziehung zwischen den Views transitiv über die Zeit fortgesetzt. Mit zunehmender Pfadlänge wird der Einfluss der zusätzlich eingefügten Kanten reduziert. Dies wird über den Faktor w in der Formel dargestellt. Dieser Faktor ist bei einer nutzerdefinierten direkten Verbindung 1 und für Hilfskanten sehr klein, um direkt modellierte Nutzervorgaben im Optimierungsprozess nicht zu überschreiben.

Die Nutzer können durch die Gewichte α , β und γ den Einfluss der zeitlichen, räumlichen und der Anzeige-Qualität auf die Layout-Bewertung festlegen. Auf diese Weise können Nutzer steuern, ob Layouts produziert werden sollen, die verwandte Inhalte eher näher zusammen darstellen (L2), ob Layouts sich lieber so wenig wie möglich verändern sollen, wenn neue Views eingefügt oder entfernt werden (L3), oder ob bei der Erstellung der Layouts die Priorisierung von Views stärker berücksichtigt werden soll (L1).

Das Ändern der Gewichte ist von den persönlichen Geräten der Nutzer aus über das Nutzerinterface möglich. Ähnlich wie Anpassungen am Präsentationsgraph, werden Änderungen an den Gewichten automatisch erkannt, bewirken eine Neubewertung der Layouts und sorgen somit für eine automatisches Aktualisieren der View-Verteilung und Anordnung. Auf diese Weise erhalten die Nutzer ein schnelles Feedback zu ihren Anpassungen.

4.2. Layout-Berechnung zur Anzeige der Präsentation

Für das Bestimmen der Layouts ist noch ein Berechnungsansatz notwendig, der eine Rückmeldung zu interaktiven Geschwindigkeiten ermöglicht.

Problematisch ist hierbei, dass der Lösungsraum zu groß ist, um alle Lösungen zu erzeugen und gemäß ihrer Qualität zu bewerten. Zudem stellt eine approximativ beste Lösung, wie sie beispielsweise von Radloff mittels eines genetischen Algorithmus ermittelt wird, nicht sicher, dass alle Nutzervorgaben durch die Layouts vollständig berücksichtigt werden.

Aus diesen Gründen wurde in dieser Dissertation ein Algorithmus entwickelt, der die global beste Lösung berechnet. Der Algorithmus deckt hierzu den gesamten Lösungsraum ab, ohne jedoch alle Lösungen zu erzeugen. Stattdessen werden nur diejenigen View-Display-Verteilungen und Layouts generiert, die Aussicht darauf haben, die beste Lösung zu sein. Auf diese Weise können die bezüglich der Qualitätsfunktion gesuchten Layouts ausreichend schnell erzeugt werden, um ein interaktives Anpassen der Nutzervorgaben durch eine automatische Inhaltsanzeige zu unterstützen.

Die Vorgehensweise des entworfenen Algorithmus zur Layout-Berechnung folgt einem *Branch & Cut*-Ansatz. Hierbei werden vielversprechende Teile des Lösungsraums durchsucht, wohingegen Teile, von denen man weiß, dass sie nicht die beste Lösung enthalten, ausgelassen werden. Genauer werden hierfür folgende Ansätze zur Beschleunigung der Berechnung angewendet:

- Eine **Heuristik** wurde entwickelt, welche die erreichbare Qualität einer View-Display-Zuordnung abschätzt ohne hierfür konkret die Layouts zu berechnen. Die Heuristik nimmt an, dass zu einer gegebenen Zuordnung je Display ein “perfektes” Layout existiert, welches die Nutzervorgaben optimal erfüllt. Konkret bedeutet das für die Anzeigequalität, dass ein solch perfektes Layout den gesamten Platz auf einem Display komplett (d.h. ohne Freiräume) und exakt gemäß der Relevanzwerte auf die Views aufteilt. Bezüglich der räumlichen Qualität bedeutet das, dass verbundene Views ohne Abstand direkt nebeneinander dargestellt werden können und bezüglich der zeitlichen Qualität, dass ein View exakt an der gleichen Stelle erscheint wie der View den er ersetzen soll. Zu diesen Annahmen wird gemäß der Qualitätsfunktion der Qualitätswert für das perfekte Layout berechnet. Die über alle Displays aufsummierten Qualitätswerte ergeben auf diese Weise eine heuristische Überabschätzung für die durch eine Zuordnung erreichbare Qualität.

Die Heuristik wird dafür eingesetzt, um frühzeitig Zuordnungen auszusortieren, deren heuristische Qualität schlechter ist, als die (exakt ermittelte) Qualität der bisher besten Lösung (mit bereits erzeugten Layouts).

- Die **Reihenfolge** mit der, der Lösungsraum durchsucht wird, wird so gewählt, dass Lösungen mit einer hohen Qualität möglichst frühzeitig bei der Suche erzeugt werden.

Für diesen Ansatz wird der Lösungsraum gemäß einer Tiefensuche abgearbeitet:

4. Assistierte Präsentation

1. Im ersten Schritt werden die möglichen View-Display-Zuordnungen gemäß der Heuristik erzeugt und sortiert.
2. Für die jeweils vielversprechendste Zuordnung werden im nächsten Schritt die Layouts für alle Displays berechnet. Durch dieses Vorgehen erhält man schnell eine vollständige Lösung.
3. In einem dritten Schritt wird diese vollständige Lösung direkt nach dem Erstellen durch die Qualitätsfunktion bewertet.

Durch dieses Vorgehen erhält man frühzeitig eine exakte und relativ hohe untere Qualitätsschranke für die bestmögliche Lösung. Im Vergleich zu einer willkürlichen Suchreihenfolge, können so (mittels der Heuristik) mehr schlechte Zuordnungen im 1. Schritt identifiziert werden und es kann somit insbesondere der Rechenaufwand für den 2. Schritt verringert werden.

- Ein **Wiederverwenden von Teilergebnissen** wird durch einen Ansatz der dynamischen Programmierung unterstützt.
Hierbei wird für eine gegebene Menge von Views zusammen mit jedem Display jeweils das beste (bereits erzeugte) Layout gespeichert.
Ordnen unterschiedliche Zuordnungen einem Display dieselben Views zu, muss das Layout für dieses Display nur beim ersten Mal berechnet werden und ist direkt für die folgenden Male wieder abrufbar.
Auf ähnliche Weise wird auch die Heuristik dynamisch verfeinert, indem in einem solchen Fall, anstelle der Qualität des perfekten Layouts, die tatsächlich erreichbare Layout-Qualität verwendet wird. Dies erlaubt es, die Qualität einer Zuordnung genauer abzuschätzen.
- **Äquivalenzklassen** werden auf den Views berechnet, um einmal ausgeführte Teilberechnungen auf ähnliche Fälle übertragen zu können.

Views werden hierbei (bzgl. der Layout-Berechnung) als identisch angesehen, wenn sie gleiche Seitenverhältnisse, gleiche Relevanzwerte und gleiche Kantenverbindungen im Präsentationsgraph haben.

Layout-Berechnungen für identische View-Mengen führen (bis auf Vertauschung von identischen Views) zu denselben Layouts mit gleichen Qualitätswerten. Entsprechend können bereits bestimmte Layouts für äquivalente View-Mengen weiterverwendet werden und (bis auf View-Äquivalenz) identische Zuordnungen brauchen nur einmal bearbeitet werden.

Ein ähnlicher Äquivalenzansatz wird auch für die Displays angewendet. Hierbei werden Displays als identisch angesehen, wenn sie bzgl. der Nutzerpositionen den gleichen Sichtbarkeitsfaktor im Raum, sowie die gleiche Größe und das gleiche Seitenverhältnis haben.

Der Ansatz für die Berechnung von Äquivalenzklassen ergänzt sich durch das Identifizieren von ähnlichen View-Mengen und Displays mit dem vorrangigsten

4.2. Layout-Berechnung zur Anzeige der Präsentation

Beschleunigungsansatz bzgl. der Wiederverwendung von Teilergebnissen, indem er die Übertragbarkeit von Teilergebnissen erhöht.

- Zur **Layout-Berechnung** wird ein Grid-basierter Ansatz gewählt, der es erlaubt, für eine gegebene Menge von Views und für ein Display, das bzgl. der Qualitätsfunktion beste Layout sehr schnell zu erzeugen.

Der Grid-basierte Ansatz unterteilt hierbei die Anzeigefläche auf einem Display durch mehrere Zeilen und Spalten. In die entstehenden Zellen werden die Views eingeordnet. Die Zeilen- und Spaltengrößen bestimmen die Position und Größe der Views.

Zur Bestimmung des besten Layouts auf einem Display, werden diejenigen Spalten- und Zeilengrößen ermittelt welche die Qualitätsfunktion optimieren.

Dieses Problem wird als quadratisches Optimierungsproblem mit linearen Constraints formuliert [HR13; BV04]. Für die spezielle Form des vorliegenden quadratischen Optimierungsproblems, kann ein (modifizierter) Simplex-Algorithmus verwendet werden, der das Problem in einer endlichen Anzahl an Schritten (schnell) löst [Wol59]. Der grundlegende Ansatz von Wolfe [Wol59] wurde hierfür in den letzten Jahren weiter verbessert [SN13; KLK12].

Für die Lösung des Optimierungsproblems wird das Toolkit *Gurobi Optimization*¹ verwendet, welches eine aktuelle Implementierung des modifizierten Simplex-Verfahrens bereitstellt. Durch den gewählten Lösungsansatz kann eine einzelne Layout-Berechnung (für ein Display und eine View-Menge) in ca. 10-100 μs ² ausgeführt werden.

Die durch den Algorithmus benötigte Zeit zum Berechnen aller Layouts schwankt stark und ist abhängig von den Kantenverbindungen im Präsentationsgraph, der Anzahl der gleichzeitig anzuzeigenden Views und der Ähnlichkeit der Views und Displays. Tabelle 4.1 zeigt die durchschnittlich benötigte Zeit für unterschiedliche Kombinationen aus Display- bzw. View-Anzahl und View-Ähnlichkeit. Eine komplette Neuberechnung für 15 Views dauert etwa 0,3s bis 0,8s. Für die gleiche Anzahl an Views ist dies etwa 3-5 mal schneller als der Ansatz von Radloff. In einer Präsentation wird diese Zeit beispielsweise benötigt, wenn Nutzer einen komplett neuen Zeitschritt aufrufen, der bisher nicht genutzte Views das erste Mal verwendet. Obwohl beim ersten Aufrufen der Views somit eine leichte Verzögerung bemerkbar bleibt, ist der Algorithmus mit einer Berechnungszeit unterhalb von 1.0s ausreichend schnell, um den Gedankenfluss der Nutzer nicht unangenehm zu unterbrechen [Nie93a; Nie93b]. Dies ermöglicht es einem vortragenden Nutzer, einen geplanten Präsentationsablauf flüssig zu moderieren und hindert das Publikum nicht daran, dem Vortrag zu folgen. Die Verzögerung ist außerdem gering genug, um in Diskussionen das Gefühl einer freien Navigation (über die verwendeten Views) zu erhalten [Nie93a; Nie93b]. Während sich die genannten Zeiten auf komplette Neuberechnungen beziehen, sind kleinere Anpassungen deutlich schneller.

¹ <http://www.gurobi.com/>

² $1 \mu\text{s} = 10^{-6} \text{ s}$

4. Assistierte Präsentation

View-Anzahl	Display-Anzahl und View-Äquivalenz								
	3 Displays			4 Displays			5 Displays		
	25%	50%	75%	25%	50%	75%	25%	50%	75%
11	89	173	101	178	349	245	270	532	487
13	137	253	179	244	451	323	368	637	544
15	283	430	376	417	680	526	562	823	786

Tabelle 4.1.: Die Tabelle zeigt, die durchschnittliche benötigten Zeiten für eine komplette Neuberechnung der gesuchten Layouts in ms ($= 10^{-3}$ s). Die Zeilen beinhalten die Fälle für 11, 13 und 15 Views. Die Spalten unterteilen hierbei weiter nach 3,4 und 5 Displays und verschiedenen Fällen von View-Äquivalenz. Die Prozentangaben geben an, wie viele der Views gleiche Seitenverhältnisse und identische Nutzervorgaben haben (und damit nach dem entwickelten Beschleunigungsansatz in die gleichen Äquivalenzklassen eingeordnet werden). Die gemessenen Zeiten sind die Durchschnittswerte aus jeweils 100 zufällig erzeugten Beispielen. Sie wurden auf einem Intel Core i7-3520 mit 2,9 GHz ermittelt.

Einzelne, interaktive Anpassungen an einer bestehenden Anzeige (wie z.B. das Hinzufügen einer neuen View oder das Ändern von Relevanzwerten) erfordern i.d.R. keine komplette Neuverteilung aller Views. In solchen Fällen sind die Berechnungen, insbesondere durch die beschriebenen Ansätze zur Wiederverwendung von Teillösungen, etwa 10 bis 100-mal schneller und liegen damit für einen Großteil der Fälle unterhalb von 0,1 s. Dies ist nach der Klassifikation von Nielsen [Nie93b] ausreichend schnell, um von den Nutzern als sofortige Reaktion auf eine Eingabe empfunden zu werden und somit ein Gefühl von direkter Manipulation (der Informationsanzeige) entstehen zu lassen. Für kleinere Fallbeispiele mit weniger Views nimmt die benötigte Berechnungszeit weiter (exponentiell) ab und liegt oft unter der benötigten Zeit zum Zeichnen der Views.

Zusammenfassend kann also festgestellt werden, dass der entwickelte Ansatz zur Layout-Berechnung einerseits die automatische Verteilung und Anordnung der Views gemäß den Nutzervorgaben realisiert (Anforderungen L1-L3) und, dass andererseits, die Aktualisierung der hierfür notwendigen Layouts ausreichend schnell ist, um ein nutzergesteuertes Anpassen der Anzeige bei interaktiven Frameraten zu ermöglichen.

In Kombination mit dem entwickelten Nutzer-Interface erlaubt der Layout-Algorithmus, die Informationsanzeige im Sinne einer Display Ecology für verschiedene Aufgaben spontan anzupassen, indem die Zusammenstellung der Views interaktiv gesteuert werden kann und die Anzeige automatisch aktualisiert wird.

Um auch Analyse-Aufgaben besser unterstützten zu können, wird nochmals ein Schritt weitergegangen: Anstelle die Aktualisierung der Anzeige auf das Kombinieren von bereits existierenden Views zu beschränken, wird nun ein Ansatz eingeführt, der auch das Generieren und Integrieren von komplett neuen Views erlaubt. Dieser Ansatz wird im nächsten Abschnitt vorgestellt.

4.3. Spontan generierte Views

Multi-Display-Umgebungen werden für das Analysieren von Daten und das Präsentieren von Informationen genutzt. Beim traditionellen Arbeitsablauf (in Abbildung 4.3 schwarz dargestellt) werden hierbei zunächst die Daten untersucht (Exploration), anschließend werden Erkenntnisse zusammengetragen und Informationen aufbereitet (Authoring) und schließlich mithilfe von ausgesuchten Bildern oder Videos einem Publikum vorgetragen (Presentation) [Lee+15; Gra+16].

Bisher werden in diesem Prozess Präsentation und Exploration noch getrennt betrachtet. Während die von Chung in [Chu+15] betrachteten Display Ecologies die Exploration unterstützten, steht in Smart Meeting Rooms die Präsentation im Vordergrund (vgl. Abschnitt 2.2). Dagegen wäre ein nahtloser Übergang vom Präsentieren zum Explorieren wünschenswert, wie es in einer aktuellen Arbeit von Gratzl et al. motiviert wird [Gra+16]. So könnten z.B. einzelne Explorationsschritte interaktiv vorgeführt werden, um die gewonnenen Erkenntnisse, sowie den Analyse-Vorgang selbst, einem Publikum vorzuführen und somit besser nachvollziehbar zu machen. Weiterhin würde es so auch möglich werden, von einem (für die Präsentation vorbereiteten) Analyseprozess bei Bedarf abzuweichen und beispielsweise Hypothesen, die während einer Diskussion spontan aufkommen, zu überprüfen.

In diesem Abschnitt wird deshalb ein Ansatz vorgestellt, der genau dies ermöglicht und somit die bestehende Lücke zwischen Exploration und Präsentation schließt.

Die bisher in dieser Dissertation vorgestellten Ansätze erlauben es, mehrere Views (in einem Authoring-Schritt) zusammenzutragen und zur Unterstützung der Präsentation auf multiplen Geräten anzuzeigen (vgl. Abschnitt 4.1). Hierbei kann bereits die Zusammenstellungen der auf den Displays gezeigten Views jederzeit manuell angepasst werden, um nachträglich zusätzliche Inhalte in eine Präsentation aufzunehmen. Die hierfür zur Verfügung stehende Auswahl beschränkt sich jedoch noch auf die von den Nutzern bereitgestellten Inhalte.

Um eine durch den Nutzer gesteuerte Exploration zu ermöglichen, sollen nun auch spontan generierte Views, bei Bedarf in die Anzeige auf den Displays integriert werden. Der Unterschied zu den bisher betrachteten Views besteht bei diesen darin, dass die gezeigten Visualisierungen nun erst während des Vortrags bzw. während der Diskussion erzeugt bzw. durch den Nutzer interaktiv angepasst werden.

Hierdurch kann gezielt auf gestellte Fragen oder auf wechselnde Diskussionsthemen reagiert werden. Beispielsweise wird es so möglich, einen für die Beantwortung einer Frage relevanten Bildausschnitt interaktiv auszuwählen und vergrößert auf einem Display anzuzeigen. Andererseits können so aber auch komplett neue Sichten auf zu untersuchende Daten bedarfsgerecht generiert und (zusammen mit den bestehenden Views) in der Multi-Display-Umgebung angezeigt werden. Insbesondere für die Exploration von großen Datenmengen mit mehreren Dimensionen und mit einem räumlichen und zeitlichen Bezug könnten so, je nach Analyse-Schwerpunkt, Visualisierungen für verschiedene

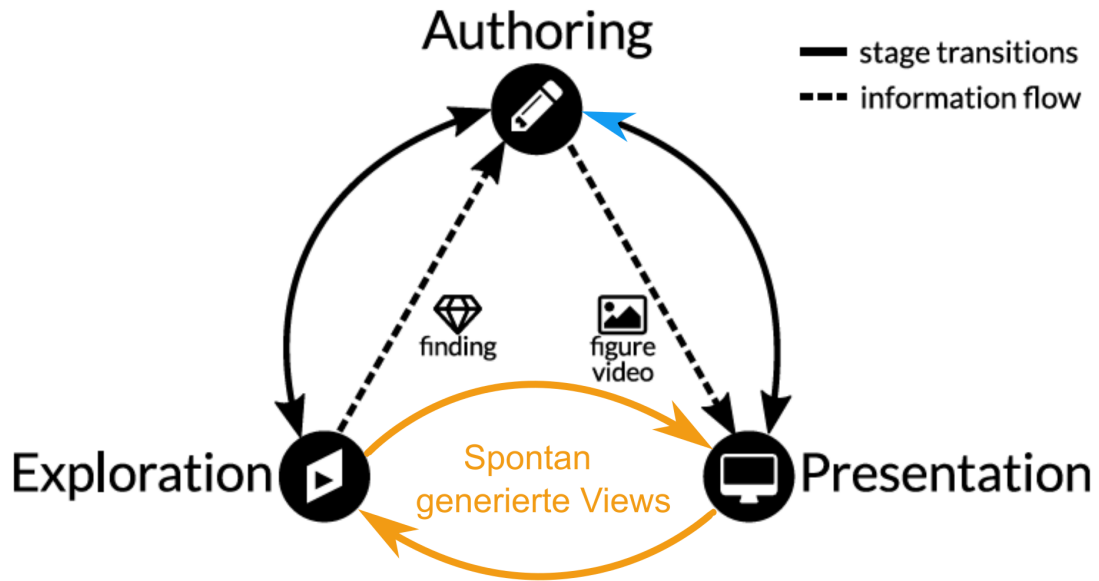


Abbildung 4.3.: Die schwarzen Teile der Abbildung zeigen den nach Gratzl et al. beschriebenen traditionellen Analyse- und Präsentationsablauf, bei dem die Schritte Exploration, Authoring und Presentation nacheinander durchlaufen werden [Lee+15; Gra+16]. Der blaue Pfeil kennzeichnet die nachträgliche Anpassbarkeit der Informationsanzeige, wie sie durch die bisher in Kapitel 4 vorgestellten Ansätze bereits ermöglicht wird. Die orangenen Pfeile zeigen das nun thematisierte spontane Generieren bzw. Anpassen von Views, sowie deren nahtlose Integration in die Anzeige, um die Lücke zwischen Exploration und Präsentation zu schließen. Die Darstellung der traditionellen Teile im Bild wurde aus [Gra+16] übernommen und durch die farbig markierten Teile ergänzt.

Aspekte interaktiv erstellt und nebeneinander dargestellt werden. Ein weiteres Anwendungsbeispiel besteht darin, je nach Bedarf, auf komplett neue Informationen aus alternativen Quellen (z.B. aus dem Internet) zuzugreifen und in den Views anzuzeigen.

Das spontane, nutzergesteuerte Erzeugen von Views und das Integrieren dieser Views in eine bestehende Anzeige auf den Displays ist somit ein wichtiger Bestandteil, um neben der Präsentation auch die Exploration von Daten in Multi-Display-Umgebungen zu unterstützen. Um hierbei auch einen nahtlosen Übergang zwischen Präsentation und Exploration zu erreichen, müssen noch folgende Anforderungen beachtet werden:

- O1:** Das Generieren von neuen Views, muss durch die Nutzer einfach von den persönlichen Geräten aus steuerbar sein. Hierfür sollen verschiedene (lokal auf den Geräten vorhandene) Tools zum interaktiven Erzeugen bzw. Anpassen der in den Views gezeigten Visualisierungen genutzt werden können.

- O2:** Diese Tools müssen einfach während der Präsentation aufgerufen werden können, um bei Bedarf Views anzupassen bzw. neue Views zu erzeugen.
- O3:** Die spontan erzeugten Views müssen sowohl in dem Nutzerinterface auf den persönlichen Geräten der Nutzer sichtbar sein, als auch in die Informationsanzeige auf den Displays der Umgebung integriert werden. Hierfür ist ein Assistenzansatz notwendig, der diese Aufgabe automatisch für die Nutzer übernimmt (und somit kein weiteres Authoring erfordert).

Bisherige Ansätze im Bereich der View Management-Systeme bieten eine umfassende Auswahl an Tools, die es erlauben, Bildausschnitte auf einem persönlichen Gerät zu definieren und über Netzwerk als neue Views an andere Geräte zu übertragen (vgl. Abschnitt 2.2). Beispiele für solche Tools sind u.a. das WinCuts-System [TMC04] oder der View-Grabber [Rad14].

Neben einer rein manuellen Definition einer View, gibt es darüber hinaus aber auch Anwendungen, die es ermöglichen, mehrere Views automatisch zu erstellen [BE14; Rad14; RNS12]. Zur Verwendung dieser Tools werden diese meist manuell durch den Nutzer gestartet, die erzeugten Views werden von den Nutzern ausgewählt und anschließend einem Displays zugewiesen. Die existierenden View-Management-Systeme decken somit die Anforderung O1 gut ab. Dagegen wird der zur Erfüllung von Anforderung O2 und O3 notwendige Konfigurationsaufwand weitestgehend dem Nutzer überlassen.

Andere Ansätze im Bereich der Präsentationssysteme für Multi-Display-Umgebungen ermöglichen es, direkt in der Präsentationsansicht interaktiv Ausschnitte zu bestimmen und diese (ohne weiteres manuelles Authoring) als Views weiterzuverwenden. Im Ansatz von Lanir et al. werden hierfür beispielsweise vom Nutzer definierte Bildausschnitte automatisch vervielfältigt und als neue Views unverändert auf einem separaten Display angezeigt [LBT08]. Während auf diese Weise teilweise die Anforderungen O2 und O3 adressiert werden, erlaubt es der beschriebene Ansatz jedoch nicht, erzeugte Views nachträglich anzupassen (Anforderung O1) oder mit den zuvor auf einem Display gezeigten Views automatisch zu kombinieren (Anforderungen O3).

Dagegen existieren für das interaktive Storytelling Ansätze, welche die Exploration stärker in die Präsentation integrieren, indem fließend zwischen Analyse- und Präsentations-Ansichten gewechselt werden kann [Kwo+14; SH14; Lee+15; Gra+16]. Die entwickelten Systeme sind jedoch i.d.R. für ein spezielles Anwendungsgebiet, wie z.B. die Analyse von ausgesuchten medizinischen Daten [Kwo+14; Lex+12], entwickelt worden und behandeln zudem weniger das Übertragen von Views zwischen verschiedenen Geräten oder deren Verwendung in einer Multi-Display-Umgebung.

Zusammenfassend lässt sich also feststellen, dass eine Lösung, welche die Anforderung O1-O3 gleichzeitig und umfassend umsetzt, in der gängigen Literatur noch nicht vorgestellt wurde.

In der vorliegenden Dissertation wurde deshalb ein Ansatz entwickelt, der die Anforderungen O1-O3 erfüllt. Über das in Abschnitt 4.1 vorgestellte Nutzerinterface (zur Konfiguration der Präsentation) können Nutzer hierfür verschiedene Visualisierungs-

4. Assistierte Präsentation

Tools mit einer konkreten View verknüpfen. Hierdurch lassen sich die verknüpften Tools bei Bedarf schnell auf den persönlichen Geräten aufrufen und zum interaktiven Erstellen bzw. Anpassen von Visualisierungen nutzen (Anforderung O1/O2). Die so erzeugten neuen Views werden automatisch in den Präsentationsgraph integriert (Anforderung O3). Hierdurch werden sie (durch den in Abschnitt 4.2 beschriebenen Layout-Algorithmus) unmittelbar auf den Displays des Raums sichtbar gemacht.

Als konkrete Beispiele wurden im Rahmen der Dissertation drei Tools zur interaktiven Generierung von Views bereitgestellt, um die Exploration von Bilddaten, von Volumendaten und die Verwendung von Web-basierten Inhalten zu unterstützen.

Für die Exploration von (großen) Bilddaten, wurde ein Lupen-Tool entworfen, das es erlaubt, von einer bestehenden View interaktiv ausgewählte Bildausschnitte vergrößert darzustellen und als neue Views anzuzeigen.

Die visuelle Analyse von Volumendaten wird anhand eines Tools zur Feature-basierten Analyse von multivariaten Volumendaten demonstriert. Dieses Tool erlaubt es, interaktiv neue Sichten von diesen Daten zu erzeugen und für die Exploration in einer Multi-Display-Umgebung anzuzeigen.

Um darüber hinaus bedarfsgerecht auch eine Vielzahl anderer Inhalte aufrufen zu können, wurde zudem ein Web-Tool entwickelt, das Internetseiten und insbesondere auch online verfügbare Analysewerkzeugen [Mwa+16] in einer Multi-Display-Umgebung nutzbar macht.

Im Folgenden wird die grundlegende Vorgehensweise für das bedarfsgerechte Generieren, Integrieren und Aktualisieren von Views beschrieben. Anschließend werden die Anwendungsfälle genauer thematisiert.

Vorbereitung: Um bei Bedarf schnell zwischen Exploration und Präsentation wechseln zu können, müssen die verschiedenen Tools für die View-Generierung aus dem Nutzerinterface (auf den persönlichen Geräten) aufgerufen werden können. Dazu werden die Tools mit denjenigen Views verlinkt, die (durch diese Tools) verändert oder ersetzt werden sollen. Die Nutzer kennzeichnen hiermit also, welches Tool auf ihrem persönlichen Gerät für die Exploration verwendet und welche Views der bestehenden Anzeige hierdurch aktualisiert werden sollen.

Zum Erstellen einer Tool-Verknüpfung können Nutzer entweder die Darstellung des Präsentationsgraphen oder die Layer Bar im Nutzerinterface verwenden (vgl. Abschnitt 4.1).

- Um ein Tool über den Präsentationsgraph mit einer View zu verbinden, wird es zunächst per Drag & Drop an eine beliebige Stelle der Zeichnung gezogen. Hierdurch wird dem Präsentationsgraphen ein neuer Knoten mit einer Verknüpfung zum ausgewählten Tool hinzugefügt. Anschließend kann durch Erstellen von gruppierenden Kanten das Tool mit anderen Views verknüpft werden.

Das beschriebene Vorgehen erlaubt es, Tools an unterschiedlichen Stellen des

Präsentationsgraphen einzubinden und diese mit Views zu verlinken, die an verschiedenen Zeitpunkten einer Präsentation verwendet werden. Hierdurch ist es möglich, die bedarfsgerechte View-Erzeugung für einen Vortrag oder für eine Diskussion im Voraus zu planen und entsprechend zu konfigurieren.

- Um ein Tool mit einem in der Layer Bar gezeigten View zu verknüpfen, kann es hierzu auch direkt auf diesen View gezogen werden (vgl. Abbildung 4.4). Dadurch wird automatisch eine neue Tool-Verknüpfung in den Präsentationsgraphen eingefügt und über eine gruppierende Kante mit der ausgewählten View verbunden.

Mit dem beschriebenen Vorgehen können Tools zur Ergänzung der Anzeige ohne weitere Vorbereitung spontan in die Präsentation eingebunden werden. Insbesondere wird so das Verknüpfen mit den gerade auf den Displays gezeigten Views unterstützt, weil die Layer Bar diese Views immer gut sichtbar im Nutzerinterface darstellt.

Die auf diese Weise erstellten Verknüpfungen zwischen Tools und Views können nun dazu genutzt werden, um die Tools während der Präsentation einfach aufzurufen.

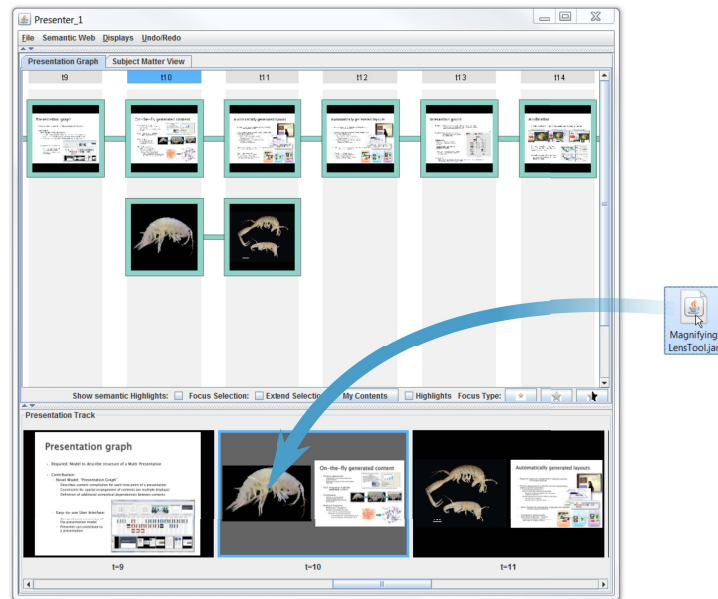
Erzeugung neuer Views: Um bei Bedarf ein Tool zu starten und somit neue Views zu erzeugen bzw. existierende Views anzupassen, reicht es nun aus, wenn ein Nutzer einfach eine verknüpfte View (doppelt) anklickt. Daraufhin werden die mit dieser Views verbunden Tools automatisch ermittelt und auf den persönlichen Geräten der Nutzer ausgeführt.

Dieses Vorgehen kann allerdings für jedes Tool individuell (durch den Nutzer, der das Tool bereitstellt) eingeschränkt werden, um ein unerwünschtes Starten des Tools (durch andere Nutzer) zu verhindern. Über ein Kontextmenü lässt sich hierfür an der Verknüpfung des Tools einstellen, ob es nur manuell durch den bereitstellenden Nutzer gestartet werden darf, ob auch andere Nutzer das Tool ausführen dürfen oder ob es durch das Präsentationssystem automatisch gestartet werden soll. Bei einem automatischen Start werden die Tools ohne weitere Nutzerinteraktion ausgeführt, sobald im Präsentationsgraph der Zeitpunkt mit einer Tool-Verknüpfung aufgerufen wird. Hierdurch kann die Exploration auch als fester Bestandteil in bspw. einem Vortrag eingeplant werden.

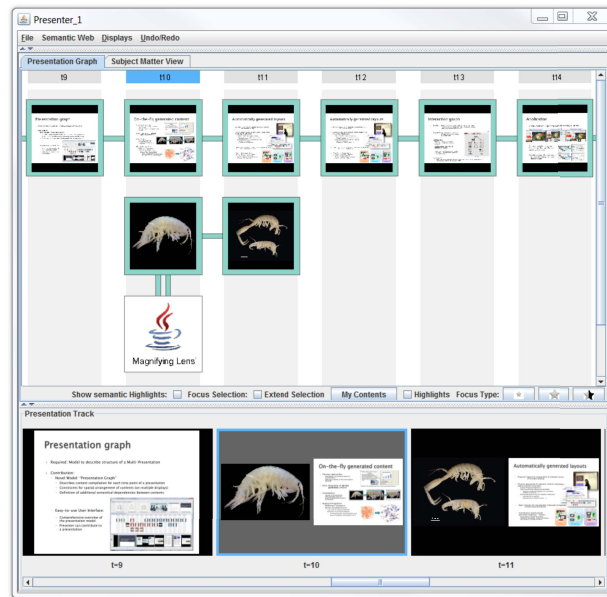
Anzumerken ist noch, dass jedem Tool darüber hinaus auch entsprechende Startparameter übergeben werden können, die beispielsweise eine aufzurufende URL für die Integration von Web-Inhalten oder den Speicherort des initial zu ladenden Datensatzes für eine Volumenvisualisierung angeben. Hierdurch ist es möglich, verschiedenartige Tools individuell vorzukonfigurieren.

Sobald die Tools gestartet wurden, erzeugen sie neue Views, welche nun wiederum auf den Displays der Umgebung angezeigt werden müssen.

4. Assistierte Präsentation



(a) Vor dem Verknüpfen



(b) Nach dem Verknüpfen

Abbildung 4.4.: Verknüpfen des Lupen-Tools mit einer View über das Nutzer-Interface auf dem persönlichen Gerät eines Nutzers. (a) Das Tool wird per Drag & Drop über die Layer Bar mit einer View verbunden. (b) Präsentationsgraph mit eingefügter Verknüpfung zum Tool und gruppierenden Kanten zum zuvor ausgewählten View. In der Zeichnung des Präsentationsgraphen wird für die Verknüpfung der Name des Tools angegeben und die gruppierende Kante wird als doppelte Linien dargestellt.

Integration der Views: Alle durch die Tools generierten Views werden zunächst in den Content Pool des Nutzerinterface eingeordnet. Von dort müssen sie wiederum in den Präsentationsgraph integriert werden, um sie auch auf den Displays des Raums sichtbar zu machen.

Prinzipiell besteht die Möglichkeit, dass die Nutzer das Interface auf den persönlichen Geräten dazu verwenden, um diese Aufgabe selber auszuführen. Der hierfür notwendige manuelle Editiervorgang würde jedoch den Präsentationsfluss unterbrechen und den Nutzer vom Bedienen des Tools und somit von der Exploration ablenken.

Um dagegen sofort nach dem Starten eines Tools die Exploration aufnehmen zu können, werden deshalb die generierten Views automatisch in den Präsentationsgraph eingefügt. Hierfür werden, ausgehend von einer Tool-Verknüpfung, die gruppierenden Kanten verfolgt, um diejenigen Zeitschritte im Präsentationsgraph zu ermitteln, an denen Views aktualisiert werden sollen. In die so ermittelten Zeitschritte werden anschließend die neu generierten Views so eingefügt, dass sie entweder die zu aktualisierenden Views ergänzen oder diese ersetzen.

Diese Integration der Views in den Präsentationsgraph führt zu einer Aktualisierung der Anzeige auf den Displays des Raums. Wenn die neu generierten Views den Präsentationsgraph ergänzen, erscheinen sie zusammen mit den alten Views auf den Displays. Dieses Variante ist in Abbildung 4.5 Nummer 3a bzw. in Abbildung 4.6(b) zu sehen.

Beim Ersetzen verschwinden die zu aktualisierenden Views und es erscheinen an ihrer Stelle die neu generierten Views (vgl. Abbildung Nummer 3b).

Die verwendete Vorgehensweise zum Integrieren der Views (Ergänzen oder Ersetzung) kann der Nutzer entweder bereits im Vorbereitungsschritt für jede Tool-Verknüpfung individuell vorgeben oder auch nachträglich jederzeit während der Exploration bzw. Präsentation ändern. Einerseits kann hierdurch das Verhalten an die Art des Tools, an die Anzahl der generierten Views und an den auf den Displays verfügbaren Platz angepasst werden. Andererseits können so aber auch, je nach dem aktuellen Explorationsfortschritt, ältere Ansichten ausgeblendet oder bei Bedarf wieder angezeigt werden.

Der in diesem Abschnitt beschriebene Ansatz wird nochmal in Abbildung 4.5 zusammengefasst. Durch die im Vorbereitungsschritt angelegten Verknüpfungen zu Explorations-Tools, ist es einem Nutzer möglich, diese bei Bedarf schnell aufzurufen. Über sein persönliches Gerät kann der Nutzer nun direkt mit den Tools interagieren, um neue Visualisierungen für die Exploration zu erzeugen oder existierende Views anzupassen. Diese so neu generierten Views werden automatisch in die bestehende Anzeige auf den Displays der Umgebung integriert. Auf diese Weise erlaubt es das beschriebene Vorgehen, jederzeit nahtlos zwischen der Präsentation und der Exploration zu wechseln. Weil

4. Assistierte Präsentation

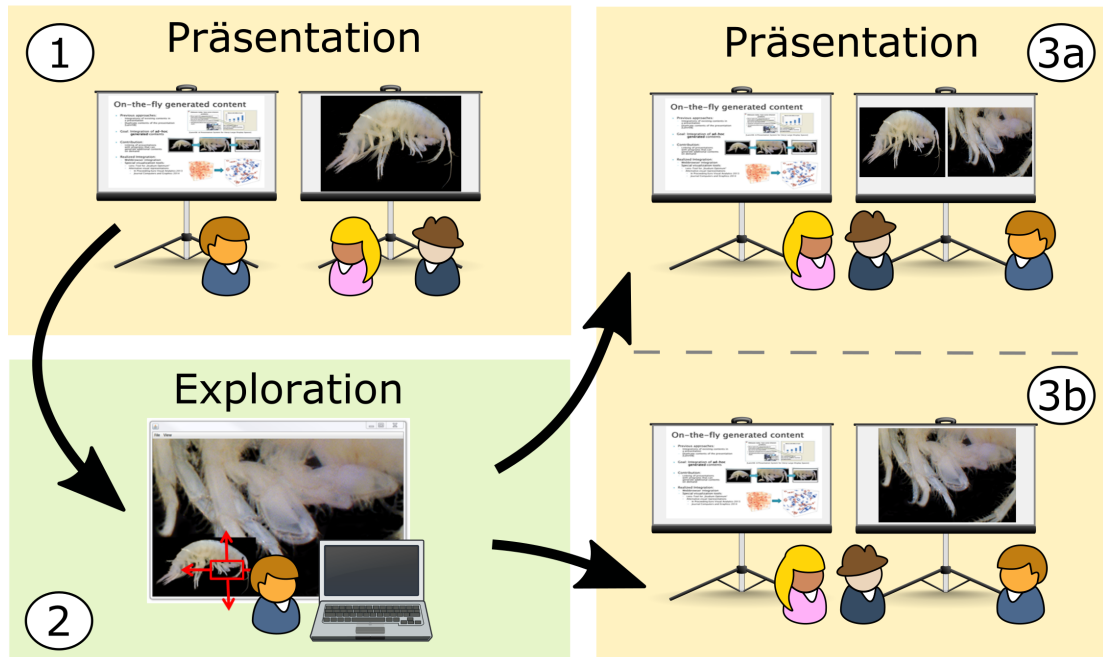


Abbildung 4.5.: Der Arbeitsablauf unter Verwendung der spontan generierten Views. (1) Aus einer bestehenden Präsentation werden verlinkte Tools aufgerufen. (2) Diese werden genutzt, um neue Views in einem Explorationschritt zu erzeugen bzw. um existierende Views anzupassen. (3) Die neu generierten Views werden automatisch in die Anzeige auf den Displays integriert, wobei sie entweder die bestehenden Views ergänzen (3a) oder die alten Views ersetzen (3b).

die generierten Views hierbei laufend aktualisiert werden, können andere Nutzer den Explorationsvorgang auf den Displays verfolgen.

In den folgenden Unterabschnitten wird die Anwendung dieses Ansatzes anhand von drei konkreten Beispielen genauer beschrieben.

4.3.1. Exploration von Bilddaten

Bei großen Bildern kann die Bildauflösung über der View-Auflösung liegen, wodurch eine Skalierung des Bilds notwendig wird und somit ursprünglich enthaltene Details nicht vollumfänglich auf den Displays dargestellt werden können. Außerdem kann es für die Analyse eines Bilds erforderlich sein, dass nur bestimmte Bildteile von Interesse (mit allen Details) gezeigt werden und andere Teile dafür bewusst ausgeblendet werden. Dabei ist in der Regel der interessante Bildausschnitt nicht statisch festgelegt, sondern wird sich während der Analyse verändern.

Um die Exploration solcher Bilddaten zu unterstützen, wurde ein Lupen-Tool entwickelt, das es ermöglicht, interaktiv festgelegte Bildausschnitte mit vollen Details in neu erzeugten Views darzustellen. Hierfür erlaubt es das Tool, entweder die restlichen Bildteile komplett auszublenden oder diese mit einer geringeren Auflösung anzuzeigen.

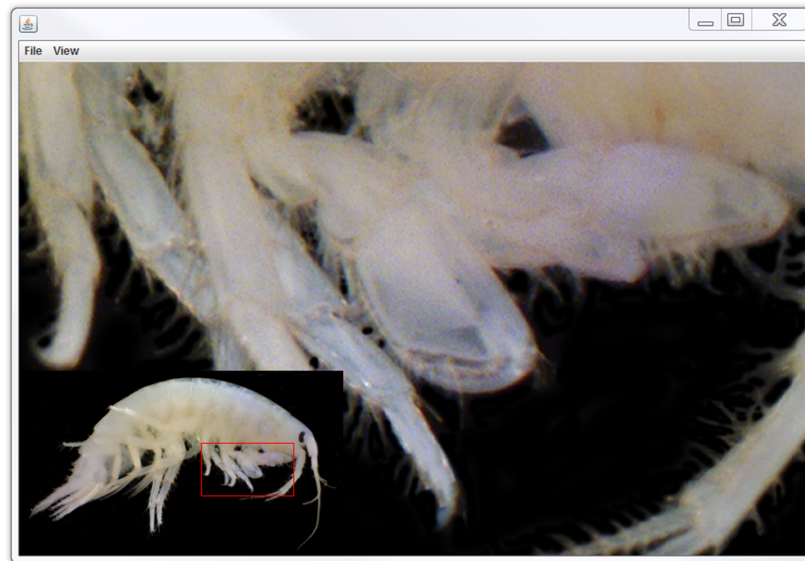
Beim Arbeiten mit dem Tool wird nach dessen Start automatisch ein separates Fenster auf dem persönlichen Gerät des Nutzers geöffnet (vgl. Abbildung 4.6(a)). Dieses Fenster zeigt die mit dem Tool verknüpfte View und kann nun vom Nutzer dazu verwendet werden, um eine Region von Interesse auf dieser View interaktiv festzulegen. Durch Reinzoomen in die Darstellung können hierzu mehr Details des ursprünglichen Bilds eingeblendet werden und durch Verschieben, des im Fenster gezeigten Bildausschnitts, kann die Region von Interesse auf andere Bildteile fokussiert werden. Eine einblendbare Overview-Ansicht in der linken unteren Ecke des Fensters hebt die so erstellte Region von Interesse durch eine rote Umrandung hervor. Anhand dieser zusätzlichen Anzeige können Nutzer die Position und Größe der Region kontrollieren. Außerdem lässt sich so ein gesuchter Bildteil einfach identifizieren und für die genauere Betrachtung in der Region zentrieren.

Der so festgelegte Bildausschnitt wird (in dem vom Nutzer festgelegten Detailgrad) aufgenommen und als neu generierte View in die Präsentation integriert. Abbildung 4.6(b) zeigt beispielsweise, wie die so neu erzeugte View die zuvor (in Abbildung 4.4) mit dem Lupen-Tool verlinkte View im Präsentationgraph ergänzt. Die daraus resultierende Anzeige auf den Displays ist in Abbildung 4.5 Nummer 3a dargestellt.

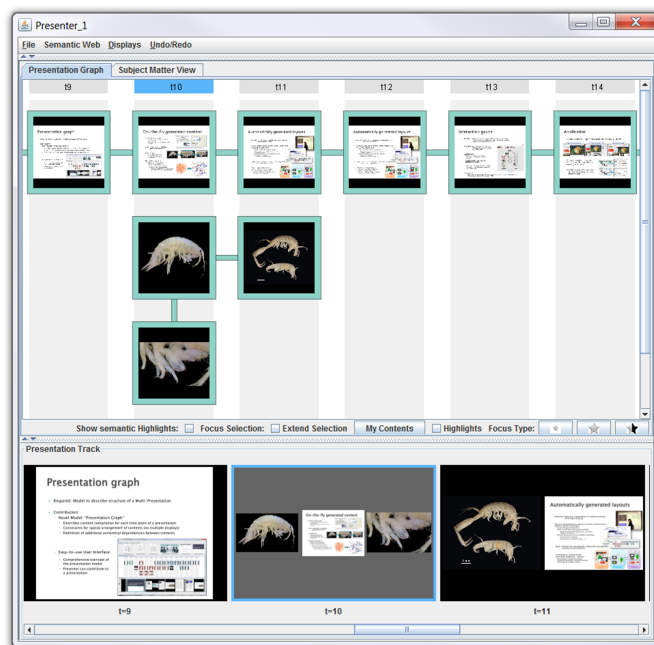
Im Folgenden soll ein konkretes Anwendungsbeispiel für das Lupen-Tool diskutiert werden. In einem Seminar der Meeresbiologie an der Universität Rostock sollen Krebstiere anhand von hochaufgelösten Aufnahmen untersucht werden. Die Bilder zeigen Tiere, die aus unterschiedlichen Lebensräumen stammen und sollen dazu verwendet werden, um die spezifischen Körpermerkmale der Tiere zu erläutern und untereinander zu vergleichen. Damit hierbei einzelne Körperteile (wie z.B. die Greifwerkzeuge oder die Beine in Abbildung 4.6(a)), genauer betrachtet werden können, müssen Teile der ursprünglichen Bilder bei Bedarf vergrößert und in voller Auflösung auf den Displays des Raum angezeigt werden.

Im Rahmen des Universitätsprojekts “Studium Optimum” wurde das Lupen-Tool hierfür im Seminar erprobt. Als positive Rückmeldung wurde von dem Seminarleiter hervorgehoben, dass die Bildausschnitte durch die bereitgestellte Funktionalität nicht mehr, wie

4. Assistierte Präsentation



(a) Gestartetes Lupen-Tool.



(b) Präsentationsgraph mit automatisch integrierter View

Abbildung 4.6.: Generierung einer neuen View durch das gestartete Lupen-Tool (a) und die automatische Integration der neuen View in den bestehenden Präsentationsgraph (b). Die neue View ergänzt die ursprünglich (in Abbildung 4.4(a)) verlinkte View auf den Displays (vgl. Abb. 4.5 Nr. 3a).

sonst üblich, manuell vor dem Seminar vorbereitet werden mussten, sondern stattdessen direkt während der Präsentation erstellt und bei Bedarf angepasst werden konnten. Hierdurch konnte der Seminarleiter das übliche Vorgehen zum Klassifizieren der abgebildeten Krebstiere an konkreten Beispielen erläutern.

Der vorgestellte Anwendungsfall demonstriert somit an einem einfachen Beispiel, wie durch bedarfsgerecht generierte Views die sonst eher klassische Präsentation von Bilddaten im Rahmen eines Seminars durch eine spontane Exploration bereichert werden kann.

Der grundlegende Ansatz basiert hierbei noch darauf, Teile von existierenden Bilddaten (gemäß den Nutzervorgaben) unterschiedlich darzustellen. Im folgenden Anwendungsfall wird nun noch einen Schritt weiter gegangen und auch die Generierung von komplett neuen Visualisierungen aus den gegebenen Daten thematisiert.

4.3.2. Visuelle Datenexploration von Volumendaten

Nach Bertin werden Visualisierungen von Daten nicht nur einmalig erstellt, sondern solange neu konstruiert und angepasst bis die grundlegenden Zusammenhänge in den Daten sichtbar werden [Ber81].

Dies trifft insbesondere auf Visual Analytics-Ansätze zu, bei denen schrittweise immer neue Visualisierungen erzeugt werden, die dann dazu dienen, immer neue Aspekte oder Teile der Daten zu untersuchen [TC05; KH13]. Mithilfe der Visualisierungen werden im Laufe von diesem Prozess bestehende Hypothesen bestätigt bzw. verworfen und wiederum neue Hypothesen aufgestellt.

Um Visual Analytics auch in einer Multi-Display-Umgebung zu ermöglichen, ist es somit entscheidend, dass alternative Darstellungen von Daten bei Bedarf durch die Nutzer interaktiv erstellt, als Views auf den Displays angezeigt und bei Bedarf nachträglich wieder angepasst werden können.

Ein Anwendungsgebiet für Visual Analytics ist das Untersuchen von multivariaten Volumendaten [KH13]. Hierbei werden typischerweise mehrere Visualisierungen gleichzeitig verwendet, um die verschiedene Datendimensionen in Kombination zu untersuchen. Weiterhin muss einerseits der Raumbezug der Daten visualisiert werden, um bspw. Regionen von Interesse im Darstellungsraum genauer betrachten zu können und andererseits muss der zeitliche Bezug berücksichtigt werden, um die Veränderungen in den Daten analysieren zu können. Neben Darstellungen die hierfür alle in den Daten enthaltenden Details zeigen, sind auch Visualisierungen erforderlich, welche nur die interessanten Eigenschaften darstellen bzw. einen abstrahierenden Gesamtüberblick über die Daten liefern.

Um insbesondere auch die letzten beiden Punkte zu adressieren, wurde das sog. “FeVis-Tool”³ entwickelt [Eic+13; Eic+14]. Das FeVis-Tools erlaubt es Nutzern, verschieden-

³Abkürzung für “Feature-Visualization-Tool”

4. Assistierte Präsentation

artige Feature-basierte Visualisierungen von multivariaten Volumendaten interaktiv zu erstellen. Bei dem verfolgten Ansatz bestimmen die Nutzer, an welchen Dateneigenschaften sie gerade interessiert sind. Basierend auf diesen Vorgaben werden anschließend Regionen im Raum ermittelt, welche diese Eigenschaft aufweisen. Die so extrahierten Regionen von Interesse werden anschließend visuell als Features in bestehenden Darstellungen hervorgehoben oder wahlweise in eigenständigen Views angezeigt. Hierbei können Visualisierungen erzeugt werden, die wahlweise mehr den räumlichen bzw. den zeitlichen Datenbezug berücksichtigen und unterschiedlich stark von den Rohdaten abstrahieren.

Für die Analyse von Volumendaten, in Multi-Display-Umgebung kann das FeVis-Tool nun auch dazu verwendet werden, um bei wechselnden Regionen von Interesse neue Visualisierungen der Daten während einer Präsentation zu generieren und als Views auf den Displays anzuzeigen.

An dieser Stelle soll hierfür wieder ein konkretes Anwendungsbeispiel diskutiert werden. Um biochemische Vorgänge zu untersuchen, werden Modelle eingesetzt, welche die Bewegungen und Reaktionen von biochemischen Partikeln simulieren [EE04; Bit+11]. Diese Modelle werden beispielsweise verwendet, um die Proteine in biologischen Zellen und ihre chemischen Reaktionen miteinander zu beschreiben [HNR02; Haa+13; Bit+12]. Untersuchungsgegenstand ist hierbei, wie bestimmte Umweltbedingungen das Diffundieren von Proteinen in andere Bereiche der Zelle, das Binden von Proteinen an Zellbestandteile und das Ablaufen von Proteinreaktionen beeinflussen.

Die Daten, die durch diese Simulationen generiert werden, enthalten Informationen über die räumliche Bewegung, die Entstehung und das Auslöschen von tausenden mikroskopischer Partikel. Um die simulierten biochemischen Vorgänge zu verstehen, müssen die Simulationsexperten verschiedene Aspekte dieser multivariaten Volumendaten analysieren:

- Die räumliche Verteilung der Partikel im 3D-Simulationsraum muss erfasst werden, um erkennen zu können, ob die Partikel gleichmäßig verteilt sind oder ob es Bereiche mit signifikant höheren oder niedrigeren Partikelkonzentrationen gibt.
- Die zeitliche Dimension der Daten muss berücksichtigt werden, um die Änderungen der Partikelverteilung über die Zeit zu untersuchen.
- Die Kombination aus räumlicher und zeitlicher Entwicklung muss in Hinsicht auf verschiedene Partikelarten betrachtet werden. Dies ist notwendig, um eine über den Simulationsverlauf stattfindende räumliche Trennung bzw. Korrelation von verschiedenen Proteinarten erkennen zu können (wie sie durch bestimmte chemische Reaktionen hervorgerufen wird).

Der Untersuchungsschwerpunkt kann sich hierbei während der Analyse verlagern (z.B. auf eine andere Proteinart oder einen anderen Zeitabschnitt), wodurch die verwendeten Darstellungen durch die Analyse-Experten laufend angepasst oder komplett neue Ansichten erstellt werden müssen. Aufgrund der Vielzahl von möglicherweise notwendigen

4.3. Spontan generierte Views

Visualisierungen ist es schwierig, diese im Vorfeld anzufertigen. Deshalb bietet sich hier ein schrittweises Verfahren an, bei dem ausgehend von einer gewissen Anzahl visueller Repräsentationen weitere Views je nach Bedarf erzeugt und spontan in die Präsentation eingefügt werden.

Um die Simulationsdaten auszuwerten, bietet es sich an, zunächst die Rohdaten genauer zu betrachten. Hierfür kann das FeVis-Tool dazu verwendet werden, um die Proteinkonzentration zu einem bestimmten Zeitpunkt mittels Direct Volume Rendering darzustellen. Abbildung 4.7 zeigt solche Darstellungen für zwei ausgewählte Proteinarten. Mittels des FeVis-Tools können bei der Analyse mehrere dieser Views für ausgesuchte Proteinarten bzw. Simulationszeitpunkte generiert und gleichzeitig in einer Multi-Display-Umgebung auf den Displays angezeigt werden (vgl. Abbildung 4.8). Hierdurch ist es bereits möglich zu erkennen, ob die Partikel gleichmäßig über den Raum verteilt sind (vgl. Abbildung 4.7(a)) oder ob es Bereiche mit einer Häufung von bestimmten Partikeln gibt (vgl. Abbildung 4.7(b)). Außerdem können auch die zeitliche Entwicklung und die räumliche Korrelation von bestimmten Proteinarten schon grob abgeschätzt und somit potentiell interessante Eigenschaften bzw. Zusammenhänge für die weitere Analyse identifiziert werden.

Um diese nun genauer zu untersuchen, sind jedoch weitere Darstellungen erforderlich.

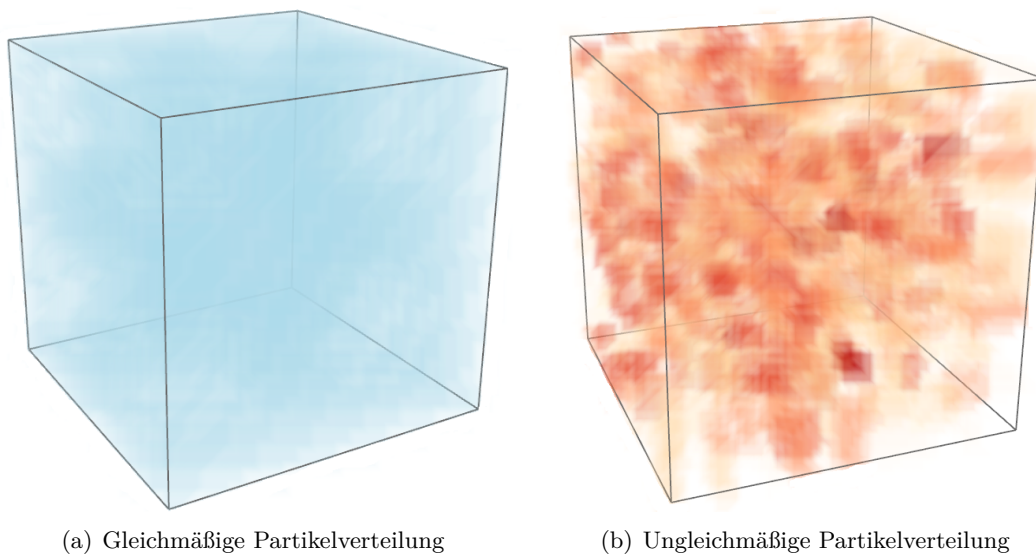


Abbildung 4.7.: Visualisierung der räumlichen Partikelverteilung für zwei Partikelarten mittels Direct Volume Rendering. Während die Partikel in der linken Darstellung gleichmäßig verteilt sind, weist die rechte Darstellung Bereiche mit unterschiedlichen Partikelkonzentrationen auf.

4. Assistierte Präsentation

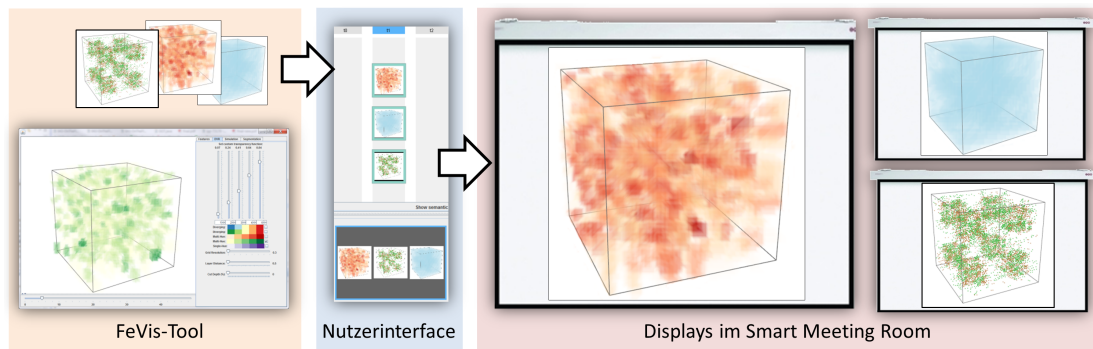


Abbildung 4.8.: Nutzung des FeVis-Tools, um die Rohdaten mittels Direct Volume Rendering darzustellen (links). Die erzeugten Views werden automatisch in den Präsentationsgraph integriert (Mitte) und erscheinen somit auf den Displays des Raums (rechts).

Damit der räumliche Bezug der Daten besser analysiert werden kann, erlaubt es das FeVis-Tool, gezielt solche Regionen im Simulationsraum zu extrahieren, die eine hohe Partikelkonzentration aufweisen. Basierend auf den zuvor gezeigten Views, wählen die Nutzer dazu eine interessante Proteinart und einen passenden Grenzwert für die Partikelkonzentration aus. Diese durch das Tool neu generierten, Feature-basierten Visualisierungen ergänzen anschließend die existierenden Views auf den Displays des Raums (vgl. Abbildung 4.10).

Abbildung 4.9 zeigt eine solche Feature-basierte Visualisierung bei der die interaktiv definierten Regionen von Interesse über offene Ellipsoid-Ikonen dargestellt werden. Die Farbe kennzeichnet hierbei Features, die für zwei unterschiedliche Proteinarten extrahiert wurden. Im Beispiel kann anhand der Ikonen die ungefähre Anzahl, Position, Größe und Ausrichtung der Regionen einfach abgelesen werden. Außerdem erlauben es die Ellipsoide leicht festzustellen, ob Regionen sich überlappen und ob die Verteilung von zwei Proteinarten somit räumlich miteinander korreliert oder ob (wie im Beispiel) sich die Proteine räumlich voneinander trennen.

Durch das Ergänzen der Anzeige mit den neuen Views, kann somit die Proteinverteilung im Raum einerseits detailliert anhand von Darstellungen der Rohdaten und andererseits vereinfacht anhand der Visualisierung von extrahierten Features untersucht werden (vgl. Abbildung 4.10).

Für eine genauere Analyse des Zeitbezugs müssen die bestehenden Views allerdings noch weiter ergänzt werden.

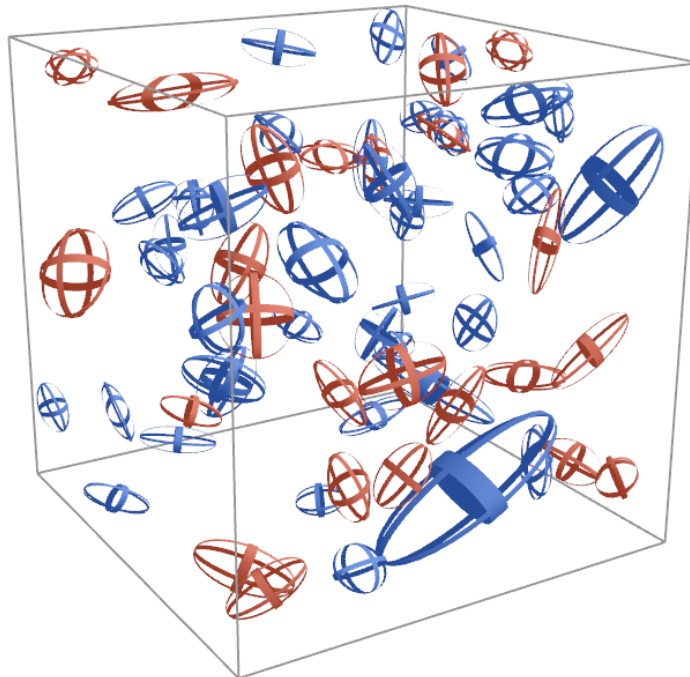


Abbildung 4.9.: Darstellung der Partikelverteilung im Simulationsraum mit Hilfe von Ellipsoid-Ikonen. Die Ellipsoide beschreiben die Größe, Orientierung und Position von Regionen mit hohen Partikelkonzentrationen. Die Farbe kodiert die Partikelart.

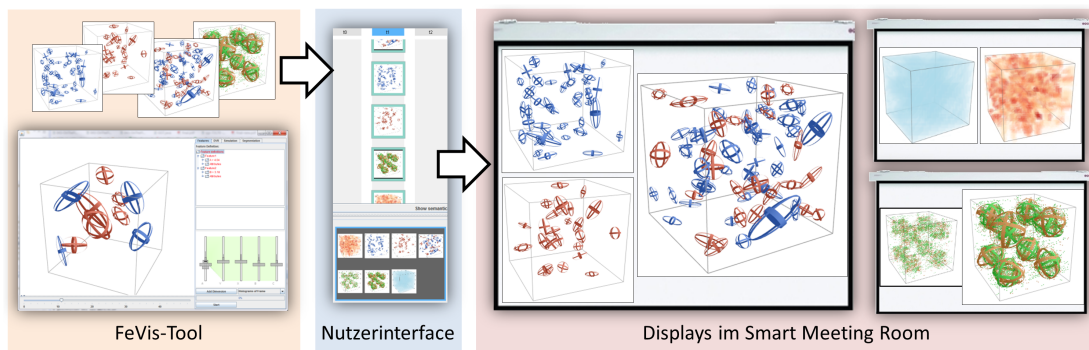
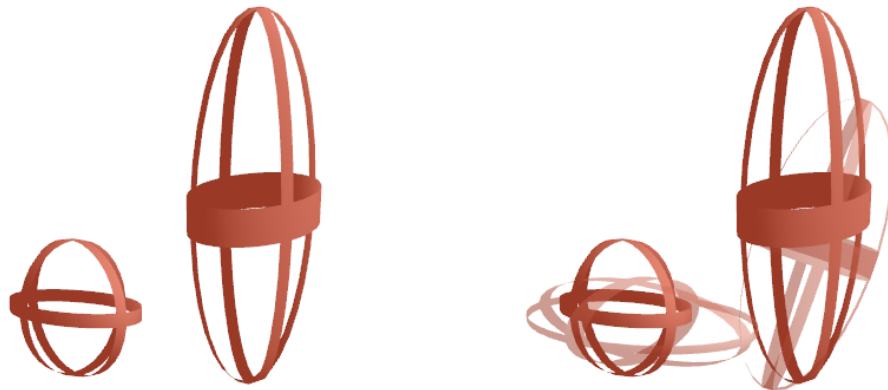


Abbildung 4.10.: Nutzung des FeVis-Tools, um den Raumbezug der Simulationsdaten anhand von Feature-basierten Visualisierungen zu untersuchen (links). Die durch das Tool generierten Views werden automatisch in den Präsentationsgraph eingefügt und ergänzen dort die bereits vorhandenen Views (Mitte). Auf den Displays des Raums erscheinen hierdurch die Feature-basierten Visualisierungen gleichzeitig mit den zuvor (in Abbildung 4.8) gezeigten Darstellungen der Daten (rechts).

4. Assistierte Präsentation



(a) Features aus einem Simulationszeitpunkt

(b) Features aus zwei Simulationszeitpunkten

Abbildung 4.11.: Anzeige von Features aus mehreren Simulationszeitpunkten in einer Darstellung, um räumliche und zeitliche Aspekte der Features in Kombination zu betrachten. In der linken Abbildung sind zwei ausgewählte Features vor dem Einblenden von zusätzlichen Zeitpunkten zu sehen. In der rechten Abbildung werden nach dem Anpassen der View nun zusätzlich auch die zeitlich veränderten Versionen dieser Features dargestellt.

Um die zeitliche Entwicklung einzelner Regionen zu untersuchen, können Nutzer Features aus mehreren unterschiedlichen Zeitpunkten des Simulationsverlaufs gleichzeitig in Views einblenden. Abbildung 4.11 zeigt eine solche Darstellung im Detail, bei der für einen interaktiv anpassbaren Zeitbereich die Features (als Ellipsoide) gleichzeitig eingezeichnet werden. Anhand der so generierbaren Darstellungen kann die Bewegungsrichtung von Features und Größenänderungen von Features einfach abgelesen werden. Das Anpassen der existierenden Views ermöglicht es somit, die räumlichen Eigenschaften zusammen mit der zeitlichen Entwicklung von ausgewählten Features zu analysieren (vgl. Abbildung 4.12). Hierdurch kann das Diffundieren und das Anlagern von Proteinen an bestimmten Bereichen (der biologischen Zelle) untersucht werden.

Die bisher generierten Views zeigen jedoch immer nur einen Teil aller Features. Um den gesamten Simulationsverlauf zu erfassen, sind deshalb noch weitere Übersichtsdarstellungen erforderlich.

Um solch einen Überblick über alle extrahierten Features zu geben, kann das FeVis-Tool nun noch dazu verwendet werden, um einen sog. “Tracking-Graph” zu extrahieren und anzuzeigen. Der Tracking-Graph beinhaltet die Features aus allen Simulationszeitpunkten (als Knoten) und beschreibt die zeitliche Entwicklung der Features über Kantenverbindungen.

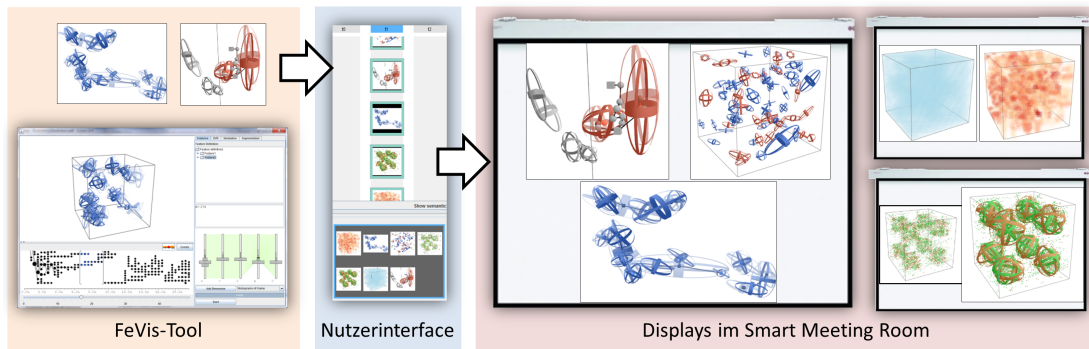


Abbildung 4.12.: Das FeVis-Tool wird dazu verwendet, um bestehende Views (aus Abbildung 4.10) anzupassen, indem Features aus zusätzlichen Zeitpunkten eingeblendet werden (links). Die so veränderten Views werden automatisch aktualisiert und erscheinen somit im Präsentationsgraphen (Mitte) und auf den Displays des Raums (rechts). Anhand der sichtbaren Views kann nun der zeitliche und der räumliche Bezug von ausgewählten Features untersucht werden.

Abbildung 4.13 zeigt eine Darstellung des Tracking-Graphs bei der die Zeit auf die horizontalen Achse abgebildet und alle über den Simulationsverlauf extrahierten Features als Knoten des Graphen gleichzeitig dargestellt werden. Der Durchmesser der Knoten beschreibt die Größe der Features und die Ikonen in den Knoten kennzeichnen Ereignisse, wie das Zerfallen oder Vereinen der extrahierten Regionen von Interesse. Mittels der View kann die zeitliche Entwicklung von mehreren interessanten Regionen über einen großen Zeitabschnitt nachvollzogen werden. Beispielsweise ist in Abbildung 4.13 zu sehen, wie zwei große Regionen mit hohen Proteinkonzentrationen langsam in kleinere Bereiche zerfallen und am Ende der Simulation die beiden mit den Features assoziierten Proteinarten komplett abgebaut wurde.

Weiterhin kann der Tracking-Graph auch in die bestehenden Feature-basierten Views eingeblendet werden, um einen größeren zeitlichen Kontext für die Entwicklung der Regionen zu geben. In Abbildung 4.14 ist beispielsweise zu sehen, wie der Tracking-Graph hierfür die zuvor in Abbildung 4.11 gezeigten Features ergänzt. In dieser Darstellung werden die Knotenpositionen auf die Ellipsoid-Mittelpunkt gesetzt. Anhand der Darstellung in Abbildung 4.14 kann hierdurch beispielsweise abgelesen werden, dass die beiden gezeigten Features sich im weiteren Simulationsverlauf miteinander vereinen werden und, dass sich die resultierende Region anschließend nach oben bewegt.

Abbildung 4.15 zeigt, wie die neu generierten Views aus Abbildung 4.13 und 4.14 die bestehende Anzeige auf den Displays ergänzt.

4. Assistierte Präsentation

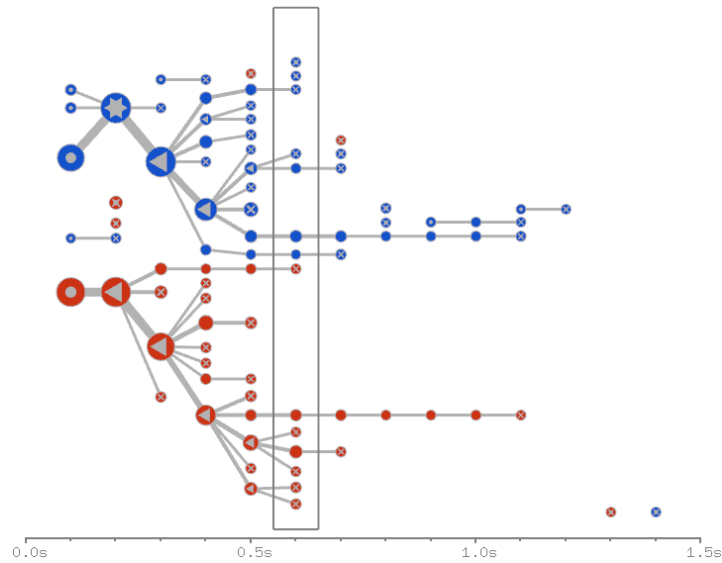


Abbildung 4.13.: Darstellung des Tracking-Graphs zur Veranschaulichung der zeitlichen Entwicklung von extrahierten Features. In der Zeichnung ist die Zeit auf die horizontale Achse abgebildet, die Knotengröße kodiert das Volumen der Features und die Ikonen in den Knoten kennzeichnen besondere Ereignisse in der zeitlichen Entwicklung der Regionen von Interesse.

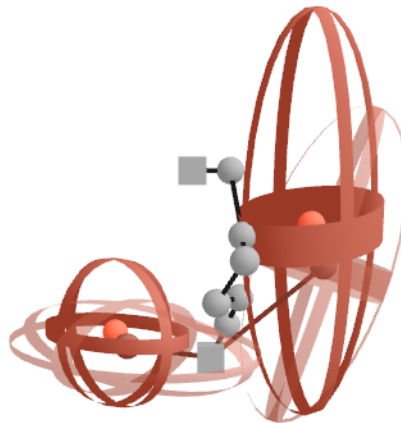


Abbildung 4.14.: Darstellung bei der zu den Ellipsoiden aus zwei Zeitschritten (vgl. Abbildung 4.11(b)) zusätzlich der Tracking-Graph eingeblendet wird. Der Graph zeigt die Feature-Positionen für 8 zusätzliche Simulationszeitpunkte. Hierdurch bietet er einen erweiterten zeitlichen Kontext, um die zukünftige Entwicklung der ausgewählten Features zu untersuchen.

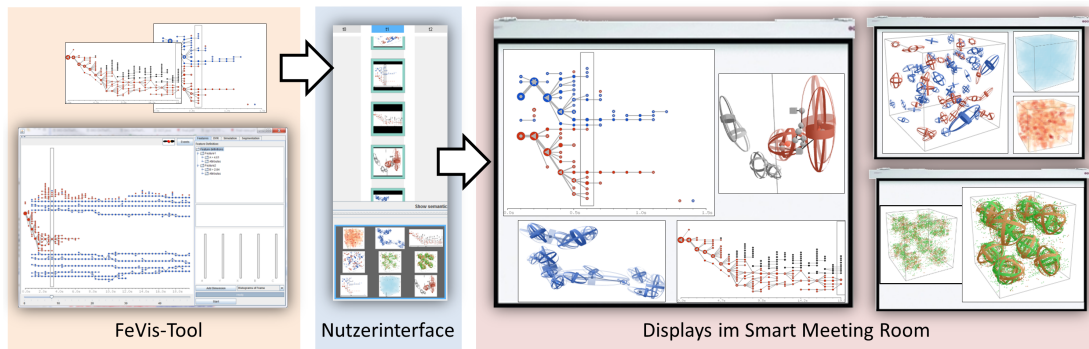


Abbildung 4.15.: Nutzung des FeVis-Tool um Übersichtsdarstellungen von allen extrahierten Features zu generieren und als Views bereitzustellen (links). Diese neuen Views werden automatisch in die Präsentation integriert (Mitte) und erscheinen somit zusammen mit den anderen Views auf den Displays des Raums (rechts).

Die auf den Displays gleichzeitig angezeigten Views erlauben es, den Simulationsverlauf zu untersuchen, indem sie sowohl die räumlichen Eigenschaften als auch die zeitliche Entwicklung der Proteinverteilung zeigen und hierfür Visualisierungen mit einem unterschiedlich starken Abstraktionsgrad verwenden (vgl. Abbildung 4.15). Während bei dem zuvor in Abschnitt 4.3.1 vorgestellten Lupen-Tool es jedoch noch relativ einfach war, einen Zusammenhang zwischen den multiplen View herzustellen (weil das in den Views gezeigte Bildmaterial nicht verändert wurde), ist dies bei den Views des FeVis-Tools nicht mehr der Fall. Hier kann dieselbe Region von Interesse in unterschiedlichen Views mal mittels Direct Volume Rendering, mal als Ellipsoid im Raum und mal als ein Knoten im Trackinggraph dargestellt werden. Um diesen inhaltlichen Zusammenhang zwischen den Views für den Nutzer zu kommunizieren, sind die Views (optional) miteinander verlinkt. Das bedeutet, dass wenn beispielsweise das selbe Feature in unterschiedlichen Views vorkommt, es bei Selektion in einer View auch in allen anderen Views hervorgehoben wird. Dies erlaubt es zum Beispiel, den räumlichen Bezug des Features in einer View abzulesen und gleichzeitig dessen zeitliche Entwicklung anhand einer anderen View einzuordnen.

Ein anderes Beispiel für die Verlinkung der Views betrifft die Definition von interessanten Eigenschaften, die über die Features visualisiert werden. Wird diese Definition angepasst (z.B. weil der Grenzwert für eine "hohe" Partikelkonzentration neu festgelegt wird), so werden automatisch auch die extrahierten Features in alle Views, welche diese Definition verwenden, entsprechend aktualisiert. Durch das beschriebene Vorgehen wird das Arbeiten mit mehreren Views vereinfacht.

Das FeVis-Tool kann auf unterschiedlichen persönlichen Geräten gleichzeitig aufgerufen werden, um verschiedenen Nutzern das Anpassen der Views zu ermöglichen. Somit kann nicht nur ein einzelner Nutzer den Explorationsvorgang steuern, sondern alle Nut-

4. Assistierte Präsentation

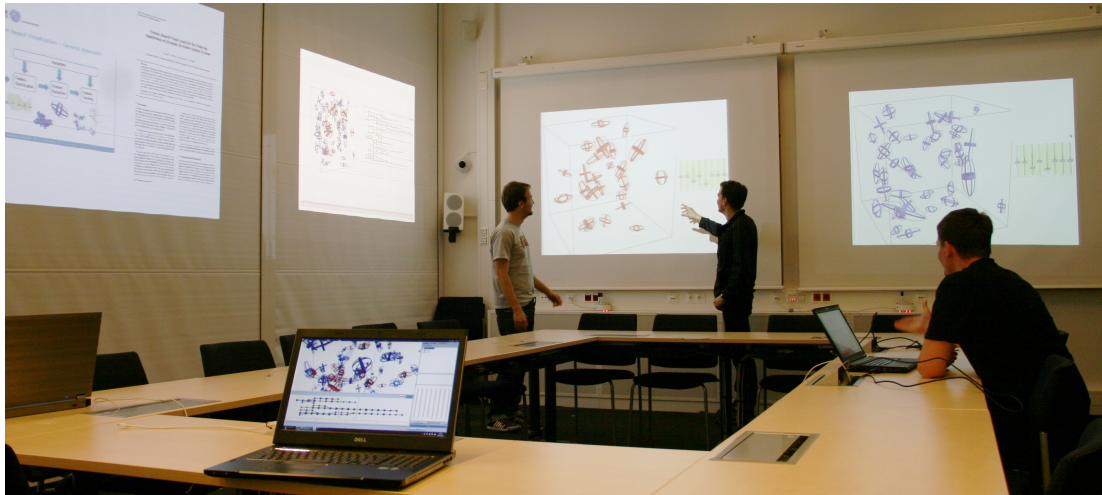


Abbildung 4.16.: Feature-basierte visuelle Analyse in einem Smart Meeting Room. Mehrere Displays werden dazu verwendet, um verschiedene Features gleichzeitig anzuzeigen. Zwei Nutzer diskutieren die Datenvisualisierung auf den Displays des Raums, während ein dritter Nutzer gerade sein persönliches Gerät zur Anpassung der Visualisierungen verwendet.

zer können an der Exploration teilnehmen, indem sie beispielsweise selber neue Features extrahieren und in die bestehenden Darstellungen einfügen. Hierdurch wird es möglich, dass unterschiedliche Experten sich auf verschiedene Teilbereiche der Exploration konzentrieren. Abbildung 4.16 zeigt ein Foto von einer solchen Situation im Smart Lab von Rostock bei der mehrere Views auf den Displays angezeigt und gemeinsam in der Gruppe diskutiert werden.

Mit dem vorgestellten Anwendungsfall wurde ein Beispiel dafür gegeben, wie durch die Verwendung von mehreren, neu generierten und durch die Nutzer anpassbaren Views, die Exploration von komplexen Daten in einer Multi-Display-Umgebung unterstützt werden kann. Weil die zur Analyse notwendigen Visualisierungen nicht im Vorhinein vorbereitet werden konnten, wurde ein (einzelnes) Tool dazu verwendet, um komplett neue Visualisierungen der Daten bedarfsgerecht während der Exploration zu erzeugen. Im Folgenden wird nun ein Ansatz vorgestellt, der diese Idee weiterführt und hierfür auch das spontane Einsetzen von zusätzlichen Visualisierungstools (aus dem Internet) erlaubt.

4.3.3. Web-basiertes Bereitstellen neuer Inhalte

Das Internet stellt eine große Anzahl an Inhalten bereit, die eine Präsentation ggf. passend ergänzen können. Techniken wie HTML5, JavaScript oder Flash erlauben es,

für deren Anzeige nicht nur statische Webseiten zu verwenden, sondern ermöglichen es auch, interaktive Visualisierungswerkzeuge online verfügbar zu machen. Eine aktuelle Übersicht über solche web-basierten Visualisierungen und die hierbei eingesetzten Techniken, ist in [Mwa+16] zu finden.

Um diese zusätzlichen Inhalte und Online-Tools nun auch für die Analyse in Multi-Display-Umgebungen nutzen zu können, wurde ein Ansatz zur Integration von web-basierten Inhalten entworfen.

Hierfür wurde ein sogenanntes “WebTool” entwickelt, das über das Nutzerinterface auf den persönlichen Geräten mit der Präsentation verbunden wird. Das WebTool stellt eine Verbindung zu dem lokal auf dem Gerät installierten Webbrowser her und nutzt diesen dazu, um Inhalte aus dem Internet abzurufen und in neuen Views darzustellen. Die so generierten Views werden (automatisch) auf den Displays des Raums angezeigt und erlauben es somit, auch Internetseiten und webbasierte Visualisierungstools in einer Multi-Display-Umgebung zu nutzen. Von den persönlichen Geräten aus können die Nutzer hierfür mit den Views interagieren, um beispielsweise einem Hyperlink auf einer Internetseite zu folgen oder um ein abgerufenes Online-Tool zu bedienen.

Als konkreter Anwendungsfall für die Verwendung von web-basierten Inhalten wurde eine Podiumsdiskussion über Klimadaten genauer betrachtet [Eic+15]. Hierbei beantworteten Experten die spontan gestellten Fragen aus einem breiten Publikum. Alle hierfür benötigten Visualisierungen (wie z.B. Klimakarten oder Temperaturkurven) können in der Regel nicht im Vorhinein vorbereitet werden. Stattdessen ist es sinnvoll, die entsprechenden Inhalte bei Bedarf aus verschiedenen Quellen abzurufen. Hierfür kommen verschiedene interaktive Online-Portale ^{4,5,6} zum Einsatz, die es ermöglichen, gezielt auf die Fragen des Auditoriums abgestimmte Views zu erzeugen und anzuzeigen.

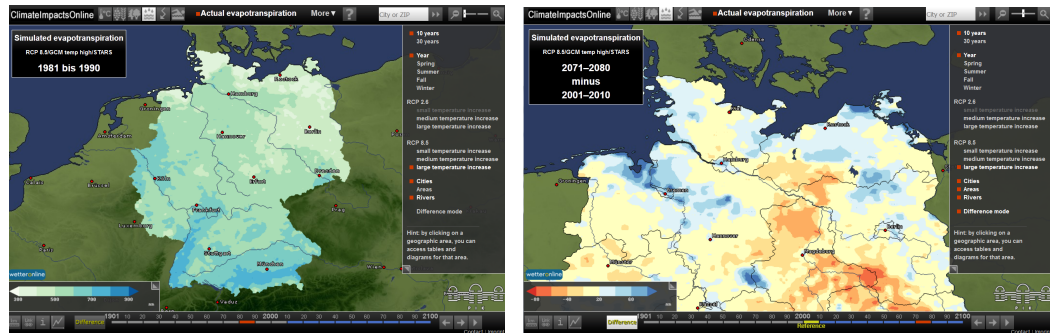
Abbildung 4.17 zeigt verschiedene Visualisierungen, die von dem Webportal *ClimateImpact-Online* abgerufen und interaktiv angepasst werden können. Eine konkrete Frage während der Podiumsdiskussion könnte beispielsweise die Auswirkungen des Klimawandels für Deutschland betreffen. Zur Beantwortung der Frage könnte nun ein zur Diskussion eingeladenes Hydrologe beginnen, indem er dieses Online-Portal auf seinem persönlichen Gerät mittels des WebTools aufruft und die gezeigten Darstellungen als neu generierte Views auf den Displays anzeigt. Beispielsweise kann er zunächst eine Übersicht über die derzeitige durchschnittliche Verdunstung in Deutschland geben indem er die Darstellung aus Abbildung 4.17(a) aufruft. Anschließend könnte er die im Rahmen des Klimawandels stattfindende Veränderungen erläutern indem ein weitere View generiert wird, welche die prognostizierte Entwicklung der Verdunstung zeigt (vgl. Abbildung 4.17(b)). Um hierbei die Änderungen in bestimmten Regionen (z.B. Norddeutschland) genauer zu besprechen, verändert er (von seinem persönlichen Gerät aus) im WebTool den sichtbaren Kartenausschnitt und dieser wird dann sofort durch

⁴<http://www.climateimpactsonline.com/>

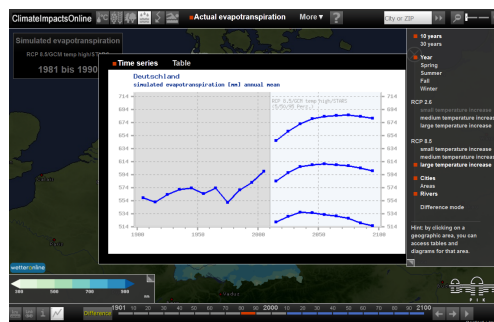
⁵<http://webnet.oecd.org/climatechange/>

⁶<https://www.ipcc.ch/report/ar5/wg2/>

4. Assistierte Präsentation



(a) Kartendarstellung, welche die Verdunstung über Deutschland zeigt. (b) Simulierte Entwicklung der Niederschläge über die nächsten 70 Jahren.



(c) Bei Bedarf abrufbare Niederschlagskurven für ausgewählte Orte.

Abbildung 4.17.: Verschiedene interaktive Visualisierungen von Niederschlagsdaten für Deutschland die über *www.climateimpactsonline.com* bei Bedarf als neu generierte Views angezeigt und angepasst werden können.

den automatischen Layout-Algorithmus auf den Displays für das Publikum sichtbar gemacht. Auf die gleiche Weise können anschließend auch alternative Darstellungen, wie beispielsweise eine Niederschlagskurve für interaktiv ausgewählte Orte, bei Bedarf eingeblendet werden (vgl. Abbildung 4.17(c)).

Ist der erste Experte mit seinen Erläuterungen fertig, kann nun ein anderer Podiumsgast das Beantworten der Fragen übernehmen und hierfür wiederum eigene Views beisteuern, welche die bisherigen Ausführungen ergänzen (vgl. Abbildung 4.18). Um beispielsweise die Frage zum Klimawandel auch aus Sicht der Temperaturentwicklung zu beleuchten, verwendet er Karten und Scatterplots von der Webseite des IPCC⁷, welche die Temperaturentwicklung in Europa zeigen. Diese Darstellungen hat er bereits, während der erste Experte noch beschäftigt war, mit dem Webtool auf der Internetseite ausgewählt und anschließend bei Übernahme des Gesprächs die Anzeige der entsprechenden Views

⁷<https://www.ipcc.ch/report/ar5/wg2/>

4.3. Spontan generierte Views

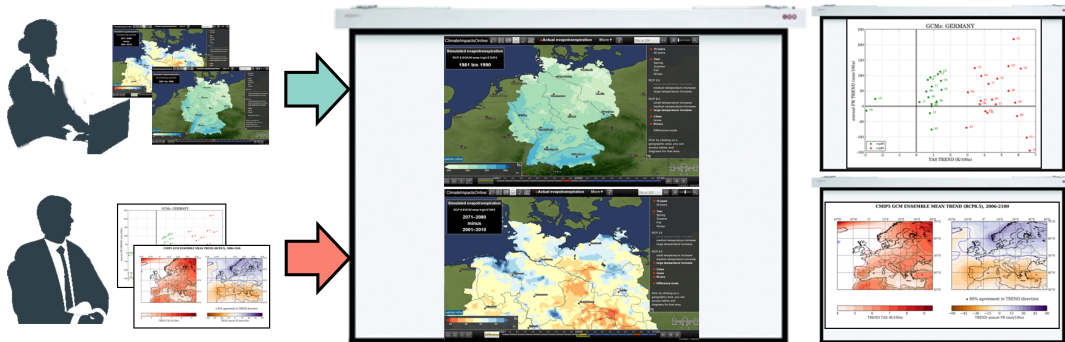


Abbildung 4.18.: Der erste Experte stellt zunächst zwei Views bereit, welche die Niederschlagsentwicklung in Deutschland zeigen. Der zweite Experte ergänzt anschließend die Anzeige mit zwei spontan aus dem letzten IPCC-Bericht ausgewählten Inhalten.

auf den Displays durch Integration in den Präsentationsgraph veranlasst. Durch diese Vorgehen sind anschließend Views auf den Displays zu sehen, welche es den Experten erlauben, sowohl die Entwicklung der Niederschlagsmengen als auch die Temperaturentwicklung in Kombination zu erläutern und somit ein sich ergänzendes Bild von den prognostizierten Folgen des Klimawandels zu zeichnen.

Das beschriebene Beispiel zeigt, wie die Nutzer durch Verwendung des WebTools zur Integration von Online-Inhalten nicht mehr auf die im Content Pool enthaltenen Inhalte beschränkt sind. Stattdessen kann eine Vielzahl von neuen Inhalten spontan aus verschiedenen Quellen im Internet nachgeladen und anschließend automatisch in die Präsentation eingefügt werden.

Hierbei kann insbesondere eine Vielzahl von verschiedenen Online-Tools genutzt werden, um bei Bedarf genau solche Visualisierungen zu erzeugen, die eine aktuelle Situation passend unterstützen. Verschiedene Online-Tools können dabei parallel genutzt und die generierten View zusammen auf den Displays angezeigt werden. Hierdurch wird es den Nutzern ermöglicht, verschiedenartige Analyse-Ansätze heranzuziehen, die jeweiligen Ergebnisse vergleichend gegenüber zu stellen und somit die aus verschiedenen Quellen gewonnene Erkenntnisse zu kombinieren.

Zusammenfassend kann also festgestellt werden, dass die Ansätze zur bedarfsgerechten Generierung und zur interaktiven Anpassung von Views, die Möglichkeiten bei der Präsentation und Exploration von Daten in Multi-Display-Umgebungen umfassend erweitern.

Bei den hier vorgestellten Ansätzen liegt die Verantwortung für das Auswählen und Anzeigen von passenden Inhalten allerdings noch beim Nutzer. Im nächsten Kapitel wird jetzt besprochen, wie der Nutzer hierbei unterstützt werden kann.

5. Assistierte Diskussion

In dem letzten Kapitel stand die Assistenz von Präsentationen in Display Ecologies im Vordergrund. Durch die dort vorgestellten Ansätze werden die Nutzer dabei unterstützt, mehrere Views aus verschiedenen Quellen zusammenzutragen, auf den Displays anzuzeigen und bedarfsgerecht durch neu generierte Views zu ergänzen.

In diesem Kapitel wird nun thematisiert, wie Diskussionen unterstützt werden können. Anders als bei einer Präsentation, macht es eine Diskussion erforderlich, die Anzeige auf den Displays häufig umzukonfigurieren. Beispielsweise ist dies notwendig, um abwechselnd die Diskussionsstandpunkte (von verschiedenen Nutzern) gegenüberzustellen, um einen neu aufkommenden Themenpunkt aus verschiedenen Perspektiven zu betrachten (und anschließend zu den zuvor besprochenen Inhalten zurückzukehren) und um Argumentationslinien an bestimmten Stellen zu bestätigen bzw. zu kritisieren.

Die im vorangegangenen Kapitel entwickelten Ansätze erlauben es bereits prinzipiell, die Anzeige hierfür entsprechend anzupassen, sehen aber gleichzeitig die Verantwortung zum Erfüllen der Aufgaben noch Großteils beim Nutzer. Das bedeutet, dass die Nutzer immer noch selber einzelne Views manuell hinzufügen bzw. wieder entfernen müssen, um die Anzeige zu verändern oder, dass die Nutzer selber geeignete Inhalte für das neue Diskussionsthema finden und für die Anzeige auf den Displays auswählen müssen. Das Ausführen dieser Aufgaben unterbricht den Fluss einer Diskussion und stört die Nutzer dabei, sich auf das eigentliche Diskussionsthema zu konzentrieren.

Im Rahmen der vorliegenden Dissertation wurden deshalb nun zwei neue Ansätze entwickelt, um die Nutzer beim Halten von Diskussionen zu unterstützen.

- Um die Interaktion zum Anpassen der Anzeige zu vereinfachen, wird zunächst einen Überblick über die während einer Diskussion stattfindenden Anpassungen der Anzeige gegeben und somit der Diskussionsverlauf bzw. der Analyseprozess für die Nutzer nachvollziehbar gemacht. Weiterhin wird auf Basis der aufgezeichneten Interaktionen ein einfaches Wiederherstellen von früheren Zuständen der Anzeige ermöglicht, um so bspw. mit geringem manuellem Aufwand zwischen verschiedenen Argumentationslinien zu wechseln oder schnell zu früheren Themenpunkten zurückkehren zu können.
- Die Suche nach semantisch passenden Inhalten wird unterstützt, um das spontane Erstellen und Ergänzen der Anzeige durch neue Views zu erleichtern. Hierfür werden Ansätze zum Editieren und Visualisieren von semantischen Beziehungen

5. Assistierte Diskussion

zwischen Views vorgestellt und dazu verwendet, um während einer Diskussion Empfehlungen zu weiterführenden Inhalten zu geben.

Im folgenden Abschnitt 5.1 wird zunächst die Assistenz der Interaktion besprochen, bevor anschließend in Abschnitt 5.2 die Suche nach Inhalten thematisiert wird.

5.1. Meta-Analyse und Wiederherstellen von Interaktionen

Der in Abschnitt 3.1 eingeführte und in Kapitel 4 verwendete Präsentationsgraph modelliert die Anzeige der Präsentation indem er die Verteilung der Views auf den Displays definiert. Der nun genauer betrachtete Interaktionsgraph beschreibt dagegen, wie der Präsentationsgraph während einer Diskussion (durch die Nutzer) angepasst wird, um die Anzeige zu verändern. Diese Änderungen (wie z.B. das Hinzufügen, Ändern und Entfernen von Views) werden hierbei bezogen auf die interagierende Person, das beeinflusste Display und bezüglich der betroffenen Views protokolliert (vgl. Abschnitt 3.3). Über die Kanten des Interaktionsgraphen werden die protokollierten Änderungen geordnet und somit die während einer Analyse bzw. im Verlauf einer Diskussion auftretenden Interaktionssequenzen beschrieben. Auf diese Weise bietet der Interaktionsgraph eine Erklärung dafür, wie durch eine konkrete Abfolge von einzelnen Interaktionen die Anzeige gestaltet wurde und liefert, bezogen auf die gemeinsame Analyse von Daten, Hintergrundinformationen über die Herkunft, bzw. die Entstehung (englisch: “provenance information”) von einzelnen Views.

In dieser Hinsicht, kann der Interaktionsgraph nun als Ausgangspunkt dafür genutzt werden, um den visuellen Analyseprozess selbst im Rahmen einer Meta-Analyse zu untersuchen. Weiterhin bietet er eine Grundlage dafür, erfolgte Interaktionen nachzuvollziehen und einzelne Anpassungen einfach rückgängig zu machen bzw. wiederherzustellen. Um diese Aufgaben zu erfüllen, müssen jedoch noch folgende Punkte adressiert werden:

- U1:** Eine Visualisierung des Interaktionsgraphen ist notwendig, um die durch Interaktion vorgenommenen Anpassungen zu kommunizieren.
- U2:** Ansätze zum Rückgängigmachen bzw. zum Wiederherstellen von Anpassungen an der Anzeige müssen entwickelt werden.
- U3:** Hilfsmittel für die Analyse des Interaktionsgraphen müssen bereitgestellt werden.

Üblicherweise werden aufgezeichnete Interaktionsfolgen mittels Node-Link-Diagrammen oder (verzweigten) Timeline-Darstellungen visualisiert [DMS14; NC14; Gou+12; Hee+08]. Bei diesen Darstellungen stehen meist die unterschiedlichen Zustände der Anzeige bzw. der Views im Vordergrund. Diese Zustände werden beispielsweise über Thumbnails in den Knoten der Graphdarstellungen oder an den Timelines angezeigt.

5.1. Meta-Analyse und Wiederherstellen von Interaktionen

Einzelne Anpassungen werden dagegen meist nur textuell, über Ikonen oder durch Hervorhebung in den Vorschauansichten der Zustände vermittelt. Weitere Ansätze werden für das Visualisieren, Navigieren und Filtern von Webbrowser-Historien sowie für die Datenanalyse und das interaktive Storytelling eingesetzt [Hee+08; Rag+16]. Für Multi-Display-Umgebungen, in denen mehrere Nutzer mit mehreren Views auf unterschiedlichen Displays interagieren, existieren dagegen nur wenige Ansätze.

Um in Multi-Display-Umgebungen Anpassungen an den Darstellungen einfach rückgängig zu machen (Anforderung U2), verwenden Seifried et al. zwar keine eigenständige Visualisierung der protokollierten Interaktionen (Anforderung U1), bieten aber Wege an, mit denen Nutzer schnell die aufgezeichneten Anpassungen nach den Displays oder nach den Nutzern filtern können. Aus den gefilterten Anpassungen können dann automatisch die jeweils zuletzt protokollierten Anpassungen rückgängig gemacht bzw. wiederhergestellt werden, um die Anzeige mit geringem Aufwand zu verändern [Sei+12]. Durch dieses Vorgehen ist ein nach Displays und Nutzern differenzierendes Zurücknehmen und Wiederherstellen von Anpassungen möglich.

Im Gegensatz zu diesem sequentiellen Undo/Redo-Mechanismus konzentrieren sich Ansätze mit selektiven Undo/Redo-Operationen darauf, die einzelnen Anpassungen (anstelle der Zustände) zu kommunizieren. Solche Undo/Redo-Mechanismen finden meist nicht in Multi-Display-Umgebungen, sondern überwiegend in Text- bzw. Code-Editoren [YM15] und in Programmen für die grafischen Bildbearbeitung und 3D-Modellierung [DMS14] Verwendung. Hierbei werden die einzelnen Interaktionen üblicherweise als Text in Listendarstellungen angezeigt.

Graphdarstellungen, welche (ähnlich wie beim Interaktionsgraphen) darüber hinaus die protokollierten Anpassungen in Pfaden zusammenfassen und zudem (beliebige) selektive Undo/Redo-Operationen zulassen, wurde bereits in älteren Arbeiten beschrieben [Ber94; Men+98]. Diese zumeist theoretischen Konzepte fanden jedoch in letzter Zeit nur wenig und nur vor anderen Anwendungshintergründen [CF07; ZBO13] Beachtung.

Zusammenfassend kann somit festgestellt werden, dass kein Ansatz in der bekannten Literatur geeignet ist, um die Anforderungen U1-U3 gleichzeitig zu erfüllen.

In der vorliegenden Dissertation wurden deshalb neue Ansätze entwickelt, welche die drei Anforderungen adressieren. Diese werden im Folgenden nacheinander vorgestellt.

5.1.1. Visualisierung des Interaktionsgraphen

In der vorliegenden Dissertation wird ein Node-Link-Diagramm für die Visualisierung des Interaktionsgraphen verwendet (vgl. Abbildung 5.1). Die Knoten stellen hierbei die Anpassungen am Präsentationsgraph dar und die Pfade beschreiben verschiedene Zustände der Anzeige (vgl. Abschnitt 3.3.2).

5. Assistierte Diskussion

Jeder Knoten des Interaktionsgraphen enthält verschiedene Arten von Informationen über eine aufgezeichnete Interaktion. Diese werden über die Darstellung der Knoten kommuniziert.

- Jeder Knoten wird als Vorschaubild mit einem farbigen Rahmen dargestellt. Das Vorschaubild zeigt die durch die Interaktion betroffene View und die Rahmenfarbe kennzeichnet den Nutzer, der die Interaktion ausgeführt hat.
- Eine Beschriftung und ein Ikone geben die Art der ausgeführten Anpassung an.
- Zusätzliche textuelle Erklärungen zu einer Anpassung, sind über Ikonen in der linken oberen Ecke der Knoten abrufbar. Diese Erklärungen können entweder manuell durch die Nutzer erstellte Notizen oder automatisch generierte Erläuterungen sein.
 - Manuell hinterlegte Notizen können von Analyseexperten dazu verwendet werden, um den Grund für eine View- bzw. Layout-Anpassung zu vermerken oder um Zustände der View-Anzeige zu kennzeichnen, welche die Ergebnisse der Analyse, die Argumentationslinien oder die Standpunkte von einzelnen Nutzern besonders gut zusammenfassen.
 - Automatisch generierte Anmerkungen werden dagegen von den Tools zur Generierung von Views (vgl. Abschnitt 4.3) bereitgestellt, wenn ein Nutzer eine View anpasst oder neu erzeugt. Beim Lupen-Tool wird beispielsweise der aktualisierte sichtbare Bereich beschrieben, beim FeVis-Tool die verwendete Visualisierungstechnik (inkl. aller Einstellungen) vermerkt und beim WebTool die veränderte URL hinterlegt. Die automatisch generierten Anmerkungen liefern somit eine rudimentäre Beschreibung der neuen Views bzw. der Veränderungen an den Views.

Die Darstellung der Knoten stimmt grundlegend mit der des Präsentationsgraphen überein (vgl. Abschnitt 4.1), stellt jedoch zusätzlich noch Hintergrundinformationen über die Veränderung von Views, die ausgeführten Interaktionen, die Ergebnisse und die Gründe für Veränderungen bereit.

Das Layout des Interaktionsgraphen soll den Fluss der Diskussion wiedergeben. Hierfür werden die Knoten in der Zeichnung so platziert, dass die Interaktionen zeitlich gesehen von links nach rechts geordnet sind. Interaktionen, die eine View zuerst verändert haben, stehen somit ganz links am Anfang eines Pfads und die zuletzt während einer Analyse vorgenommenen Anpassungen ergänzen die Pfade auf der rechten Seite. Der Verlauf einer Diskussion ist somit anhand der horizontal verlaufenden Pfade ablesbar. Derjenige Pfad, der den aktuellen Zustand des aktuellen Präsentationsgraph (und damit auch die Anzeige der Views auf den Displays) beschreibt, ist der aktive Pfad im Interaktionsgraph (vgl. Abschnitt 3.3.2). Dieser Pfad wird in der Darstellung leicht hervorgehoben und durch ein Label gekennzeichnet. Durch Undo-Operationen können auch neue, parallel verlaufende Seitenpfade entstehen, die einen alternativen Analyse-

5.1. Meta-Analyse und Wiederherstellen von Interaktionen

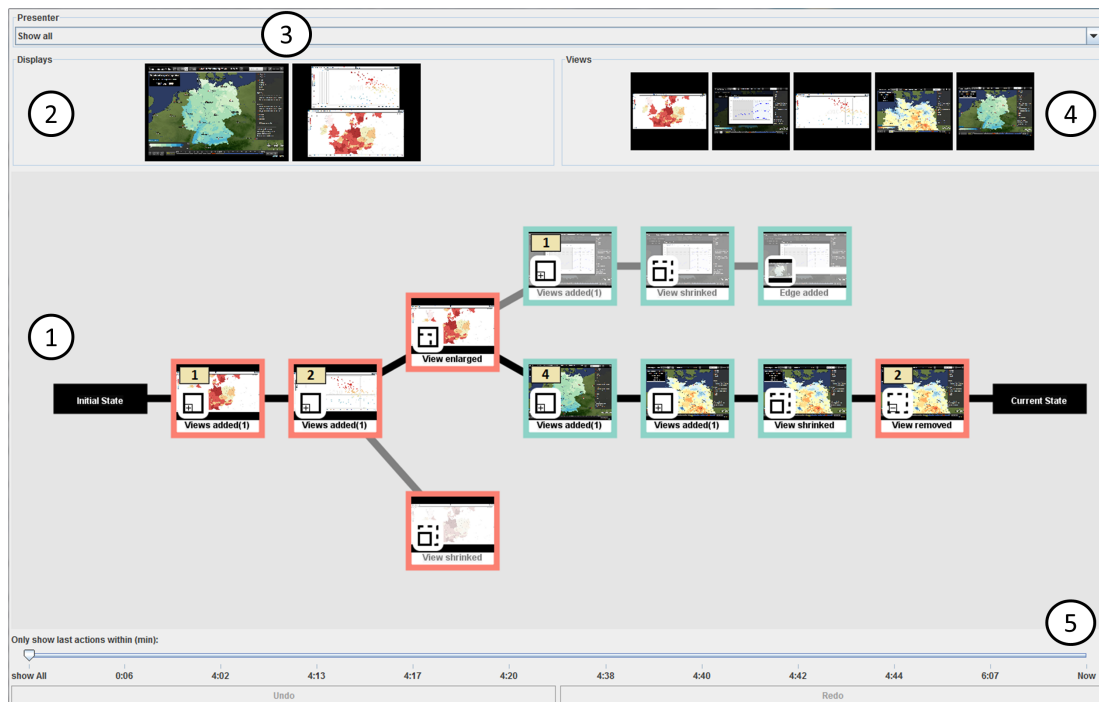


Abbildung 5.1.: Visualisierung des Interaktionsgraphen als Node-Link-Diagramm (1), welches die protokollierten Interaktionen als Knoten darstellt. Für einen Knoten wird der personelle Bezug in der Rahmenfarbe kodiert, der betroffene View als Vorschaubild gezeigt und die Art der Anpassung als Ikone bzw. textuell angezeigt. Weitere Hintergrundinformationen zu einer Anpassung sind als textuelle Anmerkungen über Ikonen in der linken oberen Knotenecke abrufbar. Durch ausgewählte Anpassungen hervorgerufenen Änderungen an den Displays werden über eine Vorschauansicht gezeigt (2). Durch weitere Bedienelemente kann die Darstellung des Graphen bezüglich der betroffenen Displays (2), nach den Nutzern (3), nach den betroffenen Views (4) oder zeitlich (5) gefiltert werden.

vorgang beschreiben. Redo-Operationen wiederum verändern den aktuellen Pfad, indem alternativen Pfade abgerufen (und somit zum aktiven Pfad gemacht) werden.

Neben der Anzeige aller Interaktionen, ergibt es Sinn, auch die Auswirkungen der Anpassungen auf die Anzeige der Views zu kommunizieren. Hierdurch könnte beispielsweise abgelesen werden, wie das Hinzufügen oder Anpassen einer einzelnen View die Gesamtanzeige auf den Displays beeinflusst. Außerdem könnte so der Grund für eine bestimmte Verteilung der Views auf die Displays und für die Art der erstellten Layouts vermittelt werden. Eine Darstellung, die alle Anpassungen für jedes Display zeigt, würde jedoch

5. Assistierte Diskussion

aufgrund der möglicherweise großen Anzahl an Anpassungen (und Displays) schnell unübersichtlich werden.

In der entwickelten Visualisierung werden deshalb Veränderungen an den Layouts und den Views nur für einzelne, von den Nutzern ausgewählte Interaktionen gezeigt. Hierzu werden Übersichtsdarstellungen verwendet, welche die Layouts auf den Displays, sowie alle verwendeten Views übersichtsweise zeigen (vgl. Nummer 2 und 4 in Abbildung 5.1). In diesen Darstellungen werden die durch eine ausgewählte Interaktion veränderten Layouts gezeigt und die hierbei betroffenen Displays und Views nochmal farblich hervorgehoben. Zusammen mit der Visualisierung des Interaktionsgraphen kann somit für jede protokollierte Interaktion gleichzeitig der Display-, der Nutzer- und der Inhalts-Bezug betrachtet und die Veränderungen der Anzeige untersucht werden.

Die Visualisierung des Interaktionsgraphen erlaubt es den Nutzern in der Folge, sich einen schnellen Überblick über den gemeinsamen Analyseprozess bzw. Diskussionsverlauf zu verschaffen. Hierfür werden die in der Modellierung des Interaktionsgraphen enthaltenen Hintergrundinformationen kommuniziert.

Ein aktuelles Paper von Ragan et al. beschreibt die grundsätzlichen Arten und Nutzungsoptionen von Provenance-Informationen [Rag+16]. Die Visualisierung des Interaktionsgraphen zeigt alle Arten und unterstützt bereits vier von sechs Verwendungszwecken:

- Recall: Bewusstsein für den aktuellen und die zuvor durchgeführten Analyseschritte schaffen.
- Collaborative communication: Teilen von Daten, Views und Ergebnissen zwischen verschiedenen Analyse-Experten unterstützen.
- Presentation: Vermitteln des Analyse-Fortschritts und Zusammenfassen der Ergebnisse für außenstehende Personen (wie z.B. Experten aus anderen Anwendungshintergründen oder z.B. Personen aus dem Management).
- Replication: Möglichkeit zum Wiederholen und somit auch zum Überprüfen der Analyse bzw. der einzelnen Analyseschritte.

Darüber hinaus kann die entwickelte Visualisierung des Interaktionsgraphen auch als Basis für die Unterstützung der beiden verbleibenden Einsatzzwecke von Hintergrundinformationen dienen: Meta-Analyse und Action Recovery.

5.1.2. Analyse des Interaktionsgraphen

Um den durch mehrere Nutzer vorangetriebenen Diskussionsverlauf untersuchen zu können, sollte der Interaktionsgraph auf den Displays der Display Ecology dargestellt werden. Darüber hinaus sollte der Graph aber auch für jeden Nutzer über die persönlichen Geräte abrufbar sein, um ein individuell gesteuertes Filtern der Anpassungen und ein Verfolgen bzw. Nachvollziehen von ausgesuchten Diskussionszweigen zu erlauben.

5.1. Meta-Analyse und Wiederherstellen von Interaktionen

Das Filtern von Informationen ist hierbei eine entscheidende Operation, welche den Analyseprozess unterstützen kann, indem weniger wichtige Informationen ausgeblendet und dafür potentiell interessante Informationen exklusiv angezeigt werden. Dies ist besonders bei großen Datenmengen eine wichtige Funktion. Bei längeren Analyseprozessen mit mehreren Nutzern, die laufend die Anzeige auf unterschiedlichen Displays verändern, kann die Anzahl der aufgezeichneten Interaktionen schnell sehr groß und unübersichtlich werden. Um im Rahmen einer Meta-Analyse nun auch größere Interaktionsgraphen untersuchen zu können, ergibt es somit Sinn, wenn die protokollierten Anpassungen nach verschiedenen Aspekten gefiltert werden könnten. Hierbei kommt das Filtern, nach einem ausgesuchten Nutzer, einer betroffenen View, einem beeinflussten Display, dem Zeitpunkt oder nach der Art der Anpassung infrage.

Abbildung 5.1 zeigt wie diese Filtermöglichkeiten unterstützt werden. Über eine Selektionsbox im oberen Teil von Abbildung 5.1 kann bezüglich eines bestimmten Nutzers gefiltert werden, während die Zeitleiste am unteren Rand das zeitliche Filtern ermöglicht. Die Vorschauansicht der Displays im linken oberen Teil wird dazu verwendet, um Displays auszuwählen und der Bereich in der rechten oberen Ecke von Abbildung 5.1 erlaubt es, Views zu selektieren bzw. nach einer bestimmten Art der Anpassung zu filtern.

Das Setzen eines Filters bewirkt, dass alle Interaktionen, die nicht den Filtereinstellungen entsprechen, aus der Zeichnung des Interaktionsgraphen entfernt werden. Um die ursprünglich gezeigte Ordnung zwischen den verbleibenden Knoten zu erhalten und somit auch weiterhin die Zugehörigkeit der nicht gefilterten Anpassungen zu verschiedenen Argumentationslinien zu kommunizieren, werden die Kantenverbindungen in der Zeichnung angepasst. Im gefilterten Interaktionsgraph werden die Knoten über neue Kanten so verbunden, dass die ursprünglichen Pfade nicht unterbrochen werden und somit weiterhin erkennbar bleiben.

Abbildung 5.2 illustriert an einem einfachen Beispiel das Filtern eines im oberen Teil gezeigten Interaktionsgraphen nach drei verschiedenen Kriterien. Nummer 1 in Abbildung 5.2 zeigt das Filtern nach einer ausgewählten View. Der resultierende Graph enthält somit nur Anpassungen, die alle die ausgewählte View betreffen. (Diese View wird somit als Vorschaubild in allen verbleibenden Knoten angezeigt). Bei Nummer 2 von Abbildung 5.2 ist das Filtern nach einem ausgewählten Nutzer zu sehen. Knoten von anderen Nutzern (erkennbar an einer abweichenden Rahmenfarbe) werden hierbei ausgeblendet. Abschließend ist unten Nummer 3 noch das Filtern nach einem Anpassungszeitraum zu sehen. Hierbei werden Knoten am linken Rand der Zeichnung ausgeblendet.

Die bereitgestellten Filter erlauben es, die Anzahl der in der Zeichnung des Interaktionsgraphen enthaltenden Knoten zu reduzieren und somit den Fokus für eine Meta-Analyse gezielt auf bestimmte Teilaspekte zu lenken. Hierfür ist es bereits möglich, Interaktionen differenziert nach den Displays, den Views, den Nutzern oder bezüglich eines Zeitpunkts zu untersuchen.

Neben dem Filtern wird zusätzlich auch das Ein- und Ausklappen von Pfaden im Interaktionsgraphen ermöglicht, um nun auch ausgesuchte Argumentationslinien besser

5. Assistierte Diskussion



Abbildung 5.2.: Einsatz von Filtern, um die Darstellung des Interaktionsgraphen zu vereinfachen. Der Ausgangsgraph für die Beispiele (A) ist oben und die gefilterten Graphen (1-3) sind unten abgebildet. Zu sehen ist (von oben nach unten), das Filtern nach einer ausgewählten View, nach einem einzelnen Nutzer und nach dem Anpassungszeitraum.

5.1. Meta-Analyse und Wiederherstellen von Interaktionen

verfolgen bzw. miteinander vergleichen zu können. Bei der Betrachtung von umfangreichen Analysesitzungen können beispielsweise besonders diejenigen Verläufe von Interesse sein, die viele Anpassungen erfahren haben. Um diese langen Pfade besser untersuchen zu können, kann es hierfür sinnvoll sein, kürzere Pfade, welche Analyseverläufe kennzeichnen, die nach kurzer Zeit verworfen wurden, auszublenden.

Das Zusammen- und Ausklappen von Pfaden kann hierbei mit den zuvor beschriebenen Filteransätzen kombiniert werden, um beispielsweise nur diejenigen Schritte eines untersuchten Analyseverlaufs zu zeigen, die einen bestimmten View betreffen.

Zusammen mit den bereitgestellten Vorschauansichten unterstützt die durch die Nutzer anpassbare Visualisierung des Interaktionsgraphen somit eine umfassende Meta-Analyse von Diskussionen mit mehreren Nutzern, Displays und Views. Hierdurch können Nutzer genau die Interaktionsfolgen untersuchen, die zu einer bestimmten Anzeige auf den Displays geführt haben und werden auf diese Weise beim Nachvollziehen der Diskussion unterstützt. Darüber hinaus wird so aber auch die Grundlage dafür gelegt, um die Anzeige auf den Displays mit geringem manuellen Aufwand anzupassen.

5.1.3. Undo & Redo

Neben dem Untersuchen von aufgezeichneten Diskussionsverläufen spielt auch das direkte Eingreifen in den Analyseprozess eine wichtige Rolle. Hierfür wird der Interaktionsgraph dazu verwendet, um auch den verbleibenden Verwendungszweck von Hintergrundinformationen (vgl. [Rag+16]), nämlich das Rückgängigmachen bzw. Wiederherstellen von Interaktion (action recovery) zu unterstützen.

Die Visualisierung des Interaktionsgraphen wird hierbei dazu verwendet, um mittels Undo/Redo-Operationen zwischen zuvor aufgezeichneten Zuständen der Anzeige wechseln oder einzelne Anpassungen einfach zurücknehmen zu können. Hierdurch wird es einerseits möglich, alternative Vorgehensweisen bei der Analyse auszuprobieren, indem mit geringem Aufwand zu früheren Analyseschritten zurückgekehrt werden kann. Andererseits können so aber auch einzelne, ungewollte Anpassungen (wie sie z.B. durch Bedienungsfehler auftreten können) einfach zurückgenommen werden. Konkret werden hierfür globale und selektive Undo-/Redo-Operationen unterstützt, welche den aktiven Pfad im Interaktionsgraph verändern.

Bei einem globalen Undo/Redo wird der komplette Zustand der Anzeige, mit allen Views auf allen Displays in einen anderen, zuvor erstellten Zustand zurückversetzt. Bezogen auf den Interaktionsgraphen bedeutet dies, dass ein neuer, manuell ausgewählter Pfad zum neuen aktiven Pfad gemacht wird. Abbildung 5.3 zeigt ein Beispiel dafür, wie ein Nutzer die Visualisierung des Interaktionsgraphen dazu verwendet, um einen solchen Zustandswechsel auszuführen. Um einen neuen Zielzustand auszuwählen, bewegt der Nutzer zunächst den Mauszeiger über die Knoten des Interaktionsgraphen. Die Vorschauansicht für die Displays hilft hierbei, den gewünschten Zustand zu identifizieren. Durch Doppelklick auf einen Knoten wird anschließend das globale Undo/Redo

5. Assistierte Diskussion

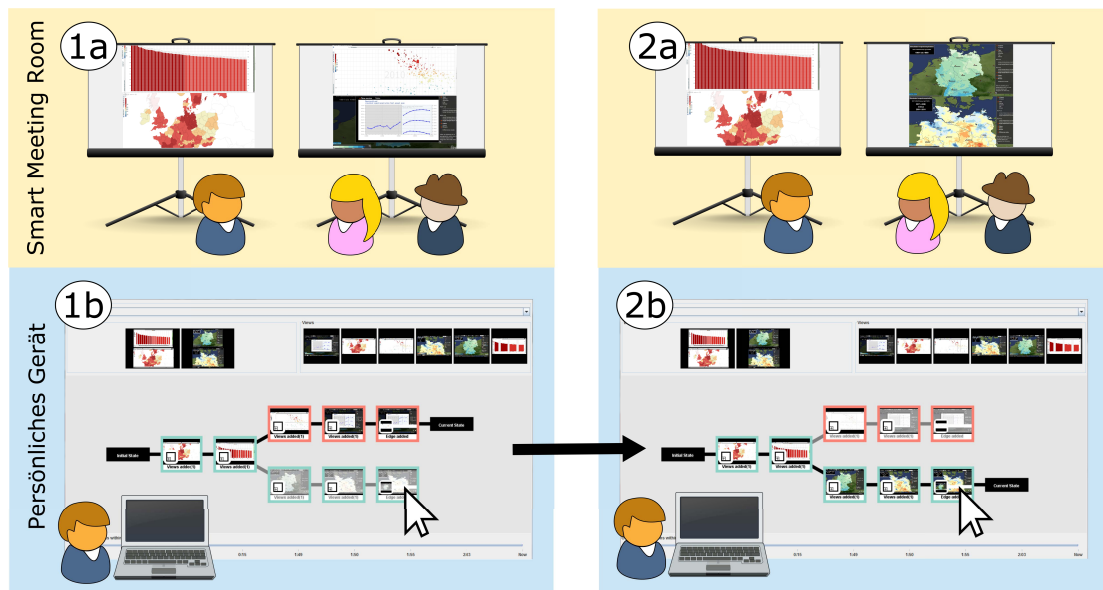


Abbildung 5.3.: Anzeige auf den Displays (a) und Interaktionsgraph (b) vor (links) und nach einem globalem Undo/Redo (rechts). (1b) Durch einen Doppelklick auf einen Knoten (siehe Mauszeiger) wird ein zuvor erstellter Zustand über den Undo/Redo-Mechanismus komplett wiederhergestellt. (2a) Hierdurch wird die Anzeige im Raum entsprechend aktualisiert und (2b) der aktive Pfad im Interaktionsgraphen endet nun bei dem zuvor ausgewählten Knoten.

ausgeführt. Dazu werden automatisch alle Anpassungen des alten aktiven Pfads mittels Undo zurückgenommen und alle Anpassungen auf dem neuen Pfad (der bei dem manuell ausgewählten Knoten endet) mittels Redo wiederhergestellt. Anschließend ist in der Zeichnung des Interaktionsgraphen der manuell ausgewählte Pfad als neuer aktiver Pfad gekennzeichnet und der alte aktive Pfad ist nun ein inaktiver Seitenpfad im Interaktionsgraphen. Auf den Displays des Raums sind die zuvor in der Vorschauansicht gezeigten Views zu sehen.

Bei dem selektiven Undo/Redo werden nicht alle Anpassungen zurückgenommen, sondern bspw. nur solche, die von einem einzelnen Nutzer ausgeführt wurden oder die einen bestimmten View bzw. ein ausgewähltes Display betreffen. Bezüglich des Interaktionsgraphen werden hierzu vom Nutzer selektierte Knoten bei einem Undo aus dem aktiven Pfad entfernt oder umgekehrt bei einem Redo wieder in den aktiven Pfad aufgenommen. Abbildung 5.4 zeigt ein selektives Redo bei dem eine zuvor inaktive Anpassung aus einem Seitenpfad auf den aktuellen Zustand angewendet wird. Im konkreten Beispiel wird ein View, der zuvor in einem alternativen Analyseverlauf angezeigt wurde, nun auch dem aktuellen Verlauf hinzugefügt. Im Ergebnis erscheint der neue View auf den Displays, während die restlichen Views unverändert bleiben.

5.1. Meta-Analyse und Wiederherstellen von Interaktionen

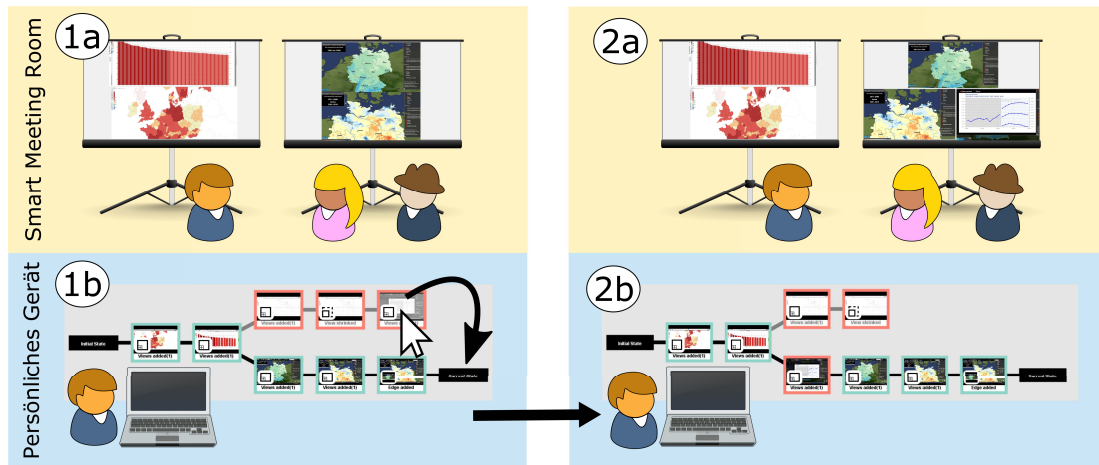


Abbildung 5.4.: Anzeigefunktion (a) und Interaktionsgraph (b) vor (links) und nach einem selektiven Redo (rechts). (1b) Eine View aus einem alternativen Analyseverlauf wird ausgewählt (siehe Mauszeiger) und dem aktuellen Pfad hinzugefügt. (2a) Hierdurch erscheint diese View auf den Displays und (2b) die ausgewählte Anpassung befindet sich nach dem Redo im aktiven Pfad des Interaktionsgraphen.

Das selektive Undo/Redo erlaubt es somit, nicht nur linear Änderungen eines Pfads zurückzunehmen oder wiederherzustellen, sondern auch Ansichten und Analyseschritte von einem Analyseverlauf auf einen anderen Verlauf zu übertragen. Hierdurch wird es möglich, bei Bedarf (Teil-)Ergebnisse aus verschiedenen Analysevorgängen zusammenzuführen und in Kombination anzuzeigen.

Um die Auswahl von passenden Knoten für ein selektives Undo/Redo zu vereinfachen, können die zuvor in Abschnitt 5.1.2 eingeführten Filter genutzt werden. Diese ermöglichen es, Knoten die nicht selektiert werden sollen, auszublenden und somit bspw. schnell die letzten Änderungen an einer View bzw. auf einem Display auszuwählen.

Um weiter auch noch diesen manuellen Aufwand für das Setzen der Filter und das Selektieren der (zeitlich gesehen zuletzt ausgeführten) Anpassungen zu reduzieren, werden diese Aufgaben automatisch übernommen, wenn eine selektive Undo/Redo-Operationen von einem bestimmten Display aus, auf einer angezeigten View oder von einem bestimmten persönlichen Gerät ausgeführt wird. Hierdurch können insbesondere ungewollte Anpassungen differenziert nach Views, Displays und Nutzern (ohne hierfür die Visualisierung des Interaktionsgraphen zu zeigen) schnell zurückgenommen werden.

Eine Besonderheit beim selektiven Undo/Redo ist, dass logische Abhängigkeiten zwischen den einzelnen Anpassungen beachtet werden müssen. Um sicherzustellen, dass solche Abhängigkeiten nicht durch Undo/Redo-Operationen verletzt werden, wird dem Nutzer beim Erstellen einer Knotenauswahl assistiert. Hierfür werden Vor- bzw. Nach-

5. Assistierte Diskussion

bedingungen für selektierte Knoten/Anpassungen automatisch ermittelt und farblich in der Zeichnung des Interaktionsgraphen hervorgehoben, bzw. automatisch der Selektion hinzugefügt. Hierdurch erhält der Nutzer ein visuelles Feedback vor dem Ausführen einer Undo/Redo-Operation und gleichzeitig wird sichergestellt, dass die durch ein selektives Undo/Redo erzeugten Zustände immer auf den Displays angezeigt werden können.

Die in diesem Abschnitt vorgestellten Ansätze adressieren die Anforderungen U1 bis U3, indem Anpassungen an der Anzeige anhand einer Visualisierung des Interaktionsgraphen nachvollzogen, darauf aufbauend der Verlauf einer Diskussion untersucht und die Anzeige mittels Undo/Redo angepasst werden kann. Sie wurde in Zusammenarbeit mit dem Potsdamer Institut für Klimafolgenforschung (PIK) für eine Paneldiskussion über Klimadaten verwendet und mit einem Journal-Artikel publiziert [Eic+15].

Zusammen mit den Ansätzen zur bedarfsgerechten Generierung von Views wurden somit bereits zwei Ansätze vorgestellt, die das Erweitern der Anzeige durch neue bzw. alternative Views erlauben. Im Folgenden werden nun Ansätze vorgestellt, welche die Nutzer beim Abrufen dieser Views unterstützen, indem sie ihm beim Auswählen bzw. beim Suchen nach semantisch passenden Inhalten assistieren.

5.2. Unterstützte Suche nach semantisch passenden Inhalten

Der zuvor vorgestellte Ansatz konzentriert sich darauf, beim schnellen Anpassen der Anzeige zu assistieren, indem der hierfür notwendige manuelle Interaktionsaufwand reduziert wird. Das Verändern der Anzeige ist jedoch auch mit einem gewissen kognitiven Aufwand verbunden. Für das Ergänzen der Anzeige ist es beispielsweise notwendig, dass zunächst eine geeignete View identifiziert bzw. ausgesucht wird, bevor diese überhaupt dem Präsentationsgraph hinzugefügt werden kann. Weil mehrere Nutzer eigene Inhalte bereitstellen und weil neue Views spontan generiert werden können, haben einzelne Nutzer jedoch nicht immer einen Gesamtüberblick von alle hierfür zur Verfügung stehenden Views. Dies erschwert es, während einer Diskussion geeignete Views für eine passende Ergänzung der Anzeige zu finden (und aufzurufen).

Um dieses Problem zu adressieren, werden nun Ansätze vorgestellt, die den Nutzer dabei helfen, einen semantisch geordneten Überblick über alle Views zu bekommen und sie somit bei der Suche nach semantisch passenden Inhalten zu unterstützen.

Die semantischen Beziehungen zwischen den Views werden auf Basis des bereits in Abschnitt 3.2 eingeführten Semantischen Graphen beschrieben. Der Semantische Graph enthält alle verfügbaren Views und beschreibt diese über Knotenattribute. Semantische Ähnlichkeiten zwischen den Views werden aufgrund gleicher Knotenattribute und basierend auf den Kantenverbindungen zwischen den Knoten abgeleitet (vgl. Abschnitt 3.2.2).

5.2. Unterstützte Suche nach semantisch passenden Inhalten

Um den Nutzer zu unterstützen, sind nun konkret folgende Punkte zu adressieren:

- S1:** Eine Visualisierung des Semantischen Graphen ist notwendig, um dem Nutzer einen Überblick über die verfügbaren Views und deren Beziehungen zueinander zu geben. Hierbei muss einerseits die Ähnlichkeit bzgl. der Graphstruktur und andererseits basierend auf den Attributwerten ablesbar sein. Weiterhin sollten auch Editier-Operationen unterstützt werden, um ggf. einzelne Beziehungen bzw. fehlerhafte Attributwerte manuell korrigieren zu können.
- S2:** Semantische Beziehungen zwischen den Views müssen visualisiert werden, um das Navigieren und das Anpassen der Anzeige in Abhängigkeit vom aktuellen Diskussionsverlauf zu erleichtern.

In der Literatur existiert eine umfangreiche Auswahl an verschiedenen Ansätzen zur Visualisieren von attributierten Graphen [HSS15]. Das gleichzeitige Vermitteln der Graphstruktur und der Attribute benötigt in der Regel eine besondere Art der Visualisierung [KPW14]. Ein weit verbreiteter Ansatz hierfür besteht darin, Node-Link-Diagramme mit Attribut-basierten Layouts zu verwenden [SA06; Bez+10; Rod+11; EW14]. In solchen Layouts werden, für ein oder zwei ausgewählte Attribute, die Knoten entsprechend ihrer Attributwerte in der Zeichnung des Graphen platziert. Eine weitere Möglichkeit besteht darin, Attributwerte in unterschiedlichen Knotenrepräsentationen zu kodieren, indem beispielsweise die Knotenfarbe variiert wird [Crn+14] oder Glyphen verwendet werden [JDK10]. Alternativ können auch mehrere Ansichten gleichzeitig verwendet werden, um unterschiedliche Aspekte des Graphen gleichzeitig zu zeigen [TAS09].

Bezüglich des Editierens von Graphen existieren Ansätze, die das Anpassen des Layouts [RRK04; SF08; Ric+12] und der Graphstruktur [BHJ09; Aub+12] erlauben. Das Editieren der Attribute benötigt, bis auf wenige Ausnahmen [GT14], jedoch typischerweise den Wechsel zu separaten Ansichten wie z.B. zu Datentabellen oder Eingabedialogen.

Eine Übersichtsdarstellung für den Semantischen Graphen, welche die Struktur sowie die Attribute (und die hieraus abgeleiteten Ähnlichkeitsbeziehungen) zeigt und zudem auch das Editieren erlaubt, ist somit noch zu entwickeln (Anforderung S1).

Um andererseits Verbindungen oder Zusammenhänge direkt zwischen angezeigten Views zu vermitteln (Anforderung S2), werden diese üblicherweise visuell miteinander verbunden [Mar12]. Hierfür werden typischerweise Highlight-Techniken verwendet, die ähnliche Objekte farblich gleich betonen oder es werden Visual Links eingesetzt, welche die Verbindungen durch zusätzlich eingezeichnete Linienzüge visualisieren [Ste+11]. Visual Links wurden insbesondere bereits eingesetzt, um gleiche Elemente in unterschiedlichen Views (von verschiedenen Anwendungen) miteinander zu verbinden [Wal+10]. Diese, auch in anderen Bereichen weit verbreiteten Techniken, müssen nun weiter modifiziert werden, um auch das Anpassen der Anzeige und das Navigieren über die Views während einer dynamisch verlaufenden Diskussion zu unterstützen.

5. Assistierte Diskussion

Im Folgenden werden die bzgl. der Anforderungen S1 und S2 entwickelten Ansätze vorgestellt. Hierbei wird zunächst auf die Visualisierung des Semantischen Graphen eingegangen, bevor anschließend vorgestellt wird, wie semantische Beziehungen dazu verwendet werden, um die Nutzer beim Anpassen der Anzeige und beim Navigieren zu unterstützen.

5.2.1. Visualisierung des Semantischen Graphen

Bei der Visualisierung des Semantischen Graphen müssen einerseits die Views selbst und andererseits die Beziehungen zwischen den Inhalten vermittelt werden. Gleichzeitig beruhen diese Beziehungen sowohl auf den Kantenverbindungen, also auf der Struktur des Graphen als auch auf der Ähnlichkeit von Knotenattributen. Weiterhin kommt hinzu, dass die Art dieser Knotenattribute nicht einheitlich ist. So können einerseits kategorische Attribute dazu verwendet werden, um beispielsweise das Themengebiet von in den Views dargestellten Inhalten über Schlüsselwörter zu charakterisieren, während andererseits über quantitative Attribute beispielsweise Visualisierungsparameter angegeben werden können, die bei der Generierung von Views verwendet wurden.

Eine einzelne Visualisierung, die alle diese Aspekte des Semantischen Graphen angemessen adressiert, ist nur schwer vorstellbar. Aus diesem Grund werden verschiedene Arten der Visualisierung eingeführt, welche jeweils den Fokus auf unterschiedliche Aspekte der Daten legen. Nutzer können während der Analyse des Semantischen Graphen zwischen den unterschiedlichen Visualisierungsarten wechseln bzw. diese miteinander kombinieren, um bspw. strukturelle und Attribut-basierte Ähnlichkeitsbeziehungen gleichzeitig zu untersuchen.

Als Basisvisualisierung für den semantischen Graph wird ein Node-Link-Diagramm verwendet (vgl. Abbildung 5.5). Die Knoten, welche die Views repräsentieren, werden (analog zur Zeichnung des Präsentationsgraphen bzw. des Interaktionsgraphen) wieder als Rechtecke mit eingebetteten Vorschaubildern dargestellt. Die Darstellung erlaubt es somit, Views, welche bereits von dem Befüllen des Content Pools, vom Editieren des Präsentationsgraphen oder von der Analyse der Interaktionen bekannt sind, auch in dieser neuen Visualisierung wiederzuerkennen. Für das Positionieren der Knoten wird ein Federkraft-Layout verwendet, das Knoten- und Kantenüberlappungen reduziert und somit die einzelnen Vorschaubilder sowie die Struktur des Graphen gut erkennbar macht. Aufgrund der Vorschaubilder können einzelne Views (mit markanten Vorschaubildern) relativ einfach identifiziert werden. Eine einblendbare Lupe erlaubt es weiterhin, auch Details, wie z.B. Texte auf Folien, besser zu erkennen und somit diese bereits vor dem eigentlichen Anzeigen auf den Displays zu überfliegen (vgl. linke Seite in Abbildung 5.5). Obwohl die unmittelbare Nachbarschaft eines Knotens meist noch relativ einfach anhand der gewählten Visualisierung ablesbar ist, kann die erweiterte Nachbarschaft (basierend auf der k-Nachbarschaft) deutlich größer sein und auch Knoten beinhalten, die sich an verschiedenen Stellen der Graphzeichnung befinden. Um deshalb diese Nachbar-

5.2. Unterstützte Suche nach semantisch passenden Inhalten

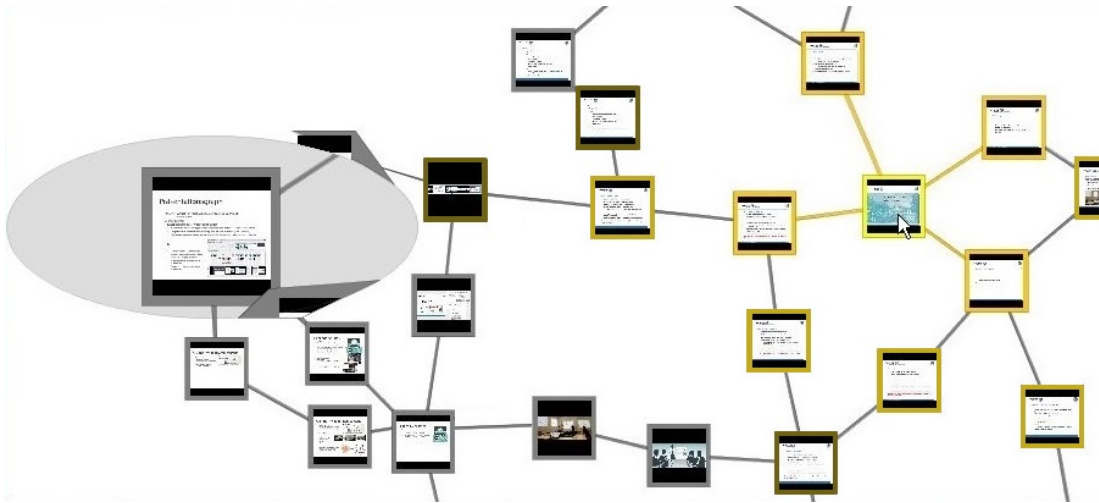


Abbildung 5.5.: Basisvisualisierung für den Semantischen Graphen an einem Beispielsgraph. Eine Lupe ist bei Bedarf verfügbar, um einzelne Inhalte vergrößert darzustellen (links). Die 3-Nachbarschaft für einen ausgewählten Inhalt wird gelb hervorgehoben und zunehmenden Abstand zum ausgewählten Knoten ausgegraut (rechts).

schaft für einen ausgewählten Knoten einfacher identifizierbar zu machen (und damit alle bezüglich der Struktur als ähnlich definierte Views einfach finden zu können), werden die benachbarten Knoten automatisch farblich hervorgehoben. Abbildung 5.5 zeigt dies exemplarisch für die 3-Nachbarschaft eines ausgewählten Knotens. Auf die gleiche Weise lassen sich auch solche Knoten hervorheben, die bezüglich der Attributwerte als ähnlich definierte sind.

Die Basisvisualisierung ermöglicht es somit den Nutzern, sich einen Gesamtüberblick über die verfügbaren Views zu verschaffen und zu einzelnen, manuell ausgewählten Views andere, semantisch passende Views zu finden. Die Darstellung kommuniziert jedoch (noch) nicht die Ausprägungen der einzelnen Attribute bzw. deren Werteverteilung. Hierdurch ist es schwer, Themengebiete zu identifizieren, die über Attribute beschrieben werden, Gruppen von Views zu finden, die ähnliche Attributwerte aufweisen oder Views auszumachen, die schlecht klassifiziert wurden. Um nun auch diese Informationen zu zeigen, wird das Knoten-Layout der Basisvisualisierung angepasst, um auch die Attributwerte für ein (vom Nutzer) ausgewähltes Attribut zu zeigen. Im Folgenden wird zunächst die Visualisierung für kategoriale und anschließend die für quantitative Attribute beschrieben.

Um die kategoriale Attributwerte der Knoten zu zeigen, werden die Knoten neu positioniert. Das hierfür verwendete Layout orientiert sich an den Semantic Substrates von Shneiderman [SA06]. Die grundlegende Idee dabei ist, dass für jeden Attributwert ein separater Bereich, in der Regel ein Rechteck, im Layout festgelegt wird. Anschließend

5. Assistierte Diskussion

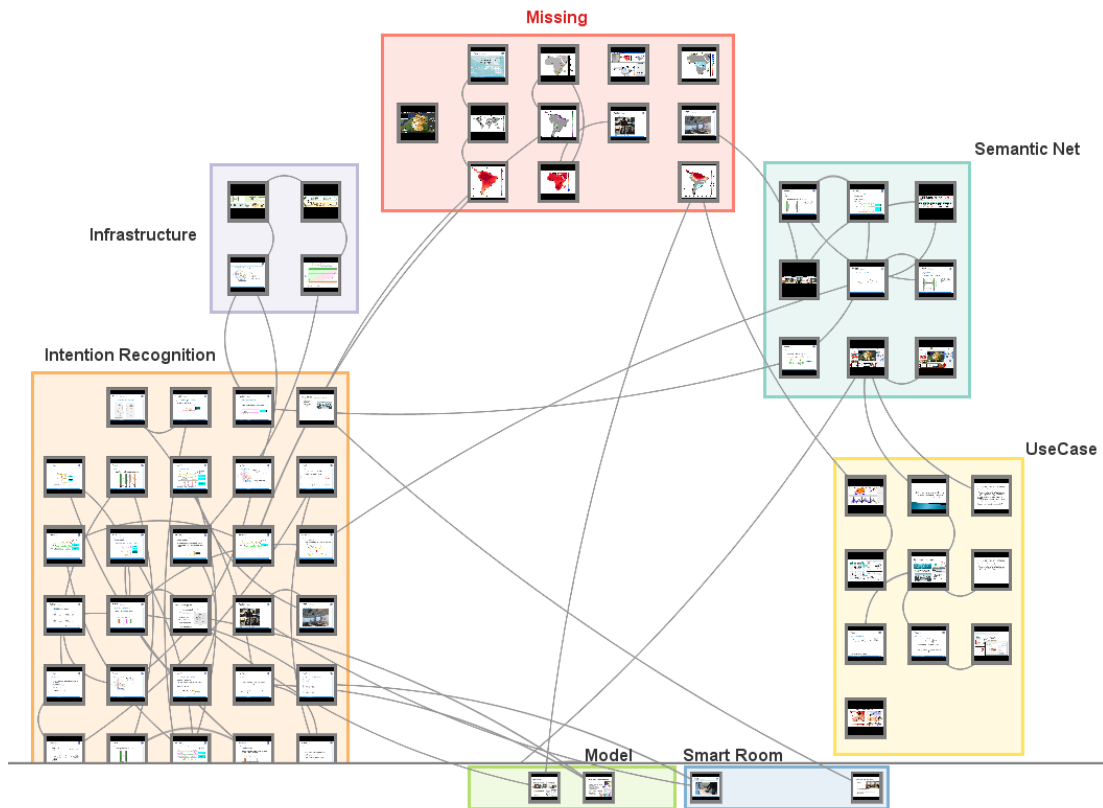


Abbildung 5.6.: Semantischer Graph mit Semantic Substrates [SA06], um kategorische Knotenattribute zu zeigen. Die Zeichnung enthält einen separaten Bereich für Fehlerwerte (oben), vier vollständig sichtbare Bereiche für vier Attributwerte (Mitte) und zwei Visual Cues für Bereiche außerhalb des sichtbaren Ausschnitts (unten).

werden die Knoten gemäß ihrer Attributwerte auf die einzelnen Bereiche verteilt. Hierdurch werden Knoten, die bzgl. des untersuchten Attributs ähnlich zueinander sind, in demselben Bereich zusammengefasst und erscheinen räumlich nahe beieinander.

Abbildung 5.6 zeigt ein solches Layout für eine Foliensammlung bei der die einzelnen Views bezüglich des Attributs “Topic” auf die entsprechenden Rechtecke verteilt wurden. Labels werden verwendet, um den assoziierten Attributwert zu kennzeichnen. Knoten mit undefinierten Attributwerten werden in einem zusätzlichen Bereich gesammelt, der über das Label “Missing” gekennzeichnet ist. Hierdurch sind noch nicht klassifizierte Views schnell zu identifizieren. Die Größe eines Rechtecks wird automatisch an die Anzahl der enthaltenden Knoten angepasst. Auf diese Weise kann die Werteverteilung für ein Attribut anhand der Größe der Bereiche abgeschätzt werden. Auffällig kleine Rechtecke (mit wenigen Knoten) können hierbei auf schlecht klassifizierte Views hindeuten

5.2. Unterstützte Suche nach semantisch passenden Inhalten

oder auf eine semantische Beschreibung, die nicht zu dem restlichen Ordnungsschema (der anderen Nutzer) passt.

Zusätzliche Visual Cues verweisen auf Regionen, die sich gerade außerhalb des sichtbaren Bildausschnitts befinden [Jus+12; FD13]. In Abbildung 5.6 sind am unteren Rand zwei solche Visual Cues für die Bereiche mit dem Attributwert “Model” und “Smart Room” zu sehen. Die Cues vermitteln die Größe und Position der Regionen im Layout und erleichtern somit die attributbasierte Exploration des Semantischen Graphen auch über den angezeigten Bildausschnitt hinaus.

Neben der reinen Analyse des Semantischen Graphen kann es gelegentlich jedoch auch notwendig sein, Attributwerte von Knoten zu korrigieren. Dies ist beispielsweise der Fall, wenn Inhalte falsch klassifiziert wurde, Fehlwerte auftreten oder das Ordnungsschema von unterschiedlichen Nutzern voneinander abweicht. Die entwickelte Visualisierung erlaubt es, das Editieren durch direkte Interaktion mit der Visualisierung vorzunehmen. Hierfür wählt ein Nutzer zu editierende Knoten aus und verschiebt diese per Drag & Drop in andere Bereiche (bzw. auf Visual Cues zu diesen). Hierdurch wird der Attributwert für die verschobenen Knoten geändert, automatisch das Modell des Semantischen Graphen aktualisiert und die abgeleiteten Ähnlichkeitsbeziehungen zwischen den Inhalten neu berechnet. Der Editieransatz wurde zusammen mit Stefan Gladisch entwickelt (vgl. [Gla16]) und ist in einem Journal-Paper [Eic+16] genauer beschrieben.

Für die Visualisierung von quantitativen Attributwerten muss eine weitere Technik bereitgestellt werden. Wir verwenden hierfür eine Scatterplot-Darstellung, bei der die Knotenpositionen die Attributwerte für zwei ausgewählte Attribute kodieren (vgl. Abbildung 5.7). Anders als bei den kategorischen Attributen haben quantitative Attribute typischerweise einen größeren Wertebereich. Für einen einzelnen Attributwert bleibt somit weniger Platz in der Zeichnung und Knoten mit ähnlichen Attributwerten müssen dichter beieinander platziert werden. Um hierbei das Überlagern der Knotendarstellungen zu vermeiden, wurde die klassische Scatterplot-Darstellung durch verschiedene Maßnahmen erweitert:

- Knoten werden standardmäßig ohne Vorschaubilder oder Kanten und nur noch als einfache Punkte im Scatterplot angezeigt. Vorschaubilder und Kantenverbindungen werden standardmäßig nur noch bei Selektion angezeigt. Um trotzdem gewisse Informationen über die Graphstruktur zu vermitteln, wird der Grad eines Knotens¹ in der Knotenfarbe kodiert. Hierdurch sind weiterhin stark bzw. besonders schwach verbundene Views leicht zu identifizieren.
- Räumlich nahe beieinander liegenden Knoten werden zusammengefasst und über Ikonen im Scatterplot repräsentiert. Die Ikonen zeigen die Anzahl an zusammengefassten Knoten und kennzeichnen somit Gruppen von semantisch verwandten

¹Der Grad eines Knotens bezeichnet die Anzahl an Knoten in dessen 1-Nachbarschaft.

5. Assistierte Diskussion

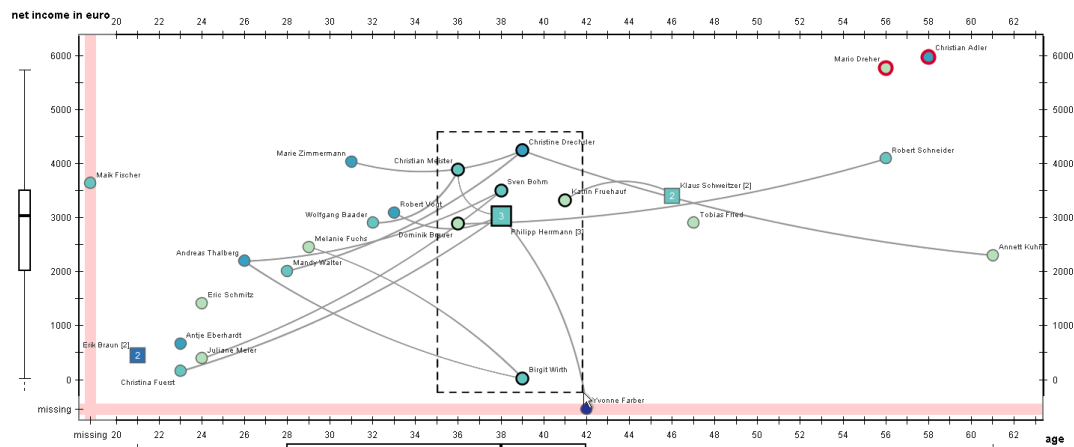


Abbildung 5.7.: Scatterplot-Darstellung für einen attribuierten Semantischen Graph. Kantenverbindungen werden nur für die Knoten in der Selektion angezeigt (Mitte). Die Knotenfärbung gibt den Grad der Knoten an. Dicht beieinander liegende Knoten werden in Gruppen zusammengefasst und als Rechtecke angezeigt. Weitere Kanten, Vorschaubilder der Views und die einzelnen Views in einer Gruppe können bei Bedarf sichtbar gemacht werden.

Views. Bewegt ein Nutzer den Mauszeiger über eine solche Gruppe, werden die in ihr enthaltenden Views sichtbar.

- Eine zuschaltbare Linse ermöglicht es, ausgewählte Bereiche des Scatterplots vergrößert darzustellen und somit die exakte Wertausprägungen und Unterschiede von dicht beieinander liegenden Knoten zu untersuchen. Weiterhin erleichtert die Lupe auch das Editieren der Knotenattribute, weil so das Verschieben von Knoten an andere Stellen des Scatterplots mit einer hohen Präzision möglich ist [Eic+16; Gla16].

Die entwickelten Darstellungen erlauben es Nutzern, einen umfassenden Gesamtüberblick über den Semantischen Graphen zu bekommen, semantisch verwandte Inhalte zu suchen und Gruppen von ähnlichen Inhalten auszumachen. Wahlweise können bei der Visualisierung hierbei entweder Struktur- oder Attribut-basierte Ähnlichkeitsbeziehungen in der Visualisierung berücksichtigt werden. Durch diese Übersichtsdarstellung wird somit Anforderung S1 adressiert. Andererseits ist für konkrete Aufgaben während einer Diskussion, wie z.B. für das spontane Editieren des Präsentationsgraphen, nicht immer ein Gesamtüberblick über alle Inhalte und Beziehungen notwendig. Stattdessen sind in solchen Fällen feiner auf die Situation abgestimmte Ansätze erforderlich, die sich stärker auf den konkreten Editiervorgang eines Nutzers, auf eine bestimmte View oder auf ein ausgesuchtes Display konzentrieren.

5.2.2. Assistenz bei der Suche nach Inhalten

Die Visualisierung des Semantischen Graphen unterstützt die Suche nach ähnlichen Views. Diese Suche ist Teil verschiedener Aufgaben, welche die Nutzer vor oder während einer Diskussion häufig ausführen:

- Zum Vorbereiten bzw. zum Anpassen der Anzeige (über die persönlichen Geräten der Nutzer), müssen passende Views im Content Pool gefunden werden, damit diese anschließend dem Präsentationsgraph hinzugefügt werden können (vgl. Abschnitt 4.1).
- Für ein zeitliches Navigieren über den Präsentationsgraph müssen diejenigen Zeitpunkte ermittelt werden, die gesuchte bzw. geeignete Views enthalten (um diese z.B. für das Beantworten von Fragen erneut aufzurufen).
- Um eine einzelne, auf einem Display gezeigte View möglichst passend zu ergänzen, müssen semantisch verwandte Views auf dem Display schnell abgerufen und einfach in die existierende View-Zusammenstellung integriert werden.

Für die Unterstützung dieser Aufgaben müssen zusätzlich zum Semantischen Graphen jedoch noch weitere Zusammenhänge sichtbar gemacht werden, die im Präsentationsgraph bzw. im Interaktionsgraphen gekapselt sind. Um einen häufigen Wechsel zwischen den verschiedenen Visualisierungen zu vermeiden, können deshalb Beziehungen aus dem Semantischen Graphen in den Repräsentationen vom Content Pool, vom Präsentationsgraph im Nutzerinterface sowie in der Darstellung der Views auf den Displays eingeblendet werden. Damit hierbei trotzdem weiterhin die wesentlichen Elemente der ursprünglichen Visualisierungen erkennen zu können, werden nicht immer alle semantischen Beziehungen zwischen den gerade gezeigten Views visualisiert. Stattdessen werden diese nur bei Bedarf eingeblendet, wenn der Nutzer eine oder mehrere Views auswählt. Nur zu dieser Auswahl werden anschließend semantische Beziehungen angezeigt.

Das der Suche zugrunde liegende Vorgehen lässt sich somit in 5 Schritten zusammenfassen:

1. Ein Nutzer selektiert eine View (über das Nutzerinterface oder direkt auf den Displays).
2. Die selektierte View wird im Semantischen Graphen ermittelt.
3. Basierend auf den Attributen und den Kantenverbindungen des Semantischen Graphen werden (alle) semantisch ähnlichen Views gefunden.
4. Von den semantisch ähnlichen Views werden diejenigen ermittelt, die gerade im Nutzerinterface oder auf den Displays verwendet werden.
5. Die Beziehungen zwischen den im 1. Schritt ausgewählten und den im 4. Schritt gefundenen Views werden visualisiert.

5. Assistierte Diskussion

Da die zu unterstützenden Aufgaben jeweils andere Teile des Nutzerinterfaces betreffen, muss jeweils auch die Visualisierung der ermittelten Beziehungen angepasst werden.

Suche im Content-Pool: Die im Nutzerinterface verwendete Darstellung des Content Pools, fasst mehrere Views, ähnlich einer Ordnerstruktur, in Gruppen zusammen (vgl. Abschnitt 4.1). Die zu einer ausgewählten View semantisch verwandten Views können somit nicht jederzeit sichtbar sein und mehrere der gesuchten Views können sich hinter einem einzelnen Vorschaubild verbergen.

Um Nutzer das Durchsuchen des Content Pools zu erleichtern, werden deshalb Labels auf den Vorschaubildern der Inhaltsgruppen platziert (vgl. oberer Teil in Abbildung 5.8). Mit den Labels wird angegeben, wie viele der in Schritt 4 ermittelten Views sich in der Gruppe befinden. Auf diese Weise können Nutzer schnell solche Gruppen identifizieren (und anschließend durchsuchen), die passende Views haben und bei Vorhandensein mehrerer solcher Gruppen einfacher entscheiden, welche davon besonders vielversprechend sind. Weniger relevante Teile des Content Pools können bei der Suche somit bewusst ausgelassen werden und der Nutzer findet schneller solche Views, die für das Ergänzen der Anzeige geeignet sind.



Abbildung 5.8.: Nutzer-Interface mit eingeblendeten semantischen Informationen zu einem selektierten Inhalt (siehe Cursor). Für den Content Pool werden Labels eingeblendet, welche die Anzahl der semantisch passenden Inhalte in einer Gruppe angeben und somit das Durchsuchen aller verfügbaren Views unterstützen (oben). Für den Präsentationsgraph werden die Beschriftungen der Zeitschritte eingefärbt und Visual Links eingezeichnet (Mitte). In der Layer Bar wird über Pfeile eine Navigationsempfehlung zu ähnlichen Views gegeben (unten).

5.2. Unterstützte Suche nach semantisch passenden Inhalten

Zeitliche Navigation: Bei der zeitlichen Navigation müssen Zeitschritte aufgerufen werden, die semantisch passende Views enthalten. Für diese Aufgabe kann entweder die Zeichnung des Präsentationsgraphen oder die Layer Bar im Nutzerinterface genutzt werden. Problematisch hierbei ist, dass diese Darstellungen immer nur einen kleinen Ausschnitt des zeitlichen Verlaufs (Layer Bar) bzw. nur wenige Views pro Zeitschritt (Darstellung des Präsentationsgraphen) zeigen können.

Um die zeitliche Navigation zu unterstützen, werden deshalb beim Präsentationsgraphen die Beschriftungen der Zeitschritte am oberen Rand der Zeichnung orange hinterlegt, um die Anzahl der semantisch interessanten Views zu kennzeichnen. Ein kräftiger Farbton bedeutet hierbei, dass viele interessante Views auf den Displays angezeigt werden, wenn der jeweilige Zeitschritt aufgerufen wird. Da die Beschriftung (unabhängig von dem gerade gezeigten Graphausschnitt) immer für alle Zeitschritte im Fokus zu sehen ist, ist somit auch dieser Navigationshinweis immer verfügbar und semantisch passende Zeitschritte können direkt durch Anklicken der Zeitschrittbearbeitungen aufgerufen werden.

Anders verhält es sich, wenn semantisch interessante Views zeitlich gesehen außerhalb des sichtbaren Bereichs liegen. Hierbei ist eine Visualisierung erforderlich, die eine visuelle Verbindung zu diesen Offscreen-Elementen herstellt. Gleichzeitig sind bei der Navigation nicht immer nur komplette Zeitschritte, sondern teilweise auch nur einzelne Views gesucht. Aus diesen beiden Gründen werden semantische Verbindungen zwischen den Views nun zusätzlich noch als Visual Links in den Präsentationsgraph eingezeichnet (vgl. Mitte in Abbildung 5.8). Damit sich diese neuen Links gegenüber den bestehenden Kanten abheben, werden die neuen Links als gebogene Linie eingezeichnet und die bestehenden Kanten und Knoten des Präsentationsgraphen werden (während der Suche temporär) desaturiert. Die Visual Links zeigen die semantischen Beziehungen auf View-Basis an und erlauben es, die Anzahl und Richtung von semantisch interessanten Inhalten außerhalb des angezeigten Bereichs abzulesen. Zusätzlich können die Visual Links auch direkt angeklickt werden, um zu den verbundenen Views zu springen.

Bei der Layer Bar muss kommuniziert werden, in welcher Richtung die gesuchten Views liegen. Diese können entweder links vom aktuell gezeigten Zeitschritt liegen, wenn sie zuvor bereits gezeigt wurden oder sie liegen rechts, wenn die Inhalte in zukünftigen Zeitschritten auftauchen. Um eine einfache Navigationshilfe zu geben, werden Pfeile in der Layer Bar eingeblendet (vgl. unten in Abbildung 5.8). Die Größe der Pfeile gibt hierbei die Entfernung zu der nächsten semantisch passenden View an. Anhand dieser Navigationshilfe kann der Nutzer somit entscheiden, ob es zum Aufrufen der gesuchten Inhalte einfacher ist, schnell wenige Zeitschritte sequentiell zu überspringen oder es besser ist, den gesuchten Zeitschritt in der Layer Bar (durch Verschieben des sichtbaren Ausschnitts) zunächst zu suchen und anschließend auf den Displays sichtbar zu machen. Der Farbton der Pfeil-Ikonen gibt dagegen die Gesamtanzahl aller Views in der Vergangenheit bzw. in der Zu-

5. Assistierte Diskussion



Abbildung 5.9.: Anzeige auf den Displays mit zusätzlichen Elementen für die Suche nach semantisch passenden Inhalten. Die Views des aktuellen Zeitschritts sind weiterhin oben links zu sehen (vgl. Nr. 1). Semantisch passende Views werden auf der rechten Seite angezeigt (vgl. Nr. 3) und können von dort direkt der Anzeige auf dem Display hinzugefügt werden. Die aus dem Interface bekannte Layer-Bar wird eingeblendet, um das zeitliche Navigieren zu passenden Views zu ermöglichen (vgl. Nr. 2).

kunft an. Hieran kann der Nutzer die Richtung ablesen, in die es sich lohnt zu navigieren bzw. den sichtbaren Ausschnitt der Layer Bar zu verschieben.

Ergänzung der Views auf den Displays: Um Views direkt auf den Displays durch semantisch verwandte Views zu ergänzen oder zu ersetzen, wird die klassische Anzeige bei Bedarf durch zusätzliche Bedienelemente erweitert. Abbildung 5.9 zeigt die erweiterte Darstellung für ein einzelnes Display. An der rechten Seite werden für die gerade auf dem Display angezeigten Views semantisch verwandte Views eingeblendet. Nutzer können diese direkt anklicken, um sie in die Anzeige aufzunehmen. Im unteren Bereich wird zudem die aus dem persönlichen Interface bekannte Layer Bar eingeblendet, um auch die Navigation zu semantisch verwandten Views direkt über die Displays zu steuern.

Während bei der Ansicht aus Abbildung 5.9 die aus der Präsentation bekannten Layouts und die gesuchten Views noch getrennt nebeneinander angezeigt werden, erlauben es andere Darstellungsvarianten diese Views auch direkt miteinander zu vermischen.

5.2. Unterstützte Suche nach semantisch passenden Inhalten

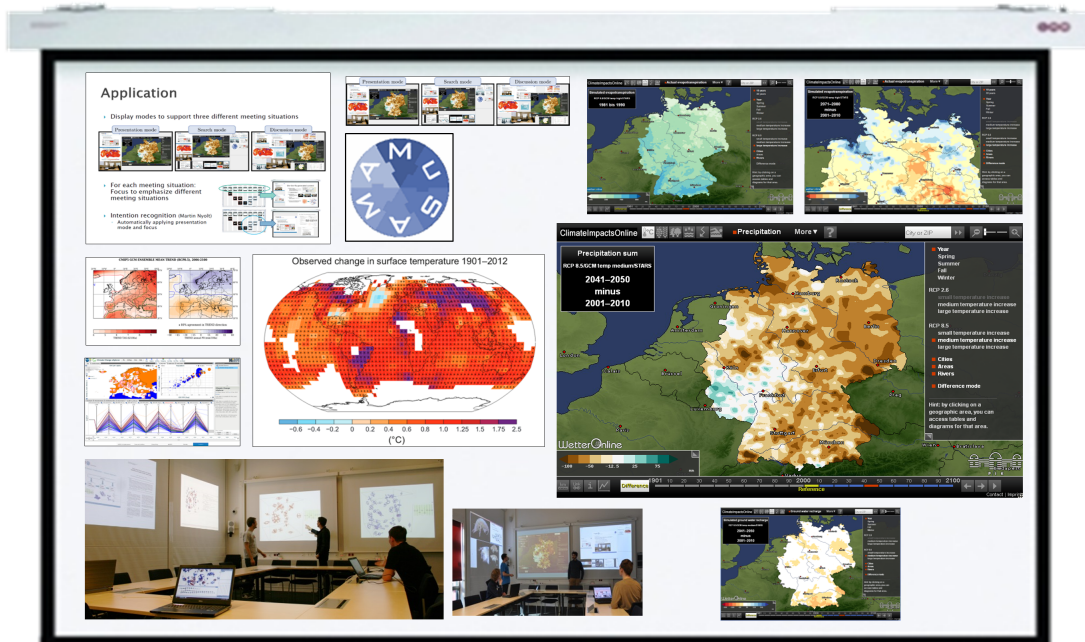


Abbildung 5.10.: Anzeigemodus für die Displays bei dem die Views des aktuellen Zeitschritts zusammen mit den semantisch verwandten Views angezeigt werden. Im Beispiel wurden die gleichen Views wie in Abbildung 5.9 verwendet. Die Views werden nun so angeordnet, dass semantisch verwandte Views Seite an Seite platziert werden.

Abbildung 5.10 zeigt den hierfür entwickelten Modus bei dem um jede View der ursprünglichen Anzeige, die semantischen verwandten Views angezeigt werden. Auf diese Weise wird pro Display die semantische Nachbarschaft der gerade gezeigten Views sichtbar. Auf diese Nachbarschaft kann nun direkt zugegriffen werden, indem Nutzer auf die eingeblendeten Views klicken. Hierdurch werden die ursprünglichen Views auf den Displays entweder ausgetauscht oder ergänzt. Nach Beenden des Suchvorgangs können die zusätzlichen Bedienelemente bzw. die semantische Nachbarschaft wieder ausgeblendet werden.

Durch die beschriebenen Ansätze werden die Nutzer bei solchen Aufgaben unterstützt, die eine gezielte Suche nach semantisch passenden Inhalten erfordern (Anforderung S2). Zusammen mit den zuvor beschriebenen Visualisierungen des Semantischen Graphen, welche einen Überblick über alle Inhalte liefern (Anforderung S1), wird somit der zuvor notwendige Aufwand zum Identifizieren und Einordnen der verschiedenen Views reduziert.

5. Assistierte Diskussion

Die zum Semantischen Graph vorgestellten Ansätze wurden in einer Nutzerstudie im Smart Lab von Rostock und in einem konkreten Anwendungsfall am HIE-RO² untersucht.

Nutzerstudie im Smart Lab

Ziel war es herauszufinden, ob die entwickelte Assistenz die Suche nach semantisch passenden Inhalten erleichtert bzw. die zeitliche Navigation während einer Präsentation unterstützen kann.

Teilnehmer und Versuchsaufbau: An der Fallstudie nahmen 10 Personen (Alter 22-29 Jahre, davon eine Frau und 9 Männer) teil, die entweder Studenten oder Mitarbeiter an der Universität Rostock waren. Von diesen Teilnehmern hatten 6 Personen einen informatikbezogenen Hintergrund. Alle Teilnehmer hatten selbst bereits mehrfach Vorträge mit anderen Präsentationssystemen (außerhalb von Smart Meeting Rooms) vorbereitet und vorgetragen. Drei Personen waren mit dem im Rahmen dieser Dissertation entwickelten System vertraut, kamen aber zuvor noch nicht mit dem Semantischen Graph oder mit den darauf aufbauenden Assistenzansätzen in Berührung.

Für die Bedienung des Systems wurde ein Laptop als persönliches Gerät bereitgestellt auf dem das entwickelte Nutzer-Interface angezeigt wurde. Die Anzeige auf den Displays wurde mit einer Gyro-Maus³ gesteuert, die per Tastendruck die zeitliche Navigation über den Präsentationsverlauf ermöglichte und es durch Gerätebewegungen erlaubte, die auf den Displays gezeigten Views auszuwählen.

Der Content Pool wurde mit 60 unterschiedlichen Views gefüllt, die jeweils unterschiedliche geometrische Figuren zeigten. Semantische Beziehungen zwischen diesen Views wurden vorgegeben, um ähnliche geometrische Eigenschaften, Farben und Größen der in den Views gezeigten Figuren zu beschreiben. Außerdem wurde eine Präsentation mit 24 Zeitschritten als Ausgangsbasis für die Suchaufgaben nach semantisch passenden Views vorgegeben.

Aufgaben und Durchführung: Jeder Studienteilnehmer hat jeweils insgesamt 32 Einzelaufgaben erhalten. Bei jeder dieser Aufgaben ging es darum, ausgehend von einem vorgegebenen Punkt in der Präsentation, eine gesuchte Figur auf einem Display anzuzeigen. Hierbei wurden zwei Aufgabentypen getestet.

- *Navigationsaufgabe:* Die gesuchte View ist bereits in der Präsentation enthalten und muss durch Änderung des angezeigten Zeitschritts auf den Displays sichtbar gemacht werden.

² Hanseatic Institute for Entrepreneurship and Regional Development an der Universität Rostock

³ Bezeichnung des verwendeten Geräts: "Mele F10 Fly Air Mouse (Wireless)"

5.2. Unterstützte Suche nach semantisch passenden Inhalten

- *Anpassen der Anzeige:* Die gesuchte View ist nicht in der vorgegeben Präsentation enthalten und muss der aktuellen View-Zusammenstellung auf den Displays hinzugefügt werden.

Jedem Teilnehmer wurden 18 Navigations- und 14 Anpassungsaufgaben zugewiesen. Die Aufgaben wurden in drei Serien bearbeitet bei denen jeweils andere Hilfsmittel eingesetzt wurden:

1. Für die erste Aufgabenserie durfte nur das entwickelte Nutzer-Interface auf dem Laptop ohne entwickelte Hilfsmittel verwendet werden. Die Messwerte dieser Serie dienten als Referenz für die beiden anderen Aufgabenserien.
2. Im zweiten Teil wurde ebenfalls das Nutzer-Interface verwendet, zusätzlich aber die entwickelten Navigationshinweise, Übersichtsdarstellungen und View-Empfehlungen für semantisch passende Views eingeblendet.
3. Bei der dritten Serie wurden zusätzliche Bedienelemente und View-Empfehlungen auf den Displays angezeigt und die Nutzer sollten diese dazu verwenden, um die gesuchten Views aufzurufen. Das Interface auf dem persönlichen Gerät durfte dagegen nicht verwendet werden.

Jeder Studienteilnehmer hat eine Einweisung bekommen und vor jeder Serie eine 5-10 minütige Probezeit erhalten, um sich mit den Konzepten und der Bedienung vertraut zu machen.

Für die Auswertung wurden die Zeiten gemessen, die benötigt wurden, um mit bzw. ohne das Einblenden von View-Empfehlungen und Navigationshinweisen die vorgegebenen Views zu finden. Nach dem Verlesen einer neuen Aufgabe startete hierbei die Zeitmessung ab der ersten registrierten Eingabe durch ein angeschlossenes Gerät (z.B. bei der ersten Bewegung des Mauszeigers) und endete sobald die gesuchte View auf dem Display erschien.

Resultate: Für die Auswertung werden die Messergebnisse nach den zwei Aufgabentypen getrennt betrachtet. Das Balkendiagramm in Abbildung 5.11 zeigt die durchschnittlich benötigten Zeiten für die Navigationsaufgaben. Die Zeiten sind auf der vertikalen Achse abgetragen und je nach verwendeter Assistenz und Navigationsentfernung weiter auf der horizontalen Achse aufgeschlüsselt. Eine Navigationsentfernung von 4 bedeutet hierbei beispielsweise, dass die anzuzeigende View bei Beginn der Suche 4 Zeitschritte entfernt war.

Bei der Betrachtung der Zeiten wird deutlich, dass die im Interface eingeblendeten Navigationsempfehlungen besonders dann einen Vorteil bieten, wenn gesuchte Views weiter vom aktuellen Zeitschritt entfernt sind und sich somit nicht mehr im sichtbaren Bereich der Layer Bar befinden. Die Navigationszeiten bei der Verwendung der Displays fallen dagegen kaum besser aus als bei der Nutzung des Interfaces ohne Assistenz. Hierbei ist jedoch zu beachten, dass in den gemessenen Zeiten ein eventuell notwendiger Gerätewechsel nicht enthalten ist. Befindet sich

5. Assistierte Diskussion

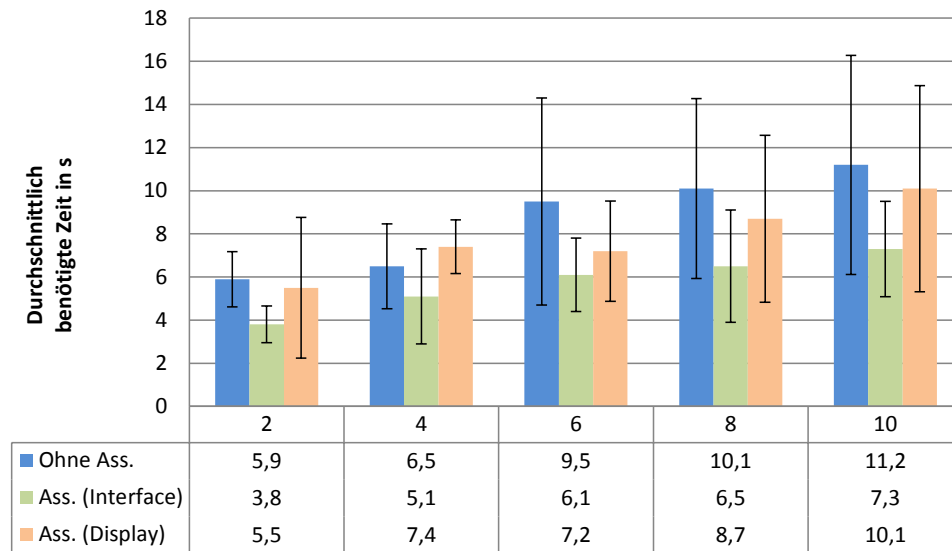


Abbildung 5.11.: Balkendiagramm, das die durchschnittlich benötigte Zeit (in Sekunden) für das Finden einer View mittels Navigation zeigt. Die Farbe der Balken kennzeichnet die drei Aufgabenreihen, welche keine bzw. unterschiedliche Arten von Assistenz verwenden. Weiterhin werden (auf der horizontalen Achse) die Aufgaben nach der minimal notwendigen Navigationsentfernung unterteilt. Die Fehlerbalken zeigen die Standardabweichung an.

ein Nutzer räumlich nahe bei einem Display und ist sein persönliches Gerät an einer anderen Stelle im Raum, kann ein Zuschalten der Navigationsassistenz auf den Displays somit trotzdem hilfreich sein.

Für den zweiten Aufgabentyp sind die ermittelten Werte in Abbildung 5.12 dargestellt. Zu sehen sind die durchschnittlich gemessenen Zeiten für das Anpassen der aktuellen Anzeige durch zuvor nicht verwendete Views. Die Zeiten werden auf der vertikalen Achse abgetragen und nach den drei Aufgabenreihen unterteilt. Zusätzlich wird danach differenziert wie weit die gesuchten Views im Semantischen Graph von den aktuell angezeigten Views entfernt waren. Ein Wert von 2 bedeutet hierbei beispielsweise, dass sich die gesuchte View in der 2-Nachbarschaft befand.

Bei der Auswertung der Zeiten fällt auf, dass insbesondere das Empfehlen von passenden Views auf den Displays die Suche stark beschleunigen kann. Ist eine View semantisch weniger stark mit den angezeigten Views verknüpft, können die Übersichtsdarstellungen im Nutzer-Interface jedoch bessere Ergebnisse liefern.

5.2. Unterstützte Suche nach semantisch passenden Inhalten

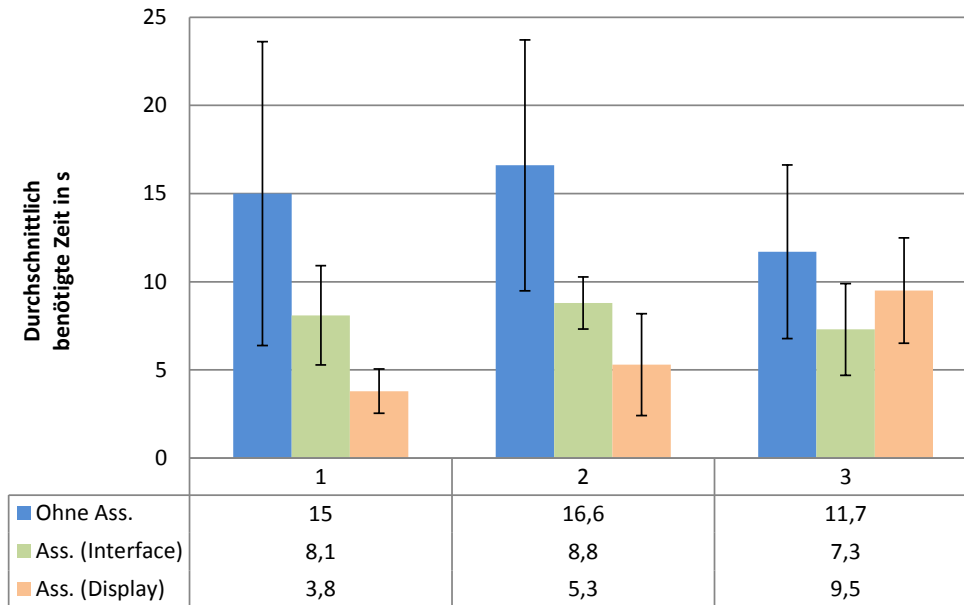


Abbildung 5.12.: Balkendiagramm, das die durchschnittlich benötigte Zeit (in Sekunden) für das Finden und Anzeigen einer zuvor nicht verwendeten View angibt. Die Farbe der Balken kennzeichnet die drei Aufgabenserien, welche keine bzw. unterschiedliche Arten von Assistenz verwenden. Weiterhin werden (auf der horizontalen Achse) die Aufgaben nach dem Abstand der gesuchten View zu den aktuell angezeigten Views (im Semantischen Graph) unterteilt. Die Fehlerbalken zeigen die Standardabweichung an.

Die Ergebnisse der Studie zeigen, dass die entwickelten Assistenzansätze die Suche nach semantisch passenden Inhalten teilweise deutlich beschleunigen können. Weiterhin wird aber auch deutlich, dass je nach Teilnehmer und Aufgabe die gemessenen Zeiten stark variieren können. Diese Variationen können u.a. mit dem relativ klar strukturierten Semantischen Graph im Versuchsaufbau, der individuell unterschiedlich schnellen Gewöhnung an das neuartige Nutzer-Interface sowie mit der teilweise ungewohnten Verwendung der Gyro-Maus zusammenhängen. Langzeituntersuchungen in anwendungsnahen Diskussionsszenarien sind deshalb noch notwendig, um das volle Ausmaß der Unterstützung genauer quantifizieren zu können (vgl. Ausblick in Abschnitt 7.2). Um die praktische Verwendung des Semantischen Graphen weiter zu erproben, wurde in Zusammenarbeit mit dem HIE-RO ein konkreter Anwendungsfall genauer betrachtet.

Anwendungsfall im HIE-RO

Der Aufbau und die Verwendung des Semantischen Graphen wurden in Zusammenarbeit mit anderen Fachbereichen entworfen. Insbesondere mit der Forschungsgruppe der Sprach- und Kommunikationswissenschaft am HIE-RO Institut wurde in einem zyklisch stattfindenden Seminar mit Studenten und Wissenschaftlern des Instituts getestet, ob und wie sich der Graph für das Untersuchen von Textpassagen verwenden lässt. Hierbei wurden Ausschnitte aus verschiedenen Reden bzw. Parlamentssitzungen ausländischer Regierungen und Abgeordneter sowie aus den deutschsprachigen Übersetzungen von diesen, betrachtet. Ziel der Wissenschaftler war es, durch die Suche im Semantischen Graph solche Reden zu identifizieren und für Untersuchungen anzuzeigen, bei denen ein ähnliches Vokabular verwendet wurde bzw. solche Zusammenhänge ausfindig zu machen, bei denen von diesem Vokabular abgewichen wurde.

Im Laufe des Seminars wurden hierzu insbesondere die in Abschnitt 5.2.1 vorgestellten Übersichtsdarstellungen des Semantischen Graphen eingesetzt und weiterentwickelt. Durch Kantenverbindungen konnten einerseits solche Reden in Zusammenhang gesetzt werden, die sich inhaltlich aufeinander beziehen bzw. die in einem historischen Kontext zueinander stehen. Auf der anderen Seite konnte über die Attribute bspw. das in einem konkreten Textausschnitt betroffene Themengebiet eingeordnet werden. Hierbei wurde insbesondere die interaktive Editierfunktion für kategorische Attribute positiv hervorgehoben. Diese wurde dazu verwendet, um die bis zu 100 Textpassagen zunächst grob zu sortieren. Anschließend wurde durch das interaktive Hinzufügen von neuen Bereichen bzw. Attributwerten diese anfängliche Einordnung mit der Zeit verfeinert. Während der Diskussion wurden dem Graph außerdem auch komplett neue Attribute hinzugefügt, um alternative Ordnungsschemata zu erstellen oder andere Untersuchungsschwerpunkte zu setzen.

Im Ergebnis hat sich hierbei herausgestellt, dass die entwickelten Visualisierungen für den Semantischen Graph nicht nur im Kontext der Präsentation von Textausschnitten hilfreich sind, sondern auch bei der Analyse der Inhaltssammlungen unterstützen können.

Zusammenfassend kann festgestellt werden, dass die in diesem Kapitel vorgestellten Ansätze, die zuvor in Kapitel 4 eingeführten Konzepte weiter ergänzen, indem der notwendige manuelle Interaktionsaufwand zum Anpassen der Anzeige weiter reduziert und Assistenz zum Finden von geeigneten Anpassungen gegeben wird. So muss der Nutzer nicht mehr zwingend manuell den Präsentationsgraph editieren, um die Anzeige auf den Displays zu verändern. Stattdessen kann er nun anhand der Visualisierung des Interaktionsgraphen unterschiedliche Zustände bzw. Argumentationslinien einer Diskussion einfach identifizieren und mittels des entwickelten Undo/Redo-Mechanismus schnell zwischen diesen wechseln, um eine sich dynamisch entwickelnden Diskussion zu rekonstruieren.

5.2. *Unterstützte Suche nach semantisch passenden Inhalten*

Darüber hinaus erlaubt es die Visualisierung des Semantischen Graphen, einen thematisch geordneten Überblick über alle verfügbaren Views zu bekommen und bei Bedarf semantische Informationen zu editieren. Das bedarfsgerechte Anzeigen der semantischen Beziehungen unterstützt hierbei nun die spontane Suche und das Aufrufen von passenden Inhalten während einer Diskussion.

Obwohl der Grad der Unterstützung somit weiter erhöht wird, bleibt die Verantwortung für die Verwendung der entwickelten Ansätze weiterhin beim Nutzer: Dieser muss sich immer noch aktiv für deren Anwendung entscheiden indem er z.B. selber die neuen Ansichten aufruft. Im nächsten Kapitel wird nochmals einen Schritt weitergegangen, indem nun die Assistenz automatisch abgerufen und ohne weitere Nutzerinteraktion auf eine erkannte Situation abgestimmt wird.

6. Integration in den Smart Meeting Room

Die bisher in dieser Dissertation vorgestellten Ansätze unterstützen mehrere Nutzer bei der Präsentation und der Diskussion ihrer Views in einer Multi-Display-Umgebung. Konkret wird den Nutzern bei der assistierten Präsentation ein vielseitig nutz- und anpassbares Interface auf den persönlichen Geräten bereitgestellt und die Anzeige von ausgesuchten (und spontan generierten) Views wird durch automatisch berechnete Layouts unterstützt (vgl. Kapitel 4). Weiterhin wird durch Assistenz bei der Interaktion der notwendige Aufwand zum Anpassen der Anzeige durch einen Undo/Redo-Mechanismus reduziert und Nutzer werden bei der Suche nach semantisch passenden Inhalten unterstützt (vgl. Kapitel 5).

Obwohl die bisher vorgestellten Ansätze die Nutzer bei der Präsentation und Diskussion entlasten, indem sie einige anfallende (Teil-)Aufgaben automatisch übernehmen, ist weiterhin ein gewisser Konfigurationsaufwand notwendig, um die Assistenz auch auf konkret vorliegende Situationen bzw. Szenarien abzustimmen. Beispielsweise müssen Nutzer weiterhin mit dem Nutzerinterface interagieren, um Vorgaben für Layout-Anpassungen zu machen.

Ziel dieses Kapitels ist es nun, auch für diese, bisher manuell durchgeführten, Konfigurationsaufgaben Assistenz anzubieten. Hierfür werden die bisher entwickelten Ansätze automatisch auf bestimmte Situationen und Szenarien abgestimmt, welche durch die Situations- und Intentionserkennung in einem Smart Meeting Rooms erfasst werden. Hierdurch kann der benötigte manuelle Interaktionsaufwand weiter reduziert und somit der Nutzer besser in den jeweiligen Situationen unterstützt werden.

Im Folgenden werden zunächst die betrachteten Situationen und Szenarien, sowie bekannte Ansätze zu deren Erkennung kurz vorgestellt. Anschließend werden in Abschnitt 6.2 die entwickelten situationsabhängigen Assistenzansätze beschrieben.

6.1. Szenarien und Situationen

6.1.1. Situations- und Intentionserkennung

Die Grundlage für das Erkennen der aktuellen Situation in einem Smart Meeting Room bildet das Erfassen der Nutzeraktivitäten. Diese Nutzeraktivitäten können beispielsweise mittels maschinellem Sehen (engl. Computer Vision) [AR11; Cao+14], durch das

6. Integration in den Smart Meeting Room

Auswerten von Daten von tragbaren Sensoren [LL13] oder durch fest installierte Sensoren [Che+12a] ermittelt werden. Allgemein muss hierbei zwischen dem Erkennen von einfachen, grundlegenden Aktivitäten (z.B. ob ein Nutzer gerade an seinem Platz sitzt, durch den Raum geht oder an einer bestimmten Stelle stehen bleibt) und komplexeren Verhaltensweisen unterschieden werden, welche zusätzlich noch einen Kontext zu den erkannten Aktivitäten bietet.

Das Erkennen von einfachen Aktivitäten bzw. Situationen findet bereits teilweise Verwendung in den bisher beschriebenen Ansätzen. So fließt in die Layout-Berechnung beispielsweise die ermittelte Sichtbarkeit eines Displays ein und somit auch die erkannten Positionen und Blickrichtungen der Nutzer (vgl. Abschnitt 4.2). Ein anderes Beispiel ist das automatische Setzen von Filteroptionen beim Undo/Redo-Mechanismus, wenn ein Nutzer auf einem bestimmten Displays die letzte Anpassung zurücknehmen oder wiederherstellen möchte und hierfür das betroffene Display bzw. der Nutzer automatisch erkannt wird (vgl. Abschnitt 5.1.3).

Neben dem Erkennen dieser einfachen Aktivitäten existieren für Smart Meeting Rooms jedoch noch Ansätze, welche auch komplexere Verhaltensweisen innerhalb von Besprechungsszenarien erkennen können. Typischerweise konzentrieren sich diese Ansätze hierbei auf das Erfassen von Aktivitäten an denen mehrere Nutzer beteiligt sind (multi-user activities). Einige Ansätze [Dai+08; Krü+13] erkennen beispielsweise unterschiedliche Phasen in der Zusammenarbeit der Nutzer oder können den aktuellen Redner in einer Besprechung ausmachen. Andere Ansätze [YZN13] versuchen darüber hinaus, auch feingranulare Interaktionsfolgen zwischen den Nutzern zu erkennen (wie z.B. das Anfordern von Informationen mit anschließender Kommentierung der aufgerufenen Views).

Das grundsätzliche Vorgehen zum Erkennen von Situationen beruht hierbei i.d.R. auf zuvor erstellten formalen Beschreibungen von verschiedenen Situationen. Während der Besprechung werden diese Modelle gegen tatsächlich aufgezeichnete Sensordaten abgeglichen und melden bei einer ausreichend großen Übereinstimmung das (wahrscheinliche) Eintreten einer bestimmten Situation. Im Smart Lab von Rostock werden z.B. CCBM (*computational causal behaviour models*) für die Beschreibung verwendet, die anschließend durch bayessches Filtern ausgewertet werden [Krü16].

Obwohl Situations- und Intentionserkennung aktuell untersuchte Forschungsthemen sind, sind die Fähigkeiten der existierenden Systeme noch stark limitiert, wenn es um das zuverlässige Erkennung von feingranularen Situationen geht [Che+12a; BBF15]. Diese Limitierung hat unter anderem ihre Ursache darin, dass die Sensoraufnahmen oft nicht ausreichend sind, um zwischen ähnlichen Interaktionen genauer differenzieren zu können [Che+12a]. Im vorliegenden Fall könnte beispielsweise noch eine Vorbereitungsphase von einer späteren Präsentation der Views (vor einem Publikum) unterschieden werden, indem man die durch Sensoren ermittelten Nutzerpositionen, die Blickrichtungen und die gemessenen Redezeiten vergleicht. Welche Interaktion ein Nutzer jedoch gerade auf einem persönlichen Gerät ausführt, ob er bspw. neue Inhalte bereitstellt, ob

er den Semantischen Graphen durchsucht oder ob er ein Tool zur View-Generierung bedient, kann durch vorhandene Sensoren bisher nicht erfasst werden.

Um eine bessere Situationserkennung zu ermöglichen, wird, zusätzlich zu den Sensoraufnahmen im Raum, die komplette Interaktion auf dem Nutzerinterface und auf den Displays software-seitig mitgeschnitten und weitergeleitet. Dies ermöglicht es der Situationserkennung, genauer zwischen verschiedenen Arten der Interaktion zu unterscheiden und es wird hierdurch möglich, komplexere Verhaltensweisen und verschiedene Situationen besser zu erkennen. Das Aufzeichnen der Nutzerinteraktionen kann jedoch auf Wunsch über das Nutzerinterface für jeden Nutzer individuell deaktiviert werden, um somit auch die Situationserkennung für diesen Nutzer abzuschalten.

Im nächsten Abschnitt werden solche Situationen genauer betrachtet, die nun einerseits durch die Ansätze für den SMR erkennbar sind und die andererseits auch durch die bisher in dieser Dissertation entwickelten Konzepte unterstützt werden können.

6.1.2. Untersuchte Szenarien und zugehörige Anforderungen

Für eine situationsangepasste Assistenz werden an dieser Stelle 3 grundlegende Arten von Situationen bzw. Szenarien genauer betrachtet:

Vorbereitung, Präsentation und Diskussion: Während einer Besprechung können mehrere unterschiedliche Phasen durchlaufen werden, die jeweils durch Unterschiede bei der Verwendung des Nutzerinterface und anhand der Positionierung der Nutzer im Raum erkannt werden können [Dai+08; Krü+12].

An dieser Stelle wird zwischen dem Vorbereiten bzw. Planen einer Präsentation, dem anschließenden Präsentieren der ausgewählten Views durch einen bzw. mehrere Redner (vor einem Publikum) und dem Führen einer Diskussion (mit mehreren aktiv beteiligten Nutzern) unterschieden.

- Während der Vorbereitungsphase werden in erster Linie Inhalte durch die Nutzer zusammengetragen und der spätere Präsentationsverlauf geplant. Für diese Aufgabe interagieren Nutzer typischerweise mit dem Interface auf ihren jeweiligen persönlichen Geräten, um den Content Pool zu füllen oder den Präsentationsgraph zu verändern.
- Während der Präsentation stellt ein Nutzer die Views einer zuvor erstellten Präsentation einem Publikum vor. Hierzu positioniert er sich typischerweise gegenüber dem Publikum dicht bei den Displays und verändert die gezeigte View-Zusammenstellung indem er nacheinander die vorbereiteten Zeitschritte des Präsentationsgraphen abruft.
- Bei einer Diskussion beteiligen sich dagegen mehrere Nutzer aktiv, indem sich Redebeiträge häufiger abwechseln. Aktualisierungen der View-Zusammenstellung erfolgen zudem nicht mehr primär durch eine zeitliche Navigation

6. Integration in den Smart Meeting Room

über einen geplanten Präsentationsverlauf. Stattdessen werden sie durch das spontane Hinzufügen bzw. Entfernen von Views, durch das Aufrufen zuvor verfolgter Argumentationslinien mittels des Undo/Redo-Mechanismus oder durch das Anfordern von semantisch verwandten Inhalten hervorgerufen.

Die unterschiedlichen Phasen erfordern es, dass die Nutzer jeweils mit anderen Teilen des Nutzerinterfaces interagieren und, dass jeweils andere Funktionalitäten bzw. Visualisierungen genutzt werden. Eine situationsangepasste Assistenz muss somit das entwickelte Nutzerinterface je nach erkannter Phase automatisch anpassen, um den Zugriff auf die verschiedenen Funktionalitäten zu vereinfachen.

Wechsel des Redners: Während einer Diskussion bzw. einer gemeinsamen Präsentation kann der gerade aktive Redner öfter wechseln.

Dieser Wechsel des Redners wird durch Auswerten von Mikrofon-Aufnahmen [Dai+08] oder dadurch erkannt, dass sich ein zuvor aktiver Redner wieder auf seinen Sitzplatz begibt und dafür eine anderer Nutzer aufsteht und sich zu einer günstigen Vortragspositionen im Raum bewegt [Krü+12].

Der neue Redner möchte i.d.R. auch eine eigene Sicht auf die gerade gezeigten Views vorstellen, den Schwerpunkt auf andere Views lenken oder andere, für ihn (räumlich) besser zugängliche Displays verwenden. Dies erfordert es, dass die View-Verteilung und die Layouts auf den Displays angepasst werden, um bestimmte Views stärker hervorzuheben, die Views anders zu gruppieren oder um andere Displays stärker zu nutzen.

Eine hierfür bisher notwendige manuelle Anpassung des Präsentationsgraph verhindert jedoch, dass der neue Redner zügig den Gesprächsfaden aufnehmen kann. Dementsprechend ist eine situationsbedingte Assistenz erforderlich, welche diese Anpassungen der Layouts beim Erkennen eines Rednerwechsels selbstständig vornehmen kann.

Spontane Suche während einer Präsentation: Während einer Präsentation können Situationen vorkommen, an denen Nutzer kurzzeitig neue Views aufrufen müssen und anschließend wieder zu einem vorbereiteten Präsentationsfluss zurückkehren wollen. Konkret wird angenommen, dass dies der Fall ist, wenn folgende Situationen erkannt werden:

- Eine Frage wird aus dem Publikum an einen Redner gerichtet und dieser bewegt sich anschließend zu seinem persönlichen Gerät, um einen passenden View aufzurufen oder zu einem Display, um eine bereits angezeigte View zum Beantworten der Frage zu nutzen.
- Ein verändertes Navigationsverhalten wird detektiert: Anstelle, wie bei einer Präsentation üblich, die vorbereiteten Zeitschritt nacheinander auf den Displays anzuzeigen, schaltet ein Nutzer (auf der Suche nach einer View) schnell durch die verschiedenen Zeitschritte einer Präsentation. Setzt sich dieses Verhalten über einen längeren Zeitraum fort oder wird durch häufige

Richtungsänderungen deutlich, dass der Nutzer nicht genau weiß, wie er den gesuchten View finden kann, wird dies als Anzeichen dafür genommen, dass der Nutzer Hilfe bei der Suche benötigt.

Im Rahmen der situationsbezogenen Unterstützung ist es somit in diesen Fällen sinnvoll, Hilfsmittel für die Suche nach passenden Views automatisch bereitzustellen. Anders als bei dem Wechsel zwischen Präsentations- und Diskussions-Szenario, ist diese Situation jedoch lediglich als eine kurze Unterbrechung anzusehen, bei der nur ein einzelner Nutzer unterstützt werden muss. Entsprechend muss die Assistenz in diesem Fall genauer auf den Nutzer, die Views oder die Displays abgestimmt sein.

Zusammenfassend kann also festgestellt werden, dass für die Unterstützung der beschriebenen Situationen Folgendes erforderlich ist:

- I1:** Eine automatische Anpassung vom Interface ist notwendig, um den Zugriff auf die in unterschiedlichen Szenarien benötigten Funktionen zu erleichtern.
- I2:** Eine (temporäre) automatische Anpassung der View-Verteilung bzw. der Layouts auf den Displays ist erforderlich, um den Rednerwechsel zu unterstützen. Hierbei muss der aktuelle Redner, müssen seine ggf. bereitgestellten Views und die für ihn zugänglichen Displays berücksichtigt werden.
- I3:** Das (kurzzeitige) Einblenden von semantischen Informationen muss automatisch erfolgen, um einzelne Nutzer bei Suchaufgabe zu unterstützen.

Im nächsten Abschnitt wird nun vorgestellt, wie die bisher vorgestellten Konzepte dazu verwendet werden, um diese Anforderungen zu unterstützen.

6.2. Situationsangepasste Assistenz

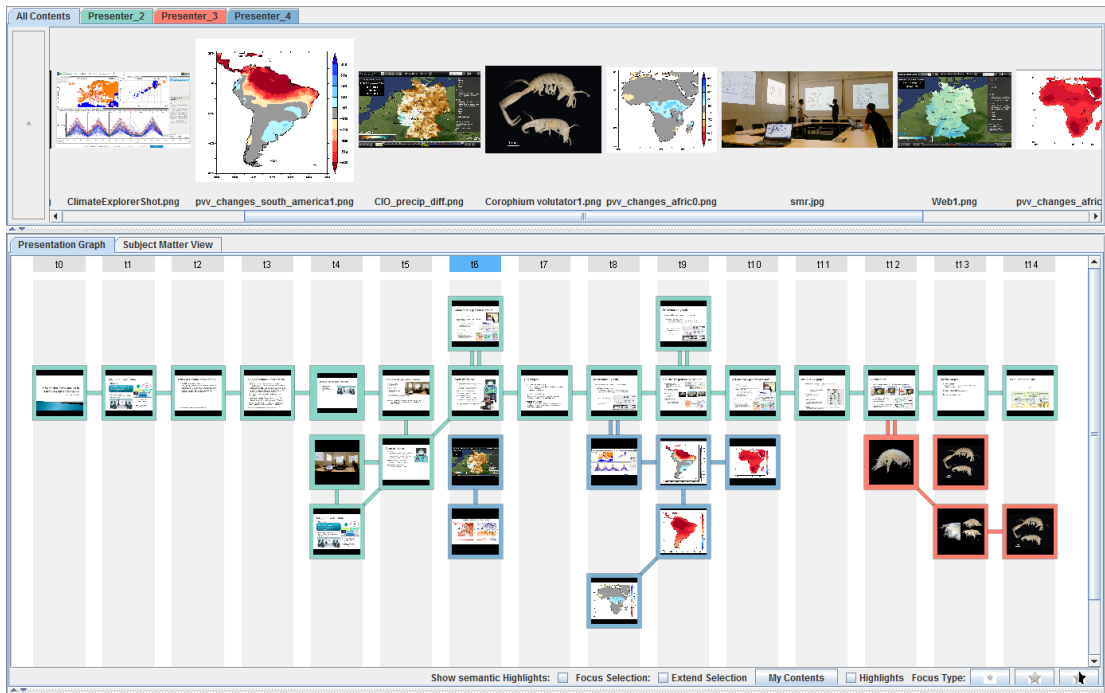
Die zur Situationserkennung verwendeten Ansätze liefern für die zuvor beschriebenen Situationen probabilistische Eintrittswahrscheinlichkeiten. Da eine falsch erkannte Situation und eine darauf ausgerichtete Assistenz im schlimmsten Fall als störend empfunden werden kann, werden die im Folgenden vorgestellten Assistenzansätze nur sehr zurückhaltend eingesetzt. Das bedeutet, dass sie nur dann im vollen Umfang angewendet werden, wenn eine Situation mit einer sehr hohen Wahrscheinlichkeit richtig erkannt wurde. Um jedoch auch bei partiell erkannten Situationen oder bei leicht höheren Unsicherheiten Assistenz anbieten zu können, verfügen die Ansätze über mehrere Abstufungen, die je nach Sicherheit automatisch hinzugeschaltet werden. In einem umfassenden Übersichtsartikel beschreiben Liao et al. hierfür u.a. Formeln mit denen der angemessene Einsatz von Assistenz abgeschätzt werden kann [Lia+06].

6.2.1. Automatische Interface-Anpassungen

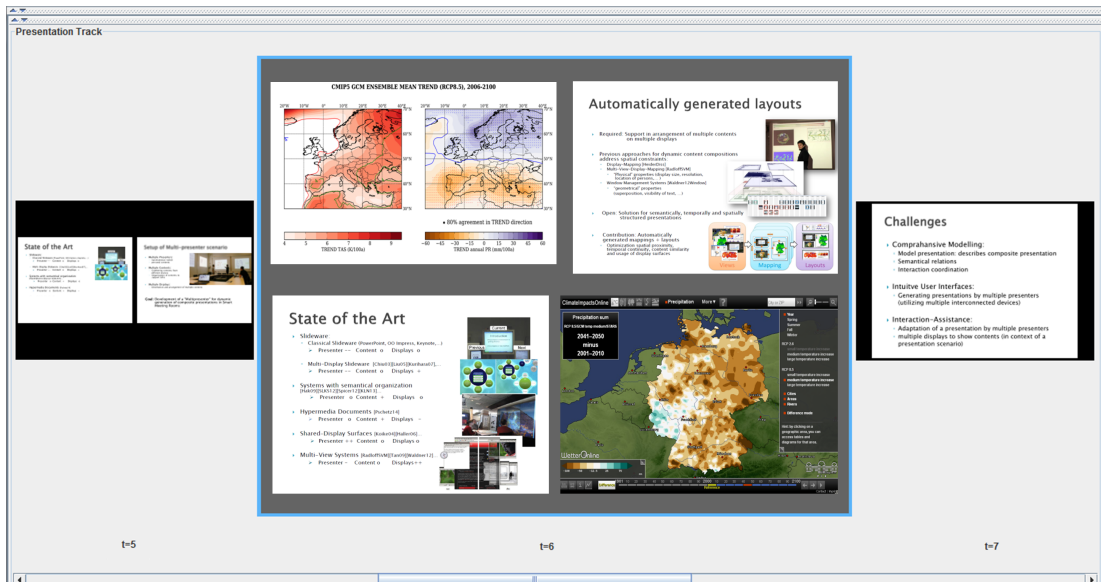
Die verschiedenen Phasen einer Präsentation erfordern es, typischerweise auf andere Funktionen zuzugreifen (Anforderung I1). Entsprechend der zuvor beschriebenen Phasen “Vorbereitung”, “Präsentation” und “Diskussion” wird hierzu das Nutzerinterface auf den persönlichen Geräten der Nutzer automatisch angepasst :

- Für die **Vorbereitung** wird automatisch mehr Platz für den Content Pool und für die Darstellung des Präsentationsgraph bereitgestellt (vgl. Abbildung 6.1(a)). Hierdurch werden die einzelne Vorschaubilder im Content Pool vergrößert dargestellt und Nutzer können die zusammengetragenen Views besser inspizieren und schneller für das Editieren des Präsentationsgraphen auswählen. Weil für den Präsentationsgraph ebenfalls mehr Platz eingeräumt wird, können längere Zeitausschnitte bzw. mehr Knoten pro Zeitschritt angezeigt werden. Hierdurch wird es einfacher, längere Präsentationen zu planen und größere Präsentationsgraphen (an verschiedenen Stellen) zu editieren.
- Während einer **Präsentation** wird dagegen automatisch die Layer Bar besonders groß angezeigt (vgl. Abbildung 6.1(b)). Hierdurch kann der aktuelle Redner die einzelnen Views gut erkennen und kann somit ggf. Stichpunkte von Folien für seinen Vortrag ablesen, ohne sich dafür vom Publikum (hin zu den Displays) abwenden zu müssen.
Zudem können Tools zur bedarfsgerechten Generierung von Views nun schnell mit den gerade gezeigten Views verbunden werden, weil für die Drag & Drop-Geste nur noch eine geringe Interaktionspräzision erforderlich ist, um eine zu ergänzende View in der Layer Bar auszuwählen. Hierdurch kann der Redner für das Präsentieren der Views, auch sehr mobile Geräte wie z.B. Smartphones oder Tablettts verwenden, die nur über kleinere Bildschirme verfügen und deren Touch-Interaktion nicht so präzise wie die Verwendung von Maus und Tastatur ist.
Das automatische Ausblenden des Content Pools bzw. des Präsentationsgraphen kann zudem auch als Aufforderung an das Publikum verstanden werden das Editieren einzustellen und sich dafür auf den aktuellen Redner zu konzentrieren. Die vergrößerte Layer Bar ermöglicht es zudem, dass jeder Teilnehmer im Publikum nochmal individuell einzelne Views detailliert betrachten oder zuvor erläuterte Ideen nachschlagen kann.
- Für **Diskussionen** wird automatisch die Darstellung des Interaktionsgraphen und die Übersichtsdarstellung des Semantischen Graphen aufgerufen. Auf diese Weise können die einzelnen Views und die durch die Nutzer vorgenommenen Anpassungen an der Anzeige unmittelbar den ausführenden Nutzern zugeordnet werden. Weiterhin können so verschiedene Diskussionszweige sofort geordnet abgelesen und bei Bedarf aufgerufen werden.

6.2. Situationsangepasste Assistenz



(a) Nutzerinterface für die Unterstützung der Vorbereitungsphase: Content Pool und Präsentationsgraph werden vergrößert angezeigt.



(b) Nutzerinterface für die Unterstützung der Präsentations-Phase: Es wird ausschließlich die Layer Bar angezeigt.

Abbildung 6.1.: Konfiguration des Nutzerinterfaces auf den persönlichen Geräten. Elemente wie z.B. Content Pool, Präsentationsgraph und Layer Bar werden automatisch ein- bzw. ausgeblendet, um das Interface an die erkannte Situation anzupassen.

6.2.2. Automatische Layout-Anpassungen

Die auf den Displays gezeigten View-Zusammenstellung kann Views von verschiedenen Nutzern enthalten. Abbildung 6.2 zeigt eine solche View-Anzeige, bei der Views von zwei Nutzern auf zwei Displays angezeigt werden. Dieses Beispiel dient im Folgenden als Ausgangslage, um die Layout-Anpassungen zu demonstrieren.

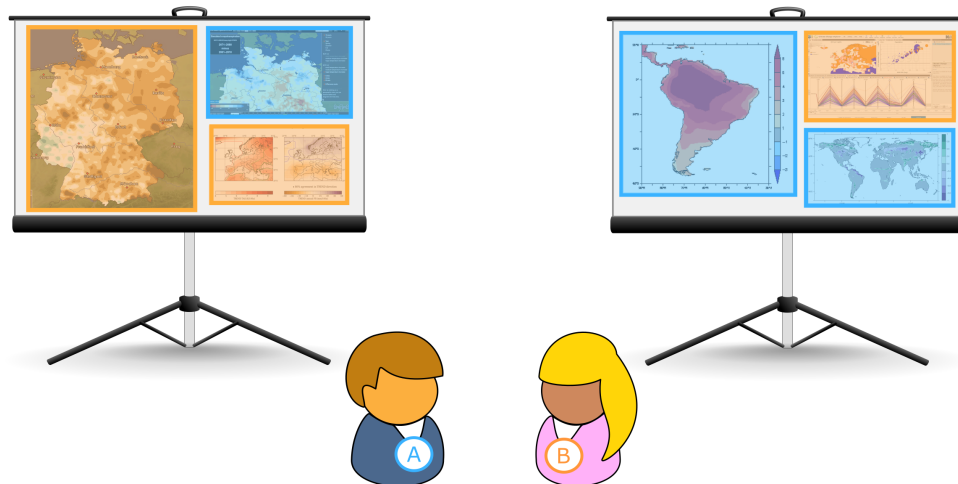


Abbildung 6.2.: Ausgangslage für die situationsbedingte automatische Layout-Anpassung. Views von zwei Nutzern werden gleichzeitig auf zwei Displays angezeigt. Die Einfärbung der Views kennzeichnet den zugehörigen Nutzer. Views von Nutzer A sind blau und Views von Nutzer B orange eingefärbt.

Für eine situationsbezogene Anpassung der View-Verteilung bzw. der Layouts (Anforderung I2), wird nun der Präsentationsgraph automatisch verändert bzw. werden andere Faktoren bei der Layout-Berechnung angepasst (vgl. Abschnitt 4.2). Hierfür werden drei Arten der automatischen Anpassung eingeführt, die jeweils einzeln oder in Kombination angewandt werden:

Nutzer-bezogene Anpassung: Wurde eine Situation erkannt, bei der ein neuer Redner eine Präsentation fortsetzt, werden nun von den angezeigten Views automatisch diejenigen Views visuell hervorgehoben, die von dem neuen Redner stammen (vgl. Abbildung 6.3). Für die Hervorhebung werden die Layouts so verändert, dass fokussierte Views mehr Platz auf den Displays bekommen. Hierfür werden (temporär) die Relevanzwerte der entsprechenden Knoten im Präsentationsgraph erhöht. Bei hohen Erkennungswahrscheinlichkeiten werden zudem nicht fokussierte Views desaturiert oder sogar komplett auf den Displays ausgeblendet.

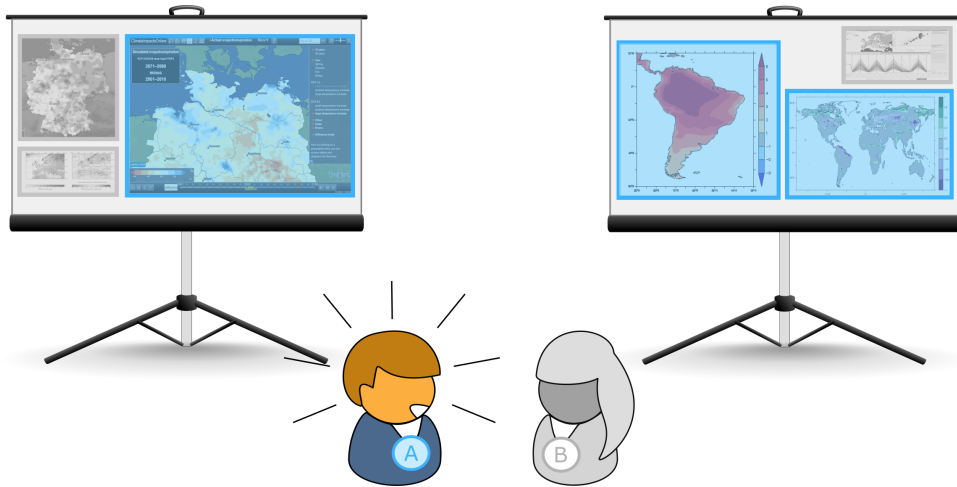


Abbildung 6.3.: Nutzer-bezogene Anpassung der View-Anzeige. Nutzer A wurde als aktiver Redner erkannt. Die Layouts auf den Displays werden daraufhin so angepasst, dass seine Views mehr Platz erhalten. Views von Nutzer B werden desaturiert angezeigt.

Durch den Nutzer-bezogenen Fokus können neue Redner besser die für ihren eigenen Redebeitrag relevanten Views vorstellen und das Publikum hat es einfacher, diese Views innerhalb der View-Zusammenstellung zu identifizieren.

Display-bezogene Anpassung: Für das Beschreiben von Views ist es oft sinnvoll, wenn sich der Redner nahe bei den relevanten Views aufhält, um z.B. direkt auf Details innerhalb der Views deuten zu können. Um dementsprechend die View-Verteilung auf den Displays anzupassen, werden beim Rednerwechsel nun automatisch diejenigen Displays (bei der Layout-Berechnung) höher bewertet, die sich nahe bei der Position des neuen Redners befinden. Hierdurch werden bei der Neuberechnung der View-Verteilung die relevanten Views eher nahe beim Redner positioniert und dieser muss nicht selbst seine Position innerhalb des Raums verändern (vgl. Abbildung 6.4).

Inhalts-bezogene Anpassung: Ursprünglich für eine Präsentation vorbereiteten Gruppierungen von Views spielen mit dem Fortschreiten einer Diskussionen eine zunehmend geringere Rolle. Dies ist insbesondere der Fall, wenn die angezeigte View-Zusammenstellung häufig durch neue Views erweitert wird, um beispielsweise auch alternativen Diskussionszweigen nachzugehen.

Um trotzdem eine Gruppierung für die Views (ohne weitere manuelle Anpassungen durch die Nutzer) bereitzustellen, werden nun automatisch solche Views näher beieinander positioniert, die semantisch zueinander ähnlich sind und die von den gleichen Nutzern hinzugefügt wurden. Dies wird erreicht, indem automatisch Kan-

6. Integration in den Smart Meeting Room

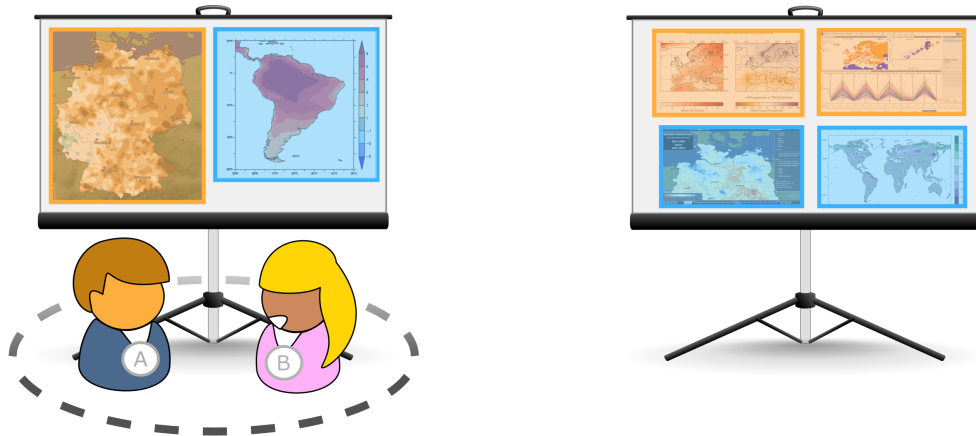


Abbildung 6.4.: Display-bezogene Anpassung der View-Anzeige. Der aktive Redner wurde in der Nähe des linken Displays lokalisiert. Daraufhin wird die View-Verteilung so verändert, dass relevante Views eher auf dem linken Display und damit in der Nähe des Redners erscheinen. (In dem Beispiel, konnte jedoch nicht eindeutig festgestellt werden, ob Nutzer A oder B der Redner ist, weshalb zusätzlich keine Nutzer-bezogene Anpassung erfolgt.)

ten aus den Semantischen Graphen als neue Kanten in den Präsentationsgraph eingefügt werden und dafür alte Kanten aus dem Präsentationsgraph entfernt werden. Mit zunehmendem Diskussionsverlauf führt dies dazu, dass sich die Views von verschiedenen Diskussionszweigen bzw. Nutzern nicht mehr vermischen, sondern sich stattdessen automatisch räumlich auf den Displays gruppieren. Dies erleichtert es, die verschiedenen Argumente der einzelnen Nutzer auseinanderzuhalten und jeweils separat auf einem einzelnen Display zu betrachten.

Die verschiedenen Arten der Anpassung am Präsentationsgraph können miteinander kombiniert werden. Nutzer- und Display-bezogene Anpassungen führen zusammen beispielsweise dazu, dass die neuen Views eines Redners vergrößert auf den Displays in seiner Nähe angezeigt werden (vgl. Abbildung 6.5). Andererseits erlaubt es die Unabhängigkeit der einzelnen Anpassungsarten auch, auf nur teilweise erkannte Situationen zu reagieren. Stehen beispielsweise zwei Nutzer räumlich dicht beieinander (oder tragen keine Nutzer-Tags), kann die Situationserkennung nicht immer sicher bestimmen, welcher der beiden gerade spricht. Somit können nur Display-bezogene Anpassungen, aber keine Nutzer-bezogenen Anpassungen vorgenommen werden (vgl. Abbildung 6.4).

Mit den Nutzer- und Display-bezogenen Anpassungen werden Rednerwechsel durch automatische Anpassungen der Layouts adressiert (Anforderung I2). Die Inhalts-bezogenen Anpassungen erlauben es hierbei außerdem, die Anzeige auch für unterschiedliche Phasen (z.B. Präsentation und Diskussion) automatisch zu konfigurieren (Anforderung I1).

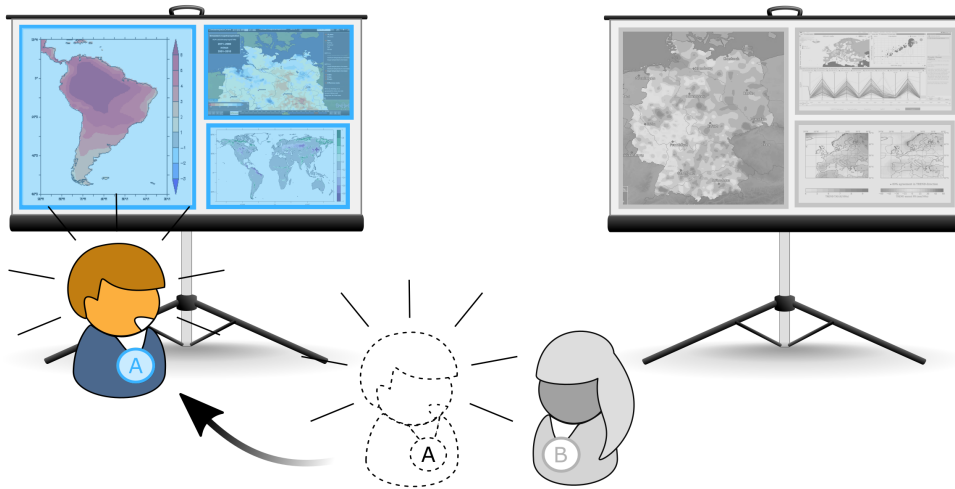


Abbildung 6.5.: Display- und Nutzer- bezogene Anpassung der View-Anzeige. Der aktive Redner wird als Nutzer A identifiziert und in der Nähe des linken Displays erkannt. Die View-Verteilung wird automatisch so verändert, dass die Views des Nutzers A auf dem Display in seiner Nähe angezeigt werden.

6.2.3. Automatisches Einblenden von Semantischen Informationen

Die Beantwortung von Fragen aus dem Publikum kann durch Anwenden der in Abschnitt 5.2 vorgestellten Hilfsmittel zur Suche nach semantisch verwandten Views unterstützt werden (Anforderung I3). Ebenfalls kann auf Situationen in denen ein “ziello- ses” Navigationsverhalten detektiert wurde reagiert werden, indem Navigationshinweise zu semantisch passenden Views gegeben werden. Die bereitgestellten Ansätze erfordern es bisher jedoch, dass als Ausgangspunkt zunächst Views (manuell) ausgewählt werden, für die dann anschließend ähnliche Views angezeigt bzw. Navigationsempfehlungen ein- geblendet werden.

- Die Auswahl an Views wird nun situationsabhängig automatisch vorgenommen, um den aktuellen Redner zu unterstützen. Hierzu werden zunächst diejenigen Views ausgewählt, die beim Stellen einer Frage gerade zu sehen sind, bzw. die gezeigt wurden, als zuletzt ein normales Präsentationsverhalten erkannt wurde. Diese Auswahl wird anschließend weiter eingeschränkt, wenn erkannt wird, dass eine bestimmte Auswahl von diesen Views sich öfter im Blickfeld der Nutzer be- findet bzw. wenn sich der Redner im Raum auf diese Views (auf den Displays) zubewegt.

Auf diese Weise ist im Idealfall keine weitere Interaktion vom Redner notwendig, um Hintergrundinformationen anzufordern. In anderen Fällen, z.B. wenn eine ge- stellte Frage eine View betrifft, die gerade nicht angezeigt wird, muss der Redner

6. Integration in den Smart Meeting Room

jedoch weiterhin Views manuell auswählen. Um dies zu erleichtern, wird das Nutzerinterface auf dem persönlichen Gerät des Redners automatisch so umgeschaltet, dass der Präsentationsgraph gut zu sehen ist und somit Views sofort ausgewählt werden können.

- Das Anzeigen von Navigationshinweisen, von Visual Links und von zusätzlichen, semantisch passenden Views kann zwar den Redner bei der Suche nach Views unterstützen, gleichzeitig aber auch das Publikum bzw. andere Diskussionsteilnehmer ablenken und somit daran hindern, der vorgetragenen Argumentationslinie zu folgen. Aus diesem Grund werden, die entwickelten Hilfsmittel für die Unterstützung der Suche nicht global auf allen persönlichen Geräten gezeigt, sondern nur selektiv auf dem Gerät des aktuellen Redners eingeblendet, um diesem kurzzeitig zu assistieren. Auf ähnliche Weise werden semantisch verwandte Views auch nur auf denjenigen Displays zusätzlich eingeblendet, die sich gerade im Sichtbereich der Nutzer befinden und auf denen sich viele der zuvor ausgewählten Views befinden.

Durch das beschriebene Vorgehen werden einzelne Nutzer gezielt bei der Bewältigung von Situationen unterstützt, die das Suchen nach und das Navigieren zu semantisch verwandten Views erfordern (Anforderung I3). Die Assistenz ist hierfür so ausgelegt, dass ein zuvor verfolgter Präsentations- bzw. Diskussionsverlauf möglichst wenig gestört wird und anschließend ein schnelles Zurückkehren zu einer gewohnten Präsentationsansicht ermöglicht wird.

Die in diesem Kapitel vorgestellten Konzepte wurden in Zusammenarbeit mit Martin Nyolt vom Lehrstuhl Mobile Multimediale Informationssysteme (MMIS) an der Universität Rostock entwickelt und im Smart Lab getestet. Durch die Zusammenarbeit konnten einerseits für diese Dissertation mögliche Assistenzansätze bestimmt und auf eine konkrete Umgebung abgestimmt werden, indem bspw. verschiedene Parametrisierung für die Layout-Anpassungen getestet werden konnten. Andererseits bietet der konkrete Anwendungshintergrund die Möglichkeit, bestehende Konzepte für die Situationserkennung im Smart Lab zu testen und neue Anforderungen hinsichtlich der Weiterentwicklung dieser Systeme abzuleiten.

Zusammenfassend kann festgestellt werden, dass die vorgestellten Ansätze eine einfache situationsabhängige Assistenz für ausgewählte Situationen innerhalb von Präsentations- und Diskussionsszenarien bieten. Durch das beschriebene Vorgehen wird die Grundlage dafür gelegt, dass die Nutzer weniger manuelle Interaktion benötigen, um die in den früheren Kapiteln eingeführten Funktionalitäten zu nutzen, die Anzeige auf den Displays anzupassen oder semantische Informationen einzublenden (Anforderungen I1-I3). Hierdurch können sich die Nutzer besser auf die eigentliche Präsentation bzw. Diskussion konzentrieren und werden weniger durch die Bedienung bzw. durch die notwendige manuelle Umgestaltung der Anzeige abgelenkt.

Mit einer zunehmend besseren Situationserkennung könnten in Zukunft weitere und

6.2. Situationsangepasste Assistenz

detailliertere Szenarien erkannt werden und andere Assistenzansätze analog eingeführt werden. Weitere Ideen hierzu finden sich in dem Ausblick dieser Arbeit (vgl. Abschnitt 7.2).

7. Zusammenfassung und Ausblick

7.1. Zusammenfassung

Die Anzeige von verschiedenen Views auf unterschiedlichen Displays ist ein gängiger Ansatz, um die Präsentation und Diskussion von Informationen zu unterstützen. Smart Meeting Rooms sind eine besondere Art von Multi-Display-Umgebungen. Sie verfolgen das Ziel, Nutzer in der Umgebung durch eine möglichst umfassende Assistenz beim Erfüllen ihrer Aufgaben zu unterstützen. Dies macht Smart Meeting Rooms zu idealen Arbeitsumgebungen, in denen Präsentationen und Diskussionen mit mehreren beteiligten Nutzern durch eine assistierte Informationsanzeige unterstützt werden können.

Wie sich jedoch bei Betrachtung der Literatur in Kapitel 2 dieser Dissertation zeigt, decken gängige Assistenzansätze für Smart Meeting Rooms nicht alle für Präsentations- und Diskussionsszenarien notwendigen Aufgabengebiete ab. Andererseits sind Präsentationssysteme für Multi-Display-Umgebungen nicht dafür ausgelegt, automatische Assistenz (wie sie bspw. durch die Situationserkennung in Smart Meeting Rooms möglich wäre) zu integrieren. Im Rahmen dieser Dissertation wurde diese Lücke in der bestehenden Literatur als Motivation genommen, um neuartige Konzepte für eine assistierte Informationsanzeige in Smart Meeting Rooms zu entwickeln.

In Kapitel 3 wurden hierzu zunächst drei Modellbeschreibungen entwickelt, welche die Grundlage für die später vorgestellten Ansätzen zur assistierten Präsentation und zur assistierten Diskussion bilden. Der Präsentationgraph modelliert hierzu Vorgaben für die räumliche Verteilung und für die zeitliche Ersetzung mehrerer Views auf unterschiedlichen Displays. Der Semantische Graph stellt einen thematischen Zusammenhang zwischen verschiedenen Views her, um eine spätere Suche nach passenden Inhalten zu unterstützen und der Interaktionsgraph beschreibt durch die Nutzer vorgenommene Anpassungen an der Anzeige.

Die in Kapitel 4 vorgestellten Ansätze zur assistierten Präsentation konzentrieren sich auf die Gestaltung und das Aktualisieren der Anzeige auf mehreren Displays unter Verwendung von mehreren Views aus verschiedenen Quellen. Hierzu wurde ein Nutzerinterface entwickelt, das es mehreren Nutzern erlaubt, Inhalte von verschiedenen persönlichen Geräten zusammenzutragen und für die Anzeige auf den Displays bereitzustellen. Der in diesem Prozess interaktiv (gemeinsam durch mehrere Nutzer) erstellte Präsentationsgraph steuert einen Layout-Algorithmus. Dieser berechnet automatisch eine günstige Verteilung der Views auf die einzelnen Displays und erstellt pro Display

7. Zusammenfassung und Ausblick

ein Layout, welches die Positionierung und Skalierung der Views bestimmt. Im Gegensatz zu vorherigen Ansätzen berücksichtigt dieser neue Layout-Algorithmus hierbei zusätzlich die (durch den Smart Meeting Room ermittelte) Sichtbarkeit der Views bzw. Displays sowie auch Nutzervorgaben für die Anzeige der Views. Er ist ausreichend schnell, um eine Aktualisieren der Layouts in interaktiven Raten zu ermöglichen.

Eine besondere Eigenschaft der assistierten Präsentation ist es, dass eine bedarfsgerechte Generierung und Aktualisierung von Views ermöglicht wird. Hierbei werden verschiedene Tools zur View-Generierung spontan (über das entwickelte Interface) mit der Präsentation verknüpft und zum interaktiven Erstellen bzw. Anpassen unterschiedlicher Views genutzt. Diese neuen Darstellungen werden automatisch als neue Views in den Präsentationsgraph integriert und durch den Layout-Algorithmus auf den Displays angezeigt. Die Erzeugung und Anzeige der neuen Views erlaubt es somit, fließend zwischen der Präsentation und der Exploration von Daten und Informationen zu wechseln. Auf diese Weise wird es bspw. möglich, mit neuen Views gezielt auf gestellte Fragen zu antworten, Analysevorgänge und deren Ergebnisse gemeinsam in der Gruppe zu betrachten und hierbei ggf. alternative Hypothesen anhand von spontan erstellten Visualisierungen zu testen. Konkret wird dies mit einem Lupen-Tool für die Analyse von großen Bilddaten, mit einem Tool für die Feature-basierte Analyse von Volumendaten und mit einem Tool zur Integration von Web-Inhalten an jeweils einem Anwendungsfall demonstriert.

Mit der assistierten Präsentation erhalten Nutzer somit Assistenz beim Gestalten der Anzeige, bei der Anpassung und Aktualisierung der View-Verteilung auf den Displays und bei der spontanen Integration von neuen Inhalten. Auf diese Weise wird der bisherige Stand der Forschung bereits deutlich erweitert. Die vorgestellten Ansätze sind allein jedoch noch nicht ausreichend, um im Rahmen einer dynamisch verlaufenden Diskussion einen häufigen Wechsel zwischen verschiedenen Argumentationslinien oder die Suche nach bestimmten Inhalten zu unterstützen.

Die in Kapitel 5 beschriebenen Ansätze zur assistierten Diskussion bauen deshalb auf der assistierten Präsentation auf, stellen nun jedoch die Assistenz bei der Interaktion in den Vordergrund.

Im ersten Teil wird eine entwickelte Visualisierung des Interaktionsgraphen vorgestellt, die es ermöglicht, den Verlauf einer Diskussion differenziert nach den verwendeten Displays, den beteiligten Nutzern und den betroffenen Views zu untersuchen. Hierauf basierend kann anschließend ein entwickelter Undo/Redo-Mechanismus dazu verwendet werden, um mit geringem manuellen Aufwand frühere Zustände der Anzeige aufzurufen. Dadurch wird es möglich, schnell zwischen verschiedenen Argumentationslinien zu wechseln. Weiterhin können ausgewählte Anpassungen mittels des selektiven Undo/Redo-Mechanismus gesondert für einzelne Displays, Views oder Nutzer rückgängig gemacht oder wiederhergestellt werden. Die nach diesen drei Aspekten differenzierende Undo/Redo-Funktionalität, geht über das sonst im Bereiche der Multi-Display-Umgebungen angebotene Maß hinaus und ist ein wichtiger Beitrag dieser Dissertation.

Im zweiten Teil von Kapitel 5 wird anschließend die Unterstützung der Suche nach semantisch passenden Inhalten thematisiert. Hierfür wird zunächst eine Visualisierung des Semantischen Graphen vorgestellt, welche die auf den Graphattributen und auf der Graphstruktur basierenden Ähnlichkeitsbeziehungen zwischen Inhalten übersichtsweise verdeutlicht. Um hierbei sowohl kategorische als auch quantitative Attributwerte zu adressieren, wurden zwei unterschiedliche Visualisierungen entwickelt, die darüber hinaus auch ein ggf. notwendiges Editieren der Attribute erlauben. In einem weiteren Schritt werden anschließend semantische Beziehungen zwischen Views auch im Interface und auf den Displays angezeigt. Dazu werden u.a. Visual-Links in die Zeichnung des Präsentationsgraphen aufgenommen oder Navigationsempfehlung zu den semantisch passenden Views auf den Displays eingeblendet.

Auf diese Weise komplettieren die Ansätzen zur assistierten Diskussion diejenigen zur assistierten Präsentation indem der manuelle Interaktionsaufwand zum Anpassen der Anzeige reduziert und die Suche nach Views vereinfacht wird. Bisher offen bleibt dabei jedoch noch die Adaption der Anzeige an unterschiedliche Szenarien und Situationen.

In Kapitel 6 wird deshalb die situationsangepasste Verwendung der zuvor genannten Funktionen beschrieben. Hierbei wird zunächst auf konkrete Szenarien eingegangen, die mittels der Situationserkennung von Smart Meeting Rooms erfasst werden können und die gleichzeitig für Präsentations- und Diskussionsszenarien eine wichtige Rolle spielen. Anschließend werden mit der automatischen Interface-Anpassung, der automatische Layout-Anpassung und dem automatischen Einblenden von semantischen Informationen Ansätze vorgestellt, die Nutzer gezielt in diesen Situationen unterstützen. Mit diesen Adaptionsmechanismen für ausgewählte Situationen in Präsentations- und Diskussionsszenarien wird eine Assistenz erreicht, die den bisherigen Stand der Forschung im Bereich der Smart Meeting Rooms erweitert.

Zusammenfassend kann somit festgestellt werden, dass die in dieser Dissertation besprochenen Konzepte und Techniken einen neuen Lösungsansatz für die Gestaltung der Informationsanzeige bieten. Einerseits kommen hierbei vergleichsweise einfache, und damit interaktiv während einer Besprechung zu editierende, Modellbeschreibung zur Anwendung. Andererseits werden nun Assistenzansätze genutzt, um anhand dieser Beschreibung eine angepasste Informationsanzeige automatisch zu berechnen. Diese Kombination erlaubt es anschließend, zuvor nur schwer umzusetzende Konzepte, wie z.B. das gemeinsame Bearbeiten der Anzeige, die spontane Integration neuer Views, die Verwendung von stark differenzierenden Undo/Redo-Operationen oder die situationsangepassten Adaptionen zu realisieren. Alle hierfür entwickelten Konzepte wurden anhand realer Fallstudien getestet.

Auf diese Weise bietet die in dieser Dissertation vorgestellte Kombination aus Modellbeschreibungen und darauf aufbauenden Ansätzen für die assistierte Präsentation und Diskussion einen wichtigen Beitrag, welcher gezielt die Lücke zwischen der Informationspräsentation in Multi-Display-Umgebungen und den verfügbaren Assistenzansätzen für Smart Meeting Rooms adressiert.

7.2. Ausblick

Obwohl die Anforderungen für die Informationsanzeige in Smart Meeting Room im Rahmen dieser Dissertation umfassend diskutiert und neue Konzepte entwickelt wurden, bleibt weiterhin Raum für zukünftige Arbeiten:

Erweiterung der Informationspräsentation: Die entwickelte assistierte Präsentation erlaubt es, mehrere Displays in einem SMR für die Anzeige von Views zu nutzen. Der Layout-Algorithmus kann hierbei mit einer wechselnden Anzahl von Displays umgehen. Wird bspw. ein weiterer Projektor zugeschaltet und für das Präsentationssystem über die Middleware freigegeben, so verändert der Ansatz automatisch die View-Verteilung und Anordnung, um das neue Display ggf. zu nutzen. Auf diese Weise kann zwar auf eine Veränderung der Gerätekonfiguration reagiert werden, es wäre aber darüber hinaus wünschenswert, wenn solche Veränderungen auch aktiv durch das Präsentationssystem bzw. den Layout-Algorithmus angestoßen werden könnten.

Viele Smarte Umgebungen sind dazu prinzipiell in der Lage. Das Smart Lab in Rostock erlaubt es beispielsweise, Steuerbefehle über die Middleware an die Geräte zu senden. Auf diese Weise können Projektoren angeschaltet, Leinwände heruntergefahren, einzelne Lampen gedimmt und Sonnenblenden geschlossen werden. Wenn es erforderlich wird, könnte somit eine Software-Komponente im Smart Meeting Room die entsprechenden Aktionen ausführen, um bei Bedarf Displays selbstständig hinzuzufügen (oder zu entfernen).

Eine Herausforderung dieser Erweiterung liegt nun darin, seitens der Informationspräsentation zu entscheiden, wann und wo Displays an- bzw. abgeschaltet werden sollen. Einen möglichen Ansatzpunkt den Bedarf an Displays zu definieren, liefert die in dieser Arbeit entwickelte Bewertungsfunktion (vgl. Abschnitt 4.2). Nach dieser Funktion wird die Anzeigequalität für jedes Display separat berechnet. Liegt die kleinste dieser separaten Anzeigequalitäten (längere Zeit) unter einem Grenzwert, bedeutet dies, dass der Layout-Algorithmus allein durch ein Umverteilen und Skalieren der Views die Informationsanzeige nicht weiter verbessern konnte und somit ein Bedarf nach einem zusätzlichen Display besteht.

Bei der Auswahl eines anzuschaltenden Displays spielen verschiedene Faktoren eine Rolle. Einerseits sollten hierbei Raumeigenschaften berücksichtigt werden (wie sie sich beispielsweise durch die Formeln in Anhang A beschreiben lassen) und andererseits sollten die Auswirkungen auf die View-Umverteilung betrachtet werden, wenn das Display hinzugeschaltet wird. Ist diese Umverteilung zu groß, könnte das Zuschalten des Displays für die Informationspräsentation als störend empfunden werden. Aus diesem Grund bietet es sich an, vor dem eigentlichen Einschalten eines Displays eine Vorberechnung (mit dem Layout-Algorithmus) auszuführen bei der verschiedene Varianten getestet werden. Basierend auf diesen vorberechneten (aber noch nicht verwendeten) Layouts könnte außerdem ein

günstiger Einschaltzeitpunkt (innerhalb der nahen Zukunft) vorhergesagt werden. Hierbei ist zu beachten, dass es einige Zeit dauern kann bis ein neuer Projektor seine volle Leuchtkraft erreicht oder eine Leinwand vollständig heruntergefahren ist. Entsprechend sollten bevorzugt solche Zeitpunkte gewählt werden bei denen das neue Display nicht genutzt oder nur Views mit geringen Relevanzwerten zeigen würde.

Erweiterung der assistierten Interaktion: Die Steuerung der Anzeige basiert bisher einerseits auf der direkten Interaktion des Nutzers mit der Umgebung, wenn dieser z.B. sein persönliches Gerät bedient und andererseits auf der angewendeten Nutzerassistenz, die auf automatisch erkanntem Verhalten reagiert. Die direkte Nutzerinteraktion ist dabei z.Z. auf solche Interaktionsarten beschränkt, die auf Maus- und Tastatur-Eingaben abgebildet werden können. Dies erlaubt bspw. den Einsatz von Touch-Geräten wie z.B. Tablets oder von Zeigegeräten, wie Wii-Controllern oder Gyromäusen (vgl. [Rad+12; Rad14]). Sprach- und Gestensteuerung ist dagegen bisher noch unbeachtet geblieben. Auf der anderen Seite ist die automatische Assistenz limitiert, da Situationen und Absichten von Nutzern teilweise nur unvollständig bzw. mit zu großen Unsicherheiten erkannt werden.

Denkbar ist ein möglicher Mittelweg, bei dem einerseits verschiedene Interaktionsarten, wie z.B. Sprache, Gesten und Maussteuerung miteinander kombiniert werden und andererseits die Situationserkennung fehlende Teilinformationen vom Nutzer anfragen kann. Ein entsprechender Ansatz wurde bereits in vorangegangenen Arbeiten erarbeitet und zum Steuern der Geräteinfrastruktur im Smart Lab von Rostock verwendet [BT12].

In zukünftigen Arbeiten könnte der entsprechende Ansatz weiterentwickelt und auf die neuen Konzepte zur assistierten Diskussion abgestimmt werden. Wenn beispielsweise für das Anwenden der automatischen Layout-Anpassungen (vgl. Abschnitt 6.2.2) der Rednerstandort, nicht aber die konkrete Person erkannt wurde, könnte das System diese fehlende Information (bspw. über einen Dialog oder eine Spracheingabe) nachfragen und so anschließend auch eine nutzerbezogene Layout-Anpassung anbieten (auf die sonst verzichtet werden müsste). Andererseits könnten Nutzer auch präventiv einfache Gesten oder knappe Sprachkommandos verwenden, um das Erkennen einer Situation zu vereinfachen. Obwohl auf diese Weise zunächst (optional) mehr direkte Nutzerinteraktion erforderlich wird, lässt sich, durch die bessere Anwendung der entwickelten Assistenzansätze, der Gesamtaufwand für den Nutzer wahrscheinlich reduzieren. Gleichzeitig könnten so auch feingranularere Situationen erkannt und zukünftige Assistenzansätze genauer auf diese abgestimmt werden.

Erweiterung der situationsbedingten Assistenz: Bei der situationsangepassten Assistenz wurden in erster Linie Präsentations- und Diskussionsszenarien betrachtet. Die Gründe dafür waren, dass diese Szenarien im Fokus der zuvor entwickelten

7. Zusammenfassung und Ausblick

Ansätze standen und die vorhandene Situationserkennung bereits erste Ansätze zum Erfassen dieser Szenarien bereitstellte.

Mit einer zukünftigen Weiterentwicklung der Situationserkennung könnten aber auch andere Szenarien besser erkannt werden. Insbesondere Explorationsszenarien bei denen Visual-Analytics-Methoden Anwendung finden, werden bereits durch die Verknüpfung von Tools zur bedarfsgerechten Erzeugung von neuen Views unterstützt. Eine situationsangepasste Assistenz könnte in solchen Szenarien nun beispielsweise das Aufrufen dieser Tools übernehmen, automatisch die bei der Erzeugung hinterlegte Hintergrundinformationen einblenden oder durch veränderte Layout-Berechnungen die Views nach Analyseschwerpunkten auf den Displays anordnen.

Evaluierung: Obwohl die vorgestellten Ansätze einerseits zusammen mit Anwendungsexperten entwickelt wurden und andererseits in realen Fallstudien getestet wurden, ist eine umfassende Evaluierung noch erforderlich. Die vorgestellten Konzepte bzw. das im Rahmen dieser Arbeit implementierte System eröffnen Nutzern neue Möglichkeiten bei der Informationspräsentation, die weit über etablierte Präsentationssysteme hinaus gehen. Bisher ist jedoch noch offen, ob so ein Präsentationsansatz, der bspw. die Integration von neuen Views erlaubt, allgemein akzeptiert und genutzt werden würde. Um den genauen Einfluss auf das Präsentationsverhalten der Nutzer und auf deren Umgang mit Informationen zu untersuchen, ist somit noch eine kontrollierte Nutzerstudie mit einer größeren Anzahl an Teilnehmern erforderlich.

Weitere Anwendungskontexte: Wie motiviert sind Smart Meeting Rooms (prinzipiell) ideale Umgebungen für die Präsentation von Informationen und für das Abhalten von Besprechungen mit mehreren Beteiligten. Allerdings ist ihre tatsächliche Verwendung zumeist auf Forschungseinrichtungen und in naher Zukunft voraussichtlich noch auf den Einsatz in Lehrinrichtungen beschränkt. Wie eine aktuelle Arbeit zeigt, spielen die in dieser Dissertation adressierten Problemstellungen aber auch zunehmend im Business-Bereich eine entscheidende Rolle [HKD16]. Allerdings ist hier der finanzielle und logistische Aufwand für die Installation und den Betrieb von Smart Meeting Rooms noch ein Grund für deren zurückhaltende Verwendung. Zusätzlich ist es in Business-Anwendungen oft erforderlich, dass Personen, die nicht physisch vor Ort sind in eine Besprechung mit einbezogen werden können. In der Praxis werden hierfür typischerweise Videokonferenzen organisiert. Die Forschung im Bereich der Telepräsenz geht aber darüber hinaus und versucht, eine gleichzeitige Anwesenheit der räumlich getrennten Personen zu simulieren, um einen natürlicheren Informationsaustausch zu ermöglichen [OWS15; Fai+16]. Einfache Systeme sind hierbei jedoch auf die gleiche Weise limitiert, wie klassische Präsentationsumgebungen bzw. Multi-Display-Umgebungen gegenüber Smart Meeting Rooms: Aufgrund der beschränkten Darstellungsfläche und Interaktionsmöglichkeiten können Informationen nur schlecht in ihrer Gesamtheit

anhand von Darstellungen übermittelt und mit mehreren Personen diskutiert werden. Komplexere Installationen für Telepräsenz haben dagegen die gleichen Nachteile, die auch Smart Meeting Rooms haben: Solche Systeme brauchen i.d.R. (mehrere) separate Räume mit vielen verteilten Displays und Kameras, was sie in der Anschaffung teuer und wenig mobil macht.

Eine Möglichkeit, um einerseits eine Brücke zwischen der assistierten Informationsanzeige und den Telepräsenz-Systemen zu schlagen und um andererseits das Kosten-, Verbreitungs- und Mobilitätsproblem zu umgehen, wird durch die aktuelle Weiterentwicklung und zunehmende Verbreitung von Head-Mounted Displays eröffnet. Entsprechende Systeme wie bspw. die Oculus Rift, die HTC Vive oder die Sony VR-Brille erlauben es theoretisch, komplett virtuelle "Meeting Rooms" zu erstellen in denen eine Besprechung (mit mehreren virtuellen Avataren) stattfinden kann. Obwohl in solchen virtuellen Räumen nicht mehr echte Computerbildschirme oder Projektoren verwendet werden müssen, um Views für die Nutzer sichtbar zu machen, lassen sich viele der prinzipiellen Problemstellungen aus echten Umgebungen auch auf diese virtuellen Umgebungen übertragen. So muss bspw. weiterhin eine Stelle im simulierten 3D Raum gefunden werden, an der eine View angezeigt werden soll, es müssen weiterhin semantisch passende Views gesucht und von mehreren Nutzern zusammengetragen, organisiert bzw. angepasst werden und es ist weiterhin erforderlich, dass hierbei die Nutzer durch ein Assistenzsystem unterstützt werden.

Die Herausforderung für zukünftige Forschungsaufgaben liegt nun in der Adressierung der besonderen Eigenheiten dieser "Virtual Smart Meeting Rooms". Beispielsweise ist es wahrscheinlich erforderlich, den Bewegungsspielraum der Nutzer innerhalb der simulierten Räume einzuschränken, um den Limitierungen der verwendeten VR-Brillen gerecht zu werden und um, im Sinne eines Telepräsenz-Systems, die Interaktion mit der Umgebung so natürlich wie möglich zu halten. Andererseits muss der Raum für die Informationsanzeige nicht mehr statisch sein: Displays sind in diesem Sinne nur noch frei positionierbare Anzeigeflächen die jederzeit in ihrer Größe, Anzahl und Form verändert werden können. (Diese Probleme und Möglichkeiten finden sich teilweise auch in Zwischenlösungen, bei denen mittels VR-Brillen virtuelle Displays in einem real existierenden Besprechungsraum eingeblendet werden.)

Viele der in dieser Dissertation erarbeiteten Ansätze könnten weiterverwendet werden, um diese Problemstellungen zu adressieren. Andere sich ergebende Limitierungen erfordern es aber ggf. existierende Lösungen weiterzuentwickeln, um bspw. die noch geringen Auflösungen der VR-Brillen bei der Sichtbarkeitsberechnung von Views zu berücksichtigen.

A. Layout-Berechnung

Die folgenden Formeln sind aus der Dissertation von Radloff entnommen wobei die Variablenbenennung leicht geändert wurde [Rad14]. Sie bewerten die Sichtbarkeit einer View abhängig von der Position und Ausrichtung eines Nutzers bzgl. eines Displays. In der vorliegenden Dissertation fließen diese Formeln in die Bewertungsfunktion für die automatische Layout-Berechnung ein (vgl. Abschnitt 4.2). Sie sind Teil der Bewertungsfunktion für die Anzeigequalität Q_d (vgl. Formel 4.2 auf Seite 64) und werden dort verwendet, um den Sichtbarkeitsfaktors $d(v)$ zu berechnen.

Für die Berechnung werden die Positionen der Nutzer in dem Smart Meeting Room durch Bodensensoren und tragbare Tags ermittelt (siehe Abschnitt 2.1.3). Die physikalische Displaygrößen und Ausrichtungen werden aus einer hinterlegten Raumbeschreibung ausgelesen.

Ansatz von Radloff:

1. Zuerst wird die für jeden Nutzer sichtbare Displayfläche γ berechnet. Sei dazu α der Rotationswinkel des betrachteten Displays bzgl. der Sichtachse des Nutzers, d der Abstand zwischen Nutzer und Display-Mittelpunkt und a die halbe Ausdehnung der Displayfläche. Die Werte werden in Abbildung A.1 illustriert.

$$\gamma = \arctan\left(\frac{\cos(\alpha) \cdot a}{d + \sin(\alpha) \cdot a}\right) + \arctan\left(\frac{\cos(\alpha) \cdot a}{d - \sin(\alpha) \cdot a}\right)$$

Die Berechnung von γ erfolgt jeweils für die vertikale und die horizontale Sichtebeine. Entsprechend wird für α einmal der horizontale und einmal der vertikale Winkel verwendet und für a einmal die Breite und einmal die Höhe des Displays. Die beiden sich ergebenden Werte werden im Folgenden als γ_h und γ_v bezeichnet.

2. Die menschliche Wahrnehmung ist nicht in allen Bereichen des Sichtfeldes gleich gut. Beispielsweise ist sie im peripheren Blickfeld schlechter als im zentralen Feld. Entsprechend wird die im ersten Schritt berechnete Displayfläche nun bewertet.

$$fov_h(\gamma_h) = \begin{cases} \frac{1}{20^\circ} \cdot \gamma_h & , \gamma_h < 20^\circ \\ \cos\left(\frac{9}{16} \cdot \gamma_h - \frac{9 \cdot 20}{16}\right) & , \gamma_h \geq 20^\circ \end{cases}$$

A. Layout-Berechnung

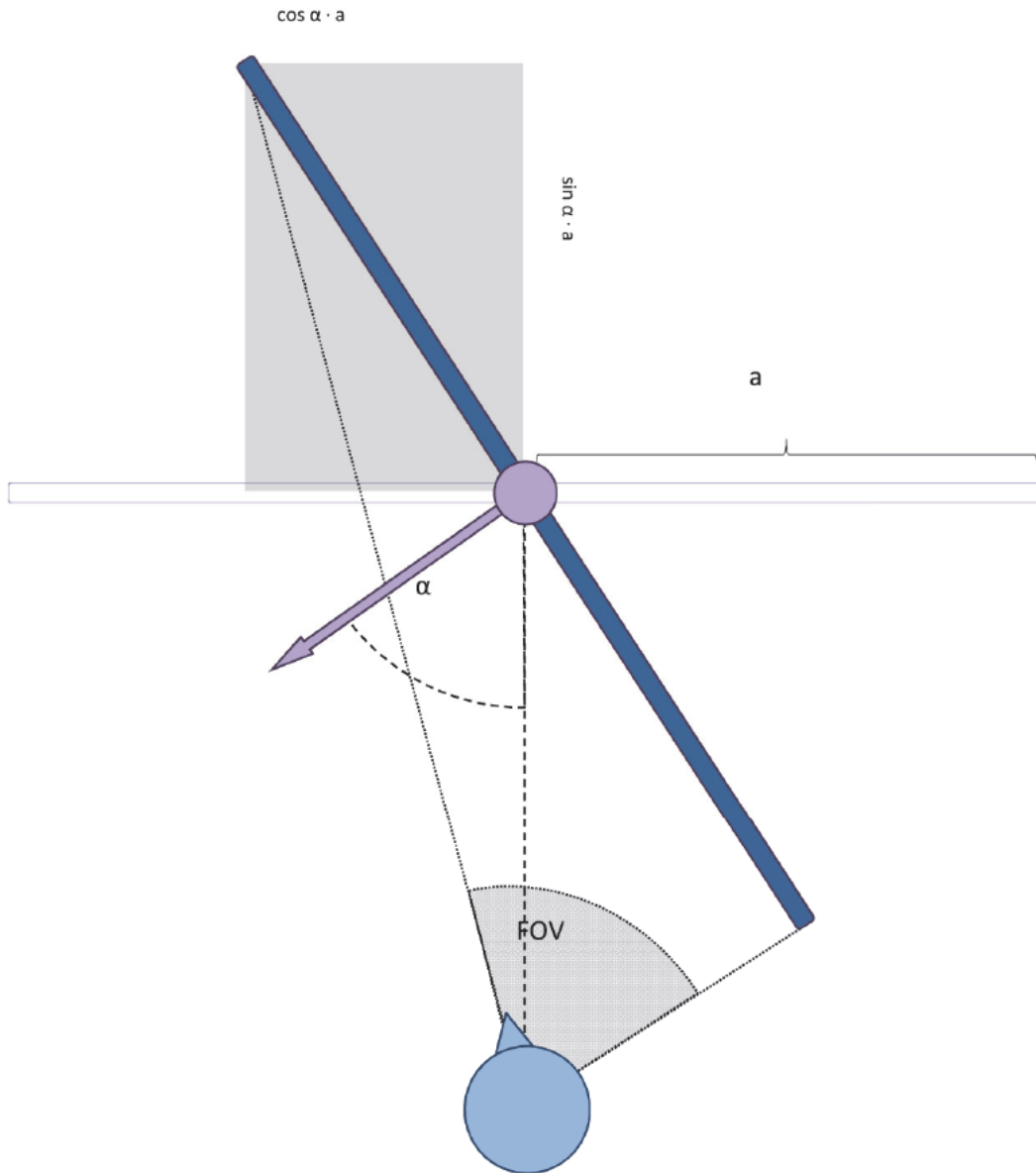


Abbildung A.1.: Aus [Rad14] übernommene Abbildung, welche die für die Berechnung der sichtbaren Displayfläche verwendeten Größen zeigt. Das Display ist dunkelblau in der Mitte und der Nutzer ist unten als hellblauer Kreis dargestellt.

$$fov_v(\gamma_v) = \begin{cases} \frac{1}{60^\circ} \cdot \gamma_v & , \gamma_v < 60^\circ \\ \cos\left(\frac{9}{12} \cdot \gamma_v - \frac{9 \cdot 60}{12}\right) & , \gamma_v \geq 60^\circ \end{cases}$$

Die Werte fov_v und fov_h stellen die bzgl. des vertikalen und horizontalen Sichtfeldes eines Nutzers bewertete Sichtbarkeit des Displays dar. Das Minimum beider Werte bestimmt die Sichtbarkeit eines Displays fov für diesen Nutzer.

3. Für die Sichtbarkeitsberechnung über alle Nutzer werden die fov -Werte von allen Nutzern normiert und miteinander multipliziert. Im Folgenden sei diese Gesamtsichtbarkeit des Displays fov_D . Für die Sichtbarkeit eines einzelnen Views wird dieser Wert zusätzlich mit einem Skalierungsfaktor multipliziert. Der Skalierungsfaktor sf ist das Verhältnis aus der Anzahl n an Views, die auf dem betrachteten Display angezeigt werden sollen und der Gesamtanzahl an Views m .

$$sf = 1 - \left(\frac{n}{m}\right)$$

Adaption des Ansatzes von Radloff:

Der unter 3. eingeführt Skalierungsfaktor wurde in der vorliegenden Dissertation angepasst, um die unterschiedliche Skalierung der Views genauer abzuschätzen. Ähnlich wie für die in Abschnitt 4.2 eingeführte Heuristik wird angenommen, dass die verfügbare Fläche eines Displays vollständig auf die auf ihm anzuzeigenden Views aufgeteilt werden kann. Zusätzlich wird angenommen, dass diese Verteilung exakt nach den Relevanzwerten der Views erfolgen kann. Der angepasste Skalierungsfaktor $sf(v)$ für eine View v mit dem Relevanzwert $r(v)$ ist somit:

$$sf(v) = \frac{r(v)}{\sum_{v_i \in V_D} r(v_i)}$$

Hierbei bezeichnet V_D die Menge aller Views v_i die auf dem Display D angezeigt werden sollen. Diese Information ist bereits für eine View-Display-Zuordnung bekannt und wird beim Bestimmen der entwickelten Heuristik mitberechnet. Der Sichtbarkeitsfaktor $d(v)$ für eine View v auf einem Display ist somit:

$$d(v) = sf(v) \cdot fov_D$$

Obwohl der Wert $d(v)$ von vielen verschiedenen Faktoren abhängig ist, ist er bzgl. des in Abschnitt 4.2 beschriebenen Optimierungsproblems eine Konstante. (Das Optimierungsproblem ist somit weiterhin ein quadratisches Optimierungsproblem mit linearen Constraints (QPLC) für das sich der Zusammenhang zwischen den Variablen durch eine positive semi-definite Matrix beschreiben lässt und somit der modifizierte Simplex-Algorithmus anwendbar ist.)

A. Layout-Berechnung

Die Werte für fov_D bzw. sf können sich allerdings zwischen zwei aufeinander folgenden Layout-Berechnungen verändern, wenn sich die Nutzer in der Zwischenzeit im Raum bewegt haben oder sich die Anzahl der Views verändert hat. Der Ansatz von Radloff sieht vor, dass fov_D möglichst oft aktualisiert wird, um Views nicht durch Nutzer zu verdecken, die sich vor den Displays entlang bewegen (siehe Abbildung 2.5 auf Seite 31). Bezüglich der in dieser Arbeit entwickelten Konzepte wäre dieses Vorgehen aber problematisch. Angenommen ein Redner (bei einer sonst geringen Anzahl an Zuhörern) möchte bspw. zu einer View mit einem geringen Relevanzwert gehen, so würde sich beim Bewegen durch den Raum fov_D verändern und eine neue Layout-Berechnung angestoßen. In der aktualisierten View-Display-Verteilung würde die ausgesuchte View nun wieder auf einem Display erscheinen, das vom Redner weit entfernt ist. Der Redner hätte es somit schwer Views zu erreichen, die einen geringen Relevanzwert haben.

Aus diesem Grund wird die Aktualisierung von fov_D im entwickelten System nicht ständig vorgenommen, sondern nur dann, wenn große Änderungen der durchschnittlichen Nutzerpositionen auftreten. Dies ist typischerweise am Anfang und am Ende einer Präsentation bzw. Diskussion der Fall oder wenn eine größere Gruppe von Nutzern nachträglich den Raum betritt oder verlässt. Durch das beschriebene Vorgehen bleibt die View-Verteilung bzgl. kleiner Positionsänderungen oder Bewegungen einzelner Nutzer relativ stabil, was häufige bzw. während einer Präsentation ungewollte View-Umpositionierungen vermeidet. Situationsabhängige Aktualisierungen der View-Verteilung und Layouts werden dagegen nun durch die neu entwickelten und in Abschnitt 6.2.2 vorgestellten Ansätze übernommen.

Im Gegensatz zu fov_D wird der Faktor sf immer dann aktualisiert, wenn sich die anzuzeigende View-Zusammenstellung ändert oder Anpassungen am Präsentationsgraph vorgenommen werden. Auf diese Weise werden sofort neue Nutzervorgaben bei der View-Verteilung und Anordnung berücksichtigt.

Literaturverzeichnis

- [Aar06] E. Aarts. “Into Ambient Intelligence”. In: *True Visions The Emergence of Ambient Intelligence*. Springer, 2006 (Zitiert auf den Seiten 13, 23).
- [Aeh+13] M. Aehnelt, S. Bader, G. Ruscher, F. Krüger, B. Urban, and T. Kirste. “Situation Aware Interaction with Multi-modal Business Applications in Smart Environments”. In: *Human Interface and the Management of Information. Information and Interaction for Learning, Culture, Collaboration and Business*. Springer, 2013, pp. 413–422 (Zitiert auf Seite 24).
- [ANA10] J. Augusto, H. Nakashima, and H. Aghajan. “Ambient Intelligence and Smart Environments: A State of the Art”. In: *Handbook of Ambient Intelligence and Smart Environments*. Springer, 2010, pp. 3–31 (Zitiert auf Seite 23).
- [AR11] J. K. Aggarwal and M. S. Ryoo. “Human Activity Analysis: A Review”. In: *ACM Computing Surveys* 43.3 (2011) (Zitiert auf Seite 125).
- [Aub+12] D. Auber, D. Archambault, R. Bourqui, A. Lambert, M. Mathiaut, P. Mary, M. Delest, J. Dubois, and G. Melançon. *The Tulip 3 Framework: A Scalable Software Library for Information Visualization Applications Based on Relational Data*. Research Report RR-7860. INRIA, 2012, p. 31 (Zitiert auf Seite 107).
- [BB04] J. T. Biehl and B. P. Bailey. “ARIS: An Interface for Application Relocation in an Interactive Space”. In: *Proc. of Graphics Interface 2004*. GI '04. Canadian Human-Computer Communications Society, 2004, pp. 107–116 (Zitiert auf Seite 34).
- [BB06] J. T. Biehl and B. P. Bailey. “Improving Interfaces for Managing Applications in Multiple-device Environments”. In: *Proc. of the Working Conference on Advanced Visual Interfaces*. AVI '06. ACM, 2006, pp. 35–42 (Zitiert auf Seite 34).
- [BBF15] A. Benmansour, A. Bouchachia, and M. Feham. “Multioccupant Activity Recognition in Pervasive Smart Home Environments”. In: *ACM Computing Surveys* 48.3 (Dec. 2015), pp. 1–36 (Zitiert auf Seite 126).

- [BDR07] M. Baldauf, S. Dustdar, and F. Rosenberg. “A Survey on Context-aware Systems”. In: *Int. J. Ad Hoc and Ubiquitous Computing* 2.4 (2007), p. 263 (Zitiert auf Seite 23).
- [BE14] S. K. Badam and N. Elmqvist. “PolyChrome: A Cross-Device Framework for Collaborative Web Visualization”. In: *Proc. of the Ninth ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’14. ACM, 2014, pp. 109–118 (Zitiert auf Seite 73).
- [Ber+10] L. Bergman, J. Lu, R. Konuru, J. MacNaught, and D. Yeh. “Outline Wizard: Presentation Composition and Search”. In: *Proc. of the 15th International Conference on Intelligent User Interfaces*. IUI ’10. ACM, 2010, pp. 209–218 (Zitiert auf Seite 46).
- [Ber81] J. Bertin. *Graphics and Graphic Information-Processing*. Walter de Gruyter, 1981 (Zitiert auf Seite 81).
- [Ber94] T. Berlage. “A Selective Undo Mechanism for Graphical User Interfaces Based on Command Objects”. In: *ACM Trans. Comput.-Hum. Interact.* 1.3 (Sept. 1994), pp. 269–294 (Zitiert auf Seite 97).
- [Bez+10] A. Bezerianos, F. Chevalier, P. Dragicevic, N. Elmqvist, and J. D. Fekete. “Graphdice: A System for Exploring Multivariate Social Networks”. In: *Proc. of the 12th Eurographics / IEEE - VGTC Conference on Visualization*. EuroVis’10. The Eurographs Association & John Wiley & Sons, Ltd., 2010, pp. 863–872 (Zitiert auf Seite 107).
- [BHJ09] M. Bastian, S. Heymann, and M. Jacomy. “Gephi: An Open Source Software for Exploring and Manipulating Networks”. In: *International AAAI Conference on Weblogs and Social Media* (2009) (Zitiert auf Seite 107).
- [Bie+08] J. Biehl, W. Baker, B. Bailey, D. Tan, K. Inkpen, and M. Czerwinski. “Impromptu: A New Interaction Framework for Supporting Collaboration in Multiple Display Environments and its Field Evaluation for Co-located Software Development”. In: *Proc. CHI 08*. ACM, 2008 (Zitiert auf den Seiten 34, 35).
- [Bit+11] A. T. Bittig, F. Haack, C. Maus, and A. M. Uhrmacher. “Adapting Rule-based Model Descriptions for Simulating in Continuous and Hybrid Space”. In: *Proc. of the 9th International Conference on Computational Methods in Systems Biology*. CMSB ’11. ACM, 2011, pp. 161–170 (Zitiert auf Seite 82).
- [Bit+12] A. T. Bittig, A. M. Uhrmacher, C. Matschegewski, and J. B. Nebe. “Spatial simulation of actin filament dynamics on structured surfaces”. In: *Proc. of the Winter Simulation Conference*. WSC ’12. Winter Simulation Conference, 2012, p. 364 (Zitiert auf Seite 82).

- [BK12] S. Bader and T. Kirste. *An Overview of the Helferlein-System*. Tech. rep. CS-03-12 under ISSN 0944-5900. Institut für Informatik, Universität Rostock, 2012 (Zitiert auf den Seiten 25–27).
- [BLI08] A. Behera, D. Lalanne, and R. Ingold. “DocMIR: An automatic document-based indexing system for meeting retrieval”. In: *Multimedia Tools and Applications* 37.2 (2008), pp. 135–167 (Zitiert auf Seite 50).
- [BN12] S. Bader and M. Nyolt. “A context-aware publish-subscribe middleware for distributed smart environments”. In: *IEEE International Conference on Pervasive Computing and Communications Workshops* (2012), pp. 100–104 (Zitiert auf Seite 27).
- [BNK13] S. Bader, R. Nicolay, and T. Kirste. “Agent-Based Proactive Support in Smart Environments”. In: *9th International Conference on Intelligent Environments (IE)*. 2013, pp. 220–223 (Zitiert auf Seite 24).
- [BRK10] S. Bader, G. Ruscher, and T. Kirste. “A Middleware for Rapid Prototyping Smart Environments: Experiences in Research and Teaching”. In: *Proc. of the 12th ACM International Conference on Ubiquitous Computing*. ACM, 2010, pp. 355–356 (Zitiert auf Seite 26).
- [BT12] S. Bader and G. R. and Thomas Kirste. “Multimodal Interaction in Dynamic and Heterogeneous Smart Environments”. In: *10th International Conference on Pervasive Computing* (2012), p. 8 (Zitiert auf Seite 143).
- [Bur+08] C. Burghardt, C. Reisse, T. Heider, M. Giersich, and T. Kirste. “Implementing Scenarios in a Smart Learning Environment”. In: *Sixth Annual IEEE International Conference on Pervasive Computing and Communications*. 2008, pp. 377–382 (Zitiert auf Seite 24).
- [BV04] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004 (Zitiert auf Seite 69).
- [Cao+14] L. Cao, W. Chen, X. Zhang, and K. Huang. “A Smart Meeting Management System With Video Based Seat Detection”. In: *Proc. of International Conference on Internet Multimedia Computing and Service*. ICIMCS ’14. ACM, 2014, pp. 232–236 (Zitiert auf Seite 125).
- [CD07] D. J. Cook and S. K. Das. “How Smart Are Our Environments? An Updated Look at the State of the Art”. In: *Pervasive Mob. Comput.* 3.2 (Mar. 2007), pp. 53–73 (Zitiert auf den Seiten 23, 24).
- [CF07] A. G. Cass and C. S. T. Fernandes. “Using Task Models for Cascading Selective Undo”. In: *Task Models and Diagrams for Users Interface Design: 5th International Workshop*. Springer, 2007, pp. 186–201 (Zitiert auf Seite 97).

- [Cha+08] M. Chan, D. Estève, C. Escriba, and E. Campo. “A review of smart homes - Present state and future challenges”. In: *Computer methods and programs in biomedicine* 91.1 (2008), pp. 55–81 (Zitiert auf Seite 23).
- [Che+09] Y. Cheng, F. He, S. Jing, and Z. Huang. “An multiuser Undo/Redo method for replicated collaborative modeling systems”. In: *13th International Conference on Computer Supported Cooperative Work in Design*. 2009, pp. 185–190 (Zitiert auf Seite 50).
- [Che+12a] L. Chen, J. Hoey, C. D. Nugent, D. J. Cook, and Z. Yu. “Sensor-Based Activity Recognition”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 42.6 (2012), pp. 790–808 (Zitiert auf Seite 126).
- [Che+12b] Y. Cheng, X. Cai, F. He, and D. Zhang. “A selective undo/redo method in 3D collaborative modeling environment”. In: *IEEE 16th International Conference on Computer Supported Cooperative Work in Design*. 2012, pp. 102–108 (Zitiert auf Seite 51).
- [Chu+14] H. Chung, C. North, J. Self, S. Chu, and F. Quek. “VisPorter: facilitating information sharing for collaborative sensemaking on multiple displays”. In: *Personal and Ubiquitous Computing* 18.5 (2014), pp. 1169–1186 (Zitiert auf Seite 35).
- [Chu+15] H. Chung, C. North, S. Joshi, and J. Chen. “Four Considerations for Supporting Visual Analysis in Display Ecologies”. In: *Proc. of Visual Analytics Science and Technology (VAST)*. IEEE, 2015 (Zitiert auf den Seiten 14, 15, 24, 35–37, 71).
- [Crn+14] T. Crnovrsanin, C. W. Muelder, R. Faris, D. Felmlee, and K.-L. Ma. “Visualization techniques for categorical analysis of social networks with multiple edge sets”. In: *Social Networks* 37 (2014), pp. 56–64 (Zitiert auf Seite 107).
- [Dai+08] P. Dai, H. Di, L. Dong, L. Tao, and G. Xu. “Group Interaction Analysis in Dynamic Context”. In: *Trans. Sys. Man Cyber. Part B* 38.1 (2008), pp. 275–282 (Zitiert auf den Seiten 126–128).
- [Den+13] T. Deng, L. Zhao, H. Wang, Q. Liu, and L. Feng. “ReFinder: A Context-Based Information Refinding System”. In: *IEEE Transactions on Knowledge and Data Engineering* 25.9 (2013), pp. 2119–2132 (Zitiert auf Seite 46).
- [DMS14] J. Doboš, N. J. Mitra, and A. Steed. “3D Timeline: Reverse Engineering of a Part-based Provenance from Consecutive 3D Models”. In: *Comput. Graph. Forum* 33.2 (2014), pp. 135–144 (Zitiert auf den Seiten 96, 97).
- [Doh+10] A. Dohr, R. Modre-Opsrian, M. Drobics, D. Hayn, and G. Schreier. “The Internet of Things for Ambient Assisted Living”. In: *Proc. of the 7th Inter-*

- national Conference on Information Technology: New Generations*. IEEE, 2010, pp. 804–809 (Zitiert auf Seite 23).
- [DWL15] D. Dou, H. Wang, and H. Liu. “Semantic data mining: A survey of ontology-based approaches”. In: *IEEE International Conference on Semantic Computing (ICSC)*. 2015, pp. 244–251 (Zitiert auf Seite 45).
- [Dyr+11] M. Dyrba, R. Nicolay, S. Bader, and T. Kirste. “Evaluation of two Control Systems for Smart Environments”. In: *Proc. of the 8th International ICST Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services - 3rd Workshop on Context Systems Design, Evaluation and Optimisation*. 2011, pp. 1–12 (Zitiert auf Seite 24).
- [EBM05] C. Endres, A. Butz, and A. MacWilliams. “A Survey of Software Infrastructures and Frameworks for Ubiquitous Computing”. In: *Mob. Inf. Syst.* 1.1 (2005), pp. 41–80 (Zitiert auf Seite 26).
- [EE04] J. Elf and M. Ehrenberg. “Spontaneous separation of bi-stable biochemical systems into spatial domains of opposite phases”. In: *Systems Biology* 1.2 (2004), pp. 230–236 (Zitiert auf Seite 82).
- [Eic+13] C. Eichner, A. Bittig, H. Schumann, and C. Tominski. “Feature-Based Visual Analytics for Studying Simulations of Dynamic Bi-Stable Spatial Systems”. In: *Proc. of the EuroVis Workshop on Visual Analytics (EuroVA)*. Eurographics Association, 2013 (Zitiert auf den Seiten 19, 81).
- [Eic+14] C. Eichner, A. Bittig, H. Schumann, and C. Tominski. “Analyzing simulations of biochemical systems with feature-based visual analytics”. In: *Computers & Graphics* 38 (2014), pp. 18–26 (Zitiert auf den Seiten 19, 81).
- [Eic+15] C. Eichner, T. Nocke, H.-J. Schulz, and H. Schumann. “Interactive Presentation of Geo-Spatial Climate Data in Multi-Display Environments”. In: *ISPRS International Journal of Geo-Information* 4.2 (2015), p. 493 (Zitiert auf den Seiten 17–19, 91, 106).
- [Eic+16] C. Eichner, S. Gladisch, H. Schumann, and C. Tominski. “Direct Visual Editing of Node Attributes in Graphs.” In: *Informatics* 3 (2016), no. 4: 17 (Zitiert auf den Seiten 111, 112).
- [EK05] J. L. Encarnacao and T. Kirste. “Ambient Intelligence: Towards Smart Appliance Ensembles”. In: *From Integrated Publication and Information Systems to Information and Knowledge Environments*. Springer, 2005 (Zitiert auf den Seiten 13, 23, 24).
- [ENS15] C. Eichner, M. Nyolt, and H. Schumann. “A novel infrastructure for supporting display ecologies”. In: *Proc. of the 11th International Symposium on Visual Computing (ISVC)*. Las Vegas, NV, USA, Dec. 2015 (Zitiert auf den Seiten 17, 20).

- [EW14] S. van den Elzen and J. J. van Wijk. “Multivariate Network Exploration and Presentation: From Detail to Overview via Selections and Aggregations”. In: *IEEE Transactions on Visualization and Computer Graphics* 20.12 (2014), pp. 2310–2319 (Zitiert auf Seite 107).
- [Fai+16] A. J. Fairchild, S. P. Champion, A. S. Garcia, R. Wolff, T. Fernando, and D. J. Roberts. “A Mixed Reality Telepresence System for Collaborative Space Operation”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 99 (2016) (Zitiert auf Seite 144).
- [FBR15] C. F. Freitas, J. Barroso, and C. Ramos. “A Survey on Smart Meeting Rooms and Open Issues”. In: *International Journal of Smart Home* 9 (2015), pp. 13–20 (Zitiert auf Seite 14).
- [FD13] M. Frisch and R. Dachsel. “Visualizing offscreen elements of node-link diagrams”. In: *Information Visualization* 12.2 (2013), pp. 133–162 (Zitiert auf Seite 111).
- [FFH00] J. Flachsbar, D. Franklin, and K. Hammond. “Improving Human Computer Interaction in a Classroom Environment Using Computer Vision”. In: *Proc. of the 5th International Conference on Intelligent User Interfaces*. ACM, 2000, pp. 86–93 (Zitiert auf Seite 23).
- [GH07] M. Giersich and T. Heider. “Team Assistance in Smart Meeting Rooms”. In: *Ubicomp 2007 Workshop Proc.: Workshop on Embodied Meeting Support: Mobile, Tangible, Senseable Interaction in Smart Environments*. 2007, pp. 123–128 (Zitiert auf Seite 30).
- [Gla16] S. Gladisch. “Supporting Graph Editing in Visual Representations”. PhD thesis. Fakultät für Informatik und Elektrotechnik, Universität Rostock, 2016 (Zitiert auf den Seiten 111, 112).
- [Gou+12] L. Gou, X. Zhang, A. Luo, and P. F. Anderson. “SocialNetSense: Supporting sensemaking of social and structural features in networks with interactive visualization”. In: *IEEE Conference on Visual Analytics Science and Technology*. 2012, pp. 133–142 (Zitiert auf Seite 96).
- [Gra+16] S. Gratzl, A. Lex, N. Gehlenborg, N. Cosgrove, and M. Streit. “From Visual Exploration to Storytelling and Back Again”. In: *Computer Graphics Forum (EuroVis ’16)* (2016) (Zitiert auf den Seiten 71–73).
- [GT14] S. Gladisch and C. Tominski. “Toward Integrated Exploration and Manipulation of Data Attributes in Graphs.” In: *Poster at IEEE Conference on Information Visualization* (2014) (Zitiert auf Seite 107).
- [Haa+13] F. Haack, K. Burrage, R. Redmer, and A. M. Uhrmacher. “Studying the Role of Lipid Rafts on Protein Receptor Bindings with Cellular Automata”.

- In: *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 10.3 (2013), pp. 760–770 (Zitiert auf Seite 82).
- [Hee+08] J. Heer, J. Mackinlay, C. Stolte, and M. Agrawala. “Graphical Histories for Visualization: Supporting Analysis, Communication, and Evaluation”. In: *IEEE Transactions on Visualization and Computer Graphics* 14.6 (2008), pp. 1189–1196 (Zitiert auf den Seiten 50, 96, 97).
- [HK05] T. Heider and T. Kirste. “Multimodal Appliance Cooperation Based on Explicit Goals: Concepts & Potentials”. In: *Proc. of the 2005 Joint Conference on Smart Objects and Ambient Intelligence: Innovative Context-aware Services: Usages and Technologies*. ACM, 2005, pp. 271–276 (Zitiert auf Seite 23).
- [HK07] T. Heider and T. Kirste. “Automatic vs. Manual Multi-Display Configuration: A Study of User Performance in a Semi-Cooperative Task Setting”. In: *Proc. of the 21st British HCI Group Annual Conference on People and Computers*. Vol. 2. British Computer Society, 2007, pp. 43–46 (Zitiert auf den Seiten 30, 31).
- [HK08] T. Heider and T. Kirste. “Evaluating the effect of automatic display management on user performance in a Smart Meeting Room”. In: *3. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2008)*. 2008 (Zitiert auf Seite 31).
- [HKD16] T. Horak, U. Kister, and R. Dachsel. “Presenting Business Data: Challenges During Board Meetings in Multi-Display Environments”. In: *Proc. of the 2016 ACM on Interactive Surfaces and Spaces*. ACM, 2016, pp. 319–324 (Zitiert auf Seite 144).
- [HNR02] R. Heinrich, B. G. Neel, and T. A. Rapoport. “Mathematical Models of Protein Kinase Signal Transduction”. In: *Molecular Cell* 9.5 (2002), pp. 957–970 (Zitiert auf Seite 82).
- [HR13] T. W. Hill and A. Ravindran. “On programming with absolute-value functions”. In: *Journal of Optimization Theory and Applications* 17.1 (2013), pp. 181–183 (Zitiert auf Seite 69).
- [HSS15] S. Hadlak, H. Schumann, and H.-J. Schulz. “A Survey of Multi-faceted Graph Visualization”. In: *Eurographics Conf. on Visualization - STARs*. The Eurographics Association, 2015 (Zitiert auf Seite 107).
- [HTK08] S. Hasegawa, A. Tanida, and A. Kashihara. “A Presentation Support Service Using Presentation Semantics”. In: *Eighth IEEE International Conference on Advanced Learning Technologies*. 2008, pp. 104–105 (Zitiert auf Seite 46).

- [Hu+14] C. Hu, Z. Xu, Y. Liu, L. Mei, L. Chen, and X. Luo. “Semantic Link Network-Based Model for Organizing Multimedia Big Data”. In: *IEEE Transactions on Emerging Topics in Computing* 2.3 (2014), pp. 376–387 (Zitiert auf Seite 46).
- [JDK10] I. Jusufi, Y. Dingjie, and A. Kerren. “The Network Lens: Interactive Exploration of Multivariate Networks Using Visual Filtering”. In: *14th International Conference Information Visualisation*. 2010, pp. 35–42 (Zitiert auf Seite 107).
- [JFW02] B. Johanson, A. Fox, and T. Winograd. “The Interactive Workspaces Project: Experiences with Ubiquitous Computing Rooms”. In: *IEEE Pervasive Computing* 1.2 (2002), pp. 67–74 (Zitiert auf Seite 23).
- [JH13] M. R. Jakobsen and K. Hornbæk. “Interactive Visualizations on Large and Small Displays: The Interrelation of Display Size, Information Space, and Scale”. In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (2013), pp. 2336–2345 (Zitiert auf Seite 29).
- [JM05] A. Jaimes and J. Miyazaki. “Building a Smart Meeting Room: From Infrastructure to the Video Gap (Research and open Issues)”. In: *21st International Conference on Data Engineering Workshops* (2005), p. 1173 (Zitiert auf Seite 24).
- [Joh+02] B. Johanson, G. Hutchins, T. Winograd, and M. Stone. “PointRight: Experience with Flexible Input Redirection in Interactive Workspaces”. In: *Proc. of the 15th Annual ACM Symposium on User Interface Software and Technology*. ACM, 2002, pp. 227–234 (Zitiert auf Seite 35).
- [Jus+12] I. Jusufi, C. Klukas, A. Kerren, and F. Schreiber. “Guiding the Interactive Exploration of Metabolic Pathway Interconnections”. In: *Information Visualization* 11.2 (2012), pp. 136–150 (Zitiert auf Seite 111).
- [KH13] J. Kehrler and H. Hauser. “Visualization and Visual Analysis of Multifaceted Scientific Data: A Survey”. In: *IEEE Transactions on Visualization and Computer Graphics* 19.3 (2013), pp. 495–513 (Zitiert auf Seite 81).
- [Kha15] S. Khan. “Visualization Assisted Enterprise Search Engine”. PhD thesis. Department of Engineering Science, University of Oxford, 2015 (Zitiert auf Seite 44).
- [KLK12] N. W. Khobragade, N. K. Lamba, and P. G. Khot. “Alternative Approach to Wolfe’s Modified Simplex Method for Quadratic Programming Problems”. In: *Int. J Latest Trend Math* 2, No. 1 (2012) (Zitiert auf Seite 69).
- [KOS09] D. Kitayama, A. Otani, and K. Sumiya. “A Scene Extracting Method Based on Structural and Semantic Analysis of Presentation Content Archi-

- ves”. In: *7th International Conference on Creating, Connecting and Collaborating through Computing*. 2009, pp. 128–135 (Zitiert auf Seite 46).
- [KPW14] A. Kerren, H. C. Purchasem, and M. O. Ward. “Multivariate Network Visualization”. In: *Lecture Notes in Computer Science* (2014) (Zitiert auf Seite 107).
- [Krü+12] F. Krüger, K. Yordanova, C. Burghardt, and T. Kirste. “Towards Creating Assistive Software by Employing Human Behavior Models”. In: *Journal of Ambient Intelligence and Smart Environments* 4.3 (2012), pp. 209–226 (Zitiert auf den Seiten 27, 127, 128).
- [Krü+13] F. Krüger, K. Yordanova, A. Hein, and T. Kirste. “Plan Synthesis for Probabilistic Activity Recognition”. In: *Proc. of the 5th International Conference on Agents and Artificial Intelligence*. 2013, pp. 283–288 (Zitiert auf den Seiten 27, 126).
- [Krü11] F. Krüger. “A Context-Aware Proactive Controller for Smart Environments”. In: *i-com Zeitschrift für interaktive und kooperative Medien* 10 (2011), pp. 41–48 (Zitiert auf Seite 24).
- [Krü16] F. Krüger. “Activity, Context, and Plan Recognition with Computational Causal Behavior Models”. PhD thesis. Fakultät für Informatik und Elektrotechnik, Universität Rostock, 2016 (Zitiert auf Seite 126).
- [Kwo+14] B. C. Kwon, F. Stoffel, D. Jäckle, B. Lee, and D. A. Keim. “VisJockey: Enriching Data Stories through Orchestrated Interactive Visualization”. In: *Computation + Journalism Symposium*. Brown Institute for Media Innovation, 2014 (Zitiert auf Seite 73).
- [LBT08] J. Lanir, K. S. Booth, and A. Tang. “MultiPresenter: A Presentation System for (Very) Large Display Surfaces”. In: *Proc. of the 16th ACM International Conference on Multimedia*. ACM, 2008, pp. 519–528 (Zitiert auf den Seiten 35, 73).
- [Lee+15] B. Lee, N. H. Riche, P. Isenberg, and S. Carpendale. “More Than Telling a Story: Transforming Data into Visually Shared Stories”. In: *IEEE Computer Graphics and Applications* 35.5 (2015), pp. 84–90 (Zitiert auf den Seiten 71–73).
- [Leh+10] G. Lehmann, A. Rieger, M. Blumendorf, and S. Albayrak. “A 3-Layer Architecture for Smart Environment Models”. In: *8th IEEE International Conference on Pervasive Computing and Communications Workshops*. 2010, pp. 636–641 (Zitiert auf Seite 24).
- [Lei11] R. Leistikow. “Multi-Agent Strategy Synthesis in Smart Meeting Environments”. In: *Proc. of 2nd International Symposium on Ambient Intelligence*. Springer, 2011 (Zitiert auf Seite 24).

- [Lex+12] A. Lex, M. Streit, H.-J. Schulz, C. Partl, D. Schmalstieg, P. Park, and N. Gehlenborg. “StratomeX: Visual Analysis of Large-Scale Heterogeneous Genomics Data for Cancer Subtype Characterization”. In: *Computer Graphics Forum* 31.3 (2012), pp. 1175–1184 (Zitiert auf Seite 73).
- [Lia+06] W. Liao, W. Zhang, Z. Zhu, Q. Ji, and W. D. Gray. “Toward a Decision-theoretic Framework for Affect Recognition and User Assistance”. In: *Int. J. Hum.-Comput. Stud.* 64.9 (2006), pp. 847–873 (Zitiert auf Seite 129).
- [Liu+05] Q. Liu, D. Kimber, F. Zhao, and J. Huang. “Framework for effective use of multiple displays”. In: *Proc. of SPIE International Symposium ITCOM 2005 on Multimedia Systems and Applications* (2005) (Zitiert auf Seite 35).
- [LL13] Ó. D. Lara and M. A. Labrador. “A survey on human activity recognition using wearable sensors”. In: *Communications Surveys Tutorials, IEEE* 15.3 (2013), pp. 1192–1209 (Zitiert auf Seite 126).
- [LST13] C. Lauterbach, A. Steinhage, and A. Techmer. “A Large-Area Sensor System Underneath the Floor for Ambient Assisted Living Applications”. In: *Pervasive and Mobile Sensing and Computing for Healthcare*. Vol. 2. Springer, 2013, pp. 69–87 (Zitiert auf Seite 27).
- [Mar12] A. L. Marc Streit Hans-Jörg Schulz. *Connecting the Dots - Showing Relationships in Data and Beyond*. VisWeek Tutorial. 2012 (Zitiert auf Seite 107).
- [Men+98] C. Meng, M. Yasue, A. Imamiya, and X. Mao. “Visualizing Histories for Selective Undo and Redo”. In: *Proc. of the 3rd Asia Pacific Computer and Human Interaction*. 1998, pp. 459–464 (Zitiert auf Seite 97).
- [Mor12] K. Moreland. “Redirecting Research in Large-Format Displays for Visualization”. In: *IEEE Symposium on Large Data Analysis and Visualization*. 2012, pp. 91–95 (Zitiert auf Seite 29).
- [Moz05] M. C. Mozer. “Lessons from an Adaptive Home”. In: *Smart Environments*. John Wiley & Sons, Inc., 2005, pp. 271–294 (Zitiert auf Seite 23).
- [MP12] D. S. Martins and M. d. G. C. Pimentel. “Browsing Interaction Events in Recordings of Small Group Activities via Multimedia Operators”. In: *Proc. of the 18th Brazilian Symposium on Multimedia and the Web*. ACM, 2012, pp. 245–252 (Zitiert auf Seite 50).
- [MuS15] MuSAMA. *Multimodal Smart Appliance Ensembles for Mobile Applications*. 2015. URL: <http://www.musama.de> (Zitiert auf den Seiten 13, 23).
- [Mwa+16] F. Mwalongo, M. Krone, G. Reina, and T. Ertl. “State-of-the-Art Report in Web-based Visualization”. In: *Computer Graphics Forum* (2016) (Zitiert auf den Seiten 74, 91).

- [NC14] M. Nancel and A. Cockburn. “Causality: A Conceptual Model of Interaction History”. In: *Proc. of the 32nd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 2014, pp. 1777–1786 (Zitiert auf Seite 96).
- [Nie93a] J. Nielsen. *Response Times: The 3 Important Limits*. 1993. URL: <https://www.nngroup.com/articles/response-times-3-important-limits/> (Zitiert auf Seite 69).
- [Nie93b] J. Nielsen. *Usability Engineering*. Morgan Kaufmann Publishers Inc., 1993 (Zitiert auf den Seiten 69, 70).
- [Nor98] D. A. Norman. *The Invisible Computer*. Cambridge, MA, USA: MIT Press, 1998 (Zitiert auf Seite 23).
- [OWS15] S. Ohl, M. Willert, and O. Staadt. “Latency in Distributed Acquisition and Rendering for Telepresence Systems”. In: *IEEE Transactions on Visualization and Computer Graphics* 21.12 (2015), pp. 1442–1448 (Zitiert auf Seite 144).
- [Pap+15] C. Papadopoulos, S. Mirhosseini, I. Gutenko, K. Petkov, A. E. Kaufman, and B. Laha. “Scalability limits of large immersive high-resolution displays”. In: *IEEE Virtual Reality*. 2015, pp. 11–18 (Zitiert auf Seite 29).
- [PK94] A. Prakash and M. J. Knister. *Undoing Actions in Collaborative Work: Framework and Experience*. Tech. rep. 1994 (Zitiert auf Seite 51).
- [PST04] T. Prante, N. Streitz, and P. Tandler. “Roomware: Computers Disappear and Interaction Evolves”. In: *Computer* 37.12 (Dec. 2004), pp. 47–54 (Zitiert auf Seite 23).
- [Rad+11] A. Radloff, M. Luboschik, M. Sips, and H. Schumann. “Supporting Display Scalability by Redundant Mapping”. In: *Proc. of the 7th International Conference on Advances in Visual Computing*. Springer, 2011, pp. 472–483 (Zitiert auf den Seiten 29, 32, 65).
- [Rad+12] A. Radloff, A. Lehmann, O. Staadt, and H. Schumann. “Smart Interaction Management: An Interaction Approach for Smart Meeting Rooms”. In: *8th International Conference on Intelligent Environments*. 2012, pp. 228–235 (Zitiert auf den Seiten 33, 35, 143).
- [Rad+15] A. Radloff, C. Tominski, T. Nocke, and H. Schumann. “Supporting Presentation and Discussion of Visualization Results in Smart Meeting Rooms”. In: *The Visual Computer* 31.9 (2015), pp. 1271–1286 (Zitiert auf den Seiten 31, 33).

- [Rad14] A. Radloff. “Smart Views in Smart Meeting Rooms”. PhD thesis. Fakultät für Informatik und Elektrotechnik, Universität von Rostock, 2014 (Zitiert auf den Seiten 24, 29, 31–33, 39, 65, 73, 143, 147, 148).
- [Rag+16] E. D. Ragan, A. Endert, J. Sanyal, and J. Chen. “Characterizing Provenance in Visualization and Data Analysis: An Organizational Framework of Provenance Types and Purposes”. In: *IEEE Transactions on Visualization and Computer Graphics* 22.1 (2016), pp. 31–40 (Zitiert auf den Seiten 49, 97, 100, 103).
- [Rei10] K. Reichenberger. *Kompendium semantische Netze*. Springer Berlin Heidelberg, 2010 (Zitiert auf den Seiten 44, 46).
- [RFS12] A. Radloff, G. Fuchs, and H. Schumann. “Supporting Visual Analysis in Smart Meeting Rooms”. In: *Euro VA 2012*. 2012 (Zitiert auf Seite 32).
- [RG99] M. Ressel and R. Gunzenhäuser. “Reducing the Problems of Group Undo”. In: *Proc. of the International ACM SIGGROUP Conference on Supporting Group Work*. ACM, 1999, pp. 131–139 (Zitiert auf Seite 51).
- [Ric+12] N. H. Riche, T. Dwyer, B. Lee, and S. Carpendale. “Exploring the Design Space of Interactive Link Curvature in Network Diagrams”. In: *Proc. of the International Working Conference on Advanced Visual Interfaces*. ACM, 2012, pp. 506–513 (Zitiert auf Seite 107).
- [RLS11] A. Radloff, M. Luboschik, and H. Schumann. “Smart views in Smart Environments”. In: *Proc. of the 11th international conference on Smart graphics*. Springer, 2011, pp. 1–12 (Zitiert auf den Seiten 24, 31).
- [RM13] P. Rashidi and A. Mihailidis. “A Survey on Ambient-Assisted Living Tools for Older Adults”. In: *IEEE Journal of Biomedical and Health Informatics* 17.3 (2013), pp. 579–590 (Zitiert auf Seite 23).
- [RNS12] A. Radloff, T. Nocke, and H. Schumann. “Supporting Climate Impact Research by a Smart View Management”. In: *Poster, IEEE InfoVis 2011* (2012) (Zitiert auf Seite 73).
- [Rod+11] E. M. Rodrigues, N. Milic-Frayling, M. Smith, B. Shneiderman, and D. Hansen. “Group-in-a-Box Layout for Multi-faceted Analysis of Communities”. In: *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust (PASSAT) and Social Computing (SocialCom)*. 2011, pp. 354–361 (Zitiert auf Seite 107).
- [RRK04] J. Raisamo, R. Raisamo, and P. Karkkainen. “A method for interactive graph manipulation”. In: *Proc. of the 8th International Conference on Information Visualisation*. 2004, pp. 581–587 (Zitiert auf Seite 107).

- [SA06] B. Shneiderman and A. Aris. “Network Visualization by Semantic Substrates”. In: *IEEE Transactions on Visualization and Computer Graphics* 12.5 (2006), pp. 733–740 (Zitiert auf den Seiten 107, 109, 110).
- [Sei+12] T. Seifried, C. Rendl, M. Haller, and S. Scott. “Regional Undo/Redo Techniques for Large Interactive Surfaces”. In: *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2012, pp. 2855–2864 (Zitiert auf den Seiten 51, 97).
- [SF08] A. S. Spritzer and C. M. D. S. Freitas. “A Physics-based Approach for Interactive Manipulation of Graph Visualizations”. In: *Proc. of the Working Conference on Advanced Visual Interfaces*. ACM, 2008, pp. 271–278 (Zitiert auf Seite 107).
- [SH14] A. Satyanarayan and J. Heer. “Authoring Narrative Visualizations with Ellipsis”. In: *Computer Graphics Forum* 33.3 (2014), pp. 361–370 (Zitiert auf Seite 73).
- [SME15] P. Suta, P. Mongkolnam, and W. Eamsinvattana. “Controlled and Collaborative Presentation via Tablet PCs for Classroom and Meeting Room Uses”. In: *12th International Joint Conference on Computer Science and Software Engineering*. 2015, pp. 102–107 (Zitiert auf Seite 24).
- [SN13] N. A. Suleiman and M. A. Nawkhass. “A New Modified Simplex Method to Solve Quadratic Fractional Programming Problem and Compared it to a Traditional Simplex Method by Using Pseudoaffinity of Quadratic Fractional Functions”. In: *Applied Mathematical Sciences* 7, no. 76 (2013), pp. 3749–3764 (Zitiert auf Seite 69).
- [Spi+12] R. Spicer, Y.-R. Lin, A. Kelliher, and H. Sundaram. “NextSlidePlease: Authoring and delivering agile multimedia presentations”. In: *ACM Trans. Multimedia Comput. Commun. Appl.* 8.4 (2012), 53:1–53:20 (Zitiert auf den Seiten 35, 46).
- [Ste+11] M. Steinberger, M. Waldner, K. Kashofer, M. Streit, and D. Schmalstieg. “Context-Preserving Visual Links”. In: *IEEE Transactions on Visualization and Computer Graphics* 17.12 (2011), pp. 2249–2258 (Zitiert auf Seite 107).
- [Sun00] C. Sun. “Undo Any Operation at Any Time in Group Editors”. In: *Proc. of the 2000 ACM Conference on Computer Supported Cooperative Work*. ACM, 2000, pp. 191–200 (Zitiert auf Seite 51).
- [TAS09] C. Tominski, J. Abello, and H. Schumann. “CGV - An Interactive Graph Visualization System”. In: *Computers & Graphics* 33.6 (2009), pp. 660–678 (Zitiert auf Seite 107).

- [TC05] J. J. Thomas and K. A. Cook, eds. *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. National Visualization and Analytics Ctr, 2005 (Zitiert auf den Seiten 29, 81).
- [Thi10] C. Thiede. “Visuelle Informationsdarstellung in Smart Environments”. PhD thesis. Fakultät für Informatik und Elektrotechnik, Universität Rostock, 2010 (Zitiert auf den Seiten 24, 29, 30, 33).
- [TMC04] D. S. Tan, B. Meyers, and M. Czerwinski. “WinCuts: Manipulating Arbitrary Window Regions for More Effective Use of Screen Space”. In: *Extended Abstracts on Human Factors in Computing Systems*. ACM, 2004, pp. 1525–1528 (Zitiert auf den Seiten 34, 35, 73).
- [Ubi15] Ubisense. *Ubisense Series 7000 IP Sensors Precision ultra - wideband measurement devices for industrial environments - Manual / Ubisense*. 2015. URL: <https://ubisense.net/> (Zitiert auf Seite 27).
- [VMP11] D. A. Vega-Oliveros, D. S. Martins, and M. d. G. C. Pimentel. “Media-oriented Operators for Authoring Interactive Multimedia Documents Generated from Capture Sessions”. In: *Proc. of the 2011 ACM Symposium on Applied Computing*. ACM, 2011, pp. 1267–1272 (Zitiert auf Seite 50).
- [Wai+03] A. Waibel, T. Schultz, M. Bett, M. Denecke, R. Malkin, I. Rogina, R. Stiefelhagen, and J. Yang. “SMaRT: The Smart Meeting Room Task at ISL”. In: *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 4. 2003, pp. 752–755 (Zitiert auf Seite 23).
- [Wal+09] M. Waldner, A. Lex, M. Streit, and D. Schmalstieg. “Design Considerations for Collaborative Information Workspaces in Multi-Display Environments”. In: *Proc. of Workshop on Collaborative Visualization on Interactive Surfaces* (2009) (Zitiert auf Seite 24).
- [Wal+10] M. Waldner, W. Puff, A. Lex, M. Streit, and D. Schmalstieg. “Visual Links Across Applications”. In: *Proc. of Graphics Interface*. Canadian Information Processing Society, 2010, pp. 129–136 (Zitiert auf Seite 107).
- [Wal+11a] M. Waldner, R. Grasset, M. Steinberger, and D. Schmalstieg. “Display-Adaptive Window Management for Irregular Surfaces”. In: *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*. ACM, 2011, pp. 222–231 (Zitiert auf Seite 34).
- [Wal+11b] M. Waldner, M. Steinberger, R. Grasset, and D. Schmalstieg. “Importance-Driven Compositing Window Management”. In: *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2011, pp. 959–968 (Zitiert auf Seite 34).

- [Wal11] M. Waldner. “WIMP Interfaces for Emerging Display Environments”. PhD thesis. Graz University of Technology Institute for Computer Graphics and Vision, 2011 (Zitiert auf Seite 35).
- [Wei99] M. Weiser. “The Computer for the 21st Century”. In: *SIGMOBILE Mob. Comput. Commun. Rev.* 3.3 (1999), pp. 3–11 (Zitiert auf Seite 23).
- [Wig+09] D. Wigdor, H. Jiang, C. Forlines, M. Borkin, and C. Shen. “WeSpace: The Design Development and Deployment of a Walk-up and Share Multi-surface Visual Collaboration System”. In: *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2009, pp. 1237–1246 (Zitiert auf den Seiten 24, 34, 35).
- [WL07] G. Wallace and K. Li. “Virtually Shared Displays and User Input Devices”. In: *Proc. of the USENIX Annual Technical Conference*. USENIX Association, 2007, 31:1–31:6 (Zitiert auf Seite 34).
- [Wol59] P. Wolfe. “The Simplex Method for Quadratic Programming”. In: *Econometrica* 27.3 (1959), pp. 382–398 (Zitiert auf Seite 69).
- [WS11] Y. Wang and K. Sumiya. “Slide KWIC: Snippet Generation for Browsing Slides Based on Conceptual Relationship and Presentational Structure”. In: *9th International Conference on Creating, Connecting and Collaborating through Computing*. 2011, pp. 40–47 (Zitiert auf Seite 46).
- [YGW04] J. Yang, N. Gu, and X. Wu. “A Document Mark Based on Method Supporting Group Undo”. In: *ACM Workshop on Collaborative Editing Systems*. 2004 (Zitiert auf Seite 51).
- [YM15] Y. S. Yoon and B. A. Myers. “Supporting Selective Undo in a Code Editor”. In: *Proc. of the 37th International Conference on Software Engineering*. Vol. 1. IEEE, 2015, pp. 223–233 (Zitiert auf den Seiten 51, 97).
- [YN10] Z. Yu and Y. Nakamura. “Smart Meeting Systems: A Survey of State-of-the-Art and Open Issues”. In: *ACM Computing Surveys* 42.2 (2010), p. 8 (Zitiert auf den Seiten 13–15, 24, 25).
- [You05] G. M. Youngblood. “Smart Environments: Technologies, Protocols, and Applications”. In: *Smart Environments: Technologies, Protocols, and Applications*. John Wiley & Sons, Inc., 2005. Chap. 5 Middleware, pp. 101–127 (Zitiert auf Seite 25).
- [YW14] J. Yang and D. Wigdor. “Panelrama: Enabling Easy Specification of Cross-device Web Applications”. In: *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2014, pp. 2783–2792 (Zitiert auf Seite 34).

Literaturverzeichnis

- [YZN13] Z. Yu, X. Zhou, and Y. Nakamura. “Extracting Social Semantics from Multimodal Meeting Content”. In: *Pervasive Computing* 12.2 (2013), pp. 68–75 (Zitiert auf Seite 126).
- [ZBO13] X. Zhao, Y. Brun, and L. J. Osterweil. “Supporting Process Undo and Redo in Software Engineering Decision Making”. In: *Proc. of the 2013 International Conference on Software and System Process*. ACM, 2013, pp. 56–60 (Zitiert auf Seite 97).
- [Zha+04] H. Zhang, Q. Liu, S. Lertsithichai, C. Liao, and D. Kimber. “A Presentation Authoring Tool for Media Devices Distributed Environments”. In: *International Conference on Multimedia and Expo*. Vol. 3. IEEE, 2004, pp. 1755–1758 (Zitiert auf Seite 35).
- [ZI97] C. Zhou and A. Imamiya. “Object-based Nonlinear Undo Model”. In: *Proc. of the 21st Annual International Computer Software and Applications Conference*. 1997, pp. 50–55 (Zitiert auf Seite 51).

Publikationen und Vorträge

Journal-Publikationen

- [Eic+14] Christian Eichner, Arne Bittig, Heidrun Schumann and Christian Tomin-ski. Analyzing Simulations of Biochemical Systems with Feature-Based Visual Analytics. *Computers & Graphics*, 38(0) : 18 - 26, 2014. DOI: 10.1016/j.cag.2013.09.001.
- [Eic+15] Christian Eichner, Thomas Nocke, Hans-Jörg Schulz and Heidrun Schu-mann. Interactive Presentation of Geo-Spatial Climate Data in Multi-Display Environments. *ISPRS International Journal of Geo-Information*, 4(2) : 493, 2015. DOI: 10.3390/ijgi4020493.
- [Eic+16] Christian Eichner, Stefan Gladisch, Heidrun Schumann and Christian To-minski. Direct Visual Editing of Node Attributes in Graphs. *Informatics*, 3(4) : 17, 2016. DOI: 10.3390/informatics3040017.

Konferenzbeiträge

- [Eic+13] Christian Eichner, Arne Bittig, Heidrun Schumann and Christian To-minski. Feature-Based Visual Analytics for Studying Simulations of Dy-namic Bi-Stable Spatial Systems. In *Proceedings of the EuroVis Work-shop on Visual Analytics (EuroVA)*. Eurographics Association, 2013. DOI: 10.2312/PE.EuroVAST.EuroVA13.025-029.
- [ENS15] Christian Eichner, Martin Nyolt and Heidrun Schumann. A Novel Infra-structure for Supporting Display Ecologies. In *Proceedings of the 11th International Symposium on Visual Computing (ISVC)*, Las Vegas, NV, USA, December 2015. DOI: 10.1007/978-3-319-27863-6_68.

Eidesstattliche Erklärung

Hiermit versichere ich an Eides statt, dass ich die eingereichte Dissertation selbstständig und ohne fremde Hilfe verfasst, andere als die von mir angegebenen Quellen und Hilfsmittel nicht benutzt und die den benutzten Werken wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Diese Arbeit hat noch keiner anderen Fakultät oder Universität zur Prüfung vorgelegen.

.....
Christian Eichner

Rostock, 11.12.2016

Thesen

1. Die Anzeige von visuellen Inhalten ist ein wichtiges Hilfsmittel, um die Informationspräsentation zu unterstützen. Die Nutzung von Multi-Display-Umgebungen (MDE) ist hierbei von besonderer Bedeutung, weil diese es ermöglichen, mehrere Inhalte aus verschiedenen Quellen zu kombinieren und für mehrere Nutzer gleichzeitig sichtbar anzuzeigen.
2. Die Verwendung von Multi-Display-Umgebungen für die Informationspräsentation stellt jedoch besondere Anforderungen. Mehrere Inhalte von verschiedenen persönlichen Geräten müssen gesammelt, für die Anzeige durch die Nutzer ausgewählt, als Views auf die Displays verteilt, während der Präsentation bzw. Diskussion ausgetauscht und ggf. nachträglich durch die Nutzer angepasst werden. Für diese Aufgaben ist Assistenz erforderlich, welche die Nutzer hierbei unterstützt.
3. Smart Meeting Rooms (SMRs) bieten als eine besondere Art von Multi-Display-Umgebungen prinzipiell neue Assistenzmöglichkeiten, um die Nutzer vielfältig zu unterstützen. Bisherige Assistenz-Systeme für SMRs adressieren aber kaum die Informationsanzeige für Präsentationen und Diskussionen. Bestehende Präsentationssysteme für MDE sind andererseits nur begrenzt dazu in der Lage, ggf. vorhandene Assistenz aus den SMRs zu integrieren.
4. Neue Modellbeschreibungen sind erforderlich, um die Informationspräsentation auf mehreren Displays in SMRs durch Assistenz zu unterstützen.

In dieser Dissertation werden mit dem Präsentationsgraph, dem Semantischen Graph und dem Interaktionsgraph hierzu drei unterschiedliche Aspekte bei der Modellierung adressiert. Der Präsentationsgraph beschreibt Nutzervorgaben für die räumliche Verteilung und für die zeitliche Ersetzung mehrerer Views auf unterschiedlichen Displays. Der Semantische Graph stellt einen thematischen Zusammenhang zwischen verschiedenen Views her, um eine spätere Suche nach passenden Inhalten zu unterstützen und der Interaktionsgraph modelliert durch die Nutzer vorgenommene Anpassungen an der Anzeige.

5. Für das Anpassen der Informationsanzeige durch mehrere Nutzer muss jeder Nutzer einen individuellen Zugang zu den Modellen erhalten.

In dieser Dissertation wurde hierzu ein Nutzer-Interface entwickelt, das auf den persönlichen Geräten der Nutzer aufgerufen werden kann. Über das Interface kann

jeder Nutzer eigene Inhalte für die Anzeige bereitstellen, Inhalte aus verschiedenen Quellen durchsuchen und für die Anzeige auf den Displays eines SMRs auswählen.

6. Das Verteilen und Anordnen einer sich laufend verändernden Anzahl an Views auf mehreren im Raum verteilten Displays erfordert Assistenz.

In dieser Dissertation wurde ein Layout-Algorithmus entwickelt, der diese Aufgabe automatisch übernimmt. Hierbei werden einerseits Nutzervorgaben für die Verteilung und Anordnung berücksichtigt und andererseits Eigenschaften des Raums wie z.B. Display- und Nutzerpositionen beachtet, um eine möglichst gute View-Verteilung zu erreichen.

7. Um während einer Präsentation auch Explorations- bzw. Analyseaufgaben übernehmen zu können, ist es erforderlich, dass je nach Bedarf auch neue Darstellungen in die Anzeige auf den Displays aufgenommen oder bereits verwendete Visualisierungen nachträglich angepasst werden können.

In dieser Dissertation wird hierzu ein Ansatz zur interaktiven Generierung und Anpassung von Views durch externe Tools vorgestellt. Die neuen Views werden automatisch in eine existierende Anzeige integriert und erlauben es den Nutzern auf diese Weise, spontan während einer Analyse alternative Hypothesen anhand von neu generierten Darstellungen zu testen oder bei Bedarf neue Visualisierungen (z.B. aus dem Internet) nachzuladen.

8. Während einer Diskussion ist es erforderlich, dass mehrere Nutzer häufig die Informationsanzeige anpassen. Hierzu müssen sie Veränderungen an der Anzeige aus früheren Diskussionsverläufen nachzuvollziehen und mit geringem manuellen Aufwand zwischen verschiedenen Argumentationslinien wechseln können.

In dieser Dissertation werden diese Aufgaben durch eine entwickelte Visualisierung der automatisch aufgezeichneten Interaktionsfolgen unterstützt. Basierend auf dieser Visualisierung wird ein Undo/Redo-Mechanismus eingeführt, der es erlaubt, verschiedene Zuständen der Anzeige aufzurufen und einzelne Änderungen mit geringem Interaktionsaufwand rückgängig zu machen oder wiederherzustellen. Diese Funktionen können hierbei auch selektiv nur für ein einzelnes Display, einen einzelnen Nutzer oder für eine bestimmte View genutzt werden.

9. Diskussionen und Präsentationen mit mehreren beteiligten Nutzern erfordern es oft, dass semantisch passende Inhalte gefunden und auf den Displays aufgerufen werden. Die große Anzahl an verschiedenen Inhalten aus unterschiedlichen Quellen erschwert hierbei die Suche.

In dieser Dissertation wird die Suche nach semantisch passenden Inhalten durch eine Visualisierung des Semantischen Graphen unterstützt. Weiterhin können Navigationshinweise und View-Empfehlungen im Nutzer-Interface auf den persönlichen Geräten und auf den Displays des SMRs bei Bedarf eingeblendet werden.

10. Wechselnde Situationen oder einzelne Szenarien während einer Präsentation bzw. Diskussion können es erforderlich machen, die entwickelten Ansätze weiter auf die aktuelle Situation abzustimmen.

In dieser Dissertation werden hierzu Situationen identifiziert, die durch die Situationserkennung in einem SMR erkannt und durch die entwickelten Ansätze auf unterschiedliche Weise unterstützt werden können. Hierzu werden automatische Interface-Anpassungen, automatische Layout-Anpassungen und das automatische Einblenden von Navigationshinweisen eingesetzt.

11. Die entwickelten Modelle und darauf aufbauenden Konzepte bieten neue Möglichkeiten, um die Informationspräsentation in SMRs zu beschreiben und durch eine assistierte Präsentation und Diskussion zu unterstützen. Mit den situationsabhängigen Adaptionenmechanismen wird zudem eine Assistenz erreicht, die den bisherigen Stand der Forschung im Bereich der SMRs erweitert.