



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Master Arbeit

Benjamin Burchard

**Evaluation und Analyse von Visualisierungen und
Visualisierungswerkzeugen für Ethernet
Netzwerkkommunikationsdaten im Fahrzeug**

*Fakultät Technik und Informatik
Studiendepartment Informatik*

*Faculty of Engineering and Computer Science
Department of Computer Science*

Benjamin Burchard

**Evaluation und Analyse von Visualisierungen und
Visualisierungswerkzeugen für Ethernet
Netzwerkkommunikationsdaten im Fahrzeug**

Master Arbeit eingereicht im Rahmen der Master Ausarbeitung

im Studiengang Master of Science Informatik
am Department Informatik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr. Franz Korf

Eingereicht am: 19. März 2017

Benjamin Burchard

Thema der Arbeit

Evaluation und Analyse von Visualisierungen und Visualisierungswerkzeugen für Ethernet
Netzwerkkommunikationsdaten im Fahrzeug

Stichworte

Schlüsselwort 1, Schlüsselwort 2

Kurzzusammenfassung

Dieses Dokument ...

Benjamin Burchard

Title of the paper

Evaluation and analysis of visualisation tools and technics for ethernet network communica-
tiondata in vehicles

Keywords

keyword 1, keyword 2

Abstract

This document ...

Inhaltsverzeichnis

1	Einleitung	1
2	Einführung in das Umfeld	4
2.1	Grundlagen der Evaluation	5
2.1.1	Evaluationsziele	7
2.1.2	Evaluationstechniken basierend auf Expertenanalyse	8
2.1.3	Evaluationstechniken mit Nutzerbeteiligung	15
2.2	Einführung in grundlegende Visualisierungen und Werkzeuge	15
2.3	Grundlagen für Anforderungen und Analyse	17
3	Evaluationsbasis	19
3.1	Häufig verwendete Evaluationen im Bereich Visualisierung	19
3.2	Faktoren zur Abgrenzung der Evaluationstechniken	20
3.3	Entscheidung für Evaluation	20
4	Literaturauswertung	21
4.1	Verwandte Arbeiten	21
4.2	Visualisierungen und Werkzeuge	21
5	Anforderungsanalyse	22
5.1	Allgemeine Anforderungen	22
5.1.1	Anforderungen an Visualisierungen	22
5.1.2	Anforderungen an Werkzeuge	22
5.2	Automotive spezifische Anforderungen	22
6	Anwendungskonzepte und -bereiche	24
7	Zusammenfassung und Fazit	25
	Glossar	29

Abbildungsverzeichnis

1.1	Grober Aufbau der Arbeit	2
2.1	An diese Arbeit angrenzende wissenschaftliche und wirtschaftliche Bereiche .	4

1 Einleitung

Die Visualisierung von Daten zielt darauf ab, Daten durch eine grafische Interpretation verwertbar zu machen. Durch die Aufbereitung von Daten entsteht ein Mehrwert für den Nutzer, in unserem Fall für den Anwender der Visualisierung oder des Werkzeugs zur Darstellung von Netzwerkkommunikationsdaten im Fahrzeug-Ethernet. Visualisierungen ermöglichen es, auf mehr als nur einem Weg, rohen Daten einen Sinn zu verleihen. Das Abbilden der Datenattribute auf visuelle Eigenschaften wie Position, Form, Farbe und Größe wird eingesetzt um Nutzern die Wahrnehmung und Interpretation von Mustern innerhalb der Daten zu erleichtern [25]. Damit visuelle Analyse Tools effektiv genutzt werden können, müssen diese eine fließende und flexible Nutzung von Visualisierungen ermöglichen.

In zukünftigen automobilen Netzwerken wird vermehrt Ethernet zum Einsatz kommen. Nicht nur die zunehmende Bandbreitenanforderung, durch mehr bandbreitenintensive Geräte im Fahrzeug (Kameras, Laserscanner etc.), verlangt nach einem neuen Übertragungsmedium. Auch die bestehende Diversität der verbauten Bustechnologien und die damit einhergehende Vielschichtigkeit des Fahrzeugnetzes, spricht für eine einheitliche und weniger komplexe Lösung wie sie mit Echtzeit Ethernet gegeben wäre. Diese Umstrukturierung bringt neue Metriken für die Daten des Fahrzeugnetzes mit sich. Um diese für das automobile Echtzeit-Ethernet zu analysieren, benötigt man eine hinreichende visuelle Aufbereitung dieser Kommunikationsdaten und deren Eigenschaften, wie Jitter oder Latenz. Kommunikationsdaten sind im Grunde alle Daten welche über das Netzwerk transportiert, also kommuniziert werden. Um Kommunikationsdaten, sowie die Eigenschaften dieser, zu visualisieren gibt es viele verschiedene Möglichkeiten und Techniken. Informationsvisualisierung versucht mit Hilfe von geeigneten grafischen Repräsentationen Nutzerinformationen zu optimieren. Diese bildlichen Darstellungsformen sollen bei der Interpretation und Auswertung von großen Datenmengen helfen. In dem im Projekt, in dessen Rahmen diese Arbeit geschrieben wird, vorhandenem Automobilern Fahrzeugnetz entstehen bis zu 3 Gigabyte Daten pro Stunde [26]. Viele moderne Fahrzeuge kommunizieren bereits mit mehr als 100 ECUs (Electronic Control Units) innerhalb ihrer Netze [5]. Anhand dieser Zahlen lässt sich sagen, dass einen Bedarf sowie Nutzen für Visualisierungen und Werkzeuge besteht, welche die Daten abstrahieren und dem Nutzer verständlich machen.

Um geeignete Visualisierungen und Werkzeuge für das Themenfeld der Netzwerkcommunication im Fahrzeugethernet zusammenzutragen, müssen vorhandene Visualisierungen hinsichtlich ihrer Verwendbarkeit im genannten Themenfeld eingeschätzt werden. Dieser Teil der Arbeit soll mit Hilfe einer Evaluation verwirklicht werden. Es gibt eine breite Auswahl an Evaluationstechniken, welche alle zum Ziel haben eine Software oder ein Softwareprodukt auf verschiedenste Eignungen hin zu untersuchen. Hier muss eine differenzierte Auswahl erfolgen welche Technik oder Techniken für diesen Themenkomplex, aber für den spezifischen Fall der Evaluation in einer Masterarbeit geeignet sind. Der grobe Aufbau und Fluss dieser Arbeit wird in Abbildung 1.1 dargestellt.



Abbildung 1.1: Grober Aufbau der Arbeit

Diese Arbeit hat zum Ziel Evaluationstechniken in Hinblick auf deren Einsatzfähigkeit im Feld der Visualisierung von Fahrzeugkommunikationsdaten und deren Werkzeuge zu Analysieren. Unter Zuhilfenahme der erarbeiteten Evaluationstechniken wird spezifische Literatur ausgewertet. Damit ist gemeint das hier Literatur speziell aus dem Bereich der Netzwerkvisualisierung ausgewertet wird, mit Hauptaugenmerk auf die Darstellung von Kommunikationsdaten im Fahrzeug. Aus der Analyse der Literatur ergeben sich Anforderungen an Visualisierungen für den Automotive Bereich, welche Aufgearbeitet werden und somit eine Grundlage für die Entwicklung neuer Visualisierungssystemen und Werkzeuge bilden. Somit ergeben sich mehrere konkrete Teilziele dieser Arbeit. Es werden geeignete Evaluationstechniken für die Analyse von Werkzeugen und Visualisierungen im Bereich der Visualisierung von Fahrzeugkommunikation erarbeitet. Es werden konkrete Anforderungen für zukünftige und aktuelle Darstellungsformen und die zugehörige Software zusammengestellt.

In Kapitel 2 wird grob das wissenschaftliche und wirtschaftliche Umfeld umrissen. In drei Unterabschnitten werden hier die Grundlagen und angrenzenden Wissensbereiche dieser Arbeit beleuchtet. Es wird dargelegt inwiefern, nach einschlägiger Literatur, die Wahl der richtigen Evaluationsmethode die Qualität der Ergebnisse und der möglichen Schlussfolgerungen beeinflusst. Aktuelle und bewährte Konzepte der eingesetzten Ethernet Netzwerke, sowie deren Methoden zur Visualisierung von Kommunikationsdaten in Bereichen der Flugzeugtechnik, der Automatisierungstechnik sowie der werden erörtert, als auch die Anforderungen und die Analyse dieser betrachtet. Kapitel 3 vergleicht verschiedene Evaluationstechniken und

erläutert den Aufbau und die Herangehensweise dieser. Im Verlauf des Kapitels wird Aufgrund von verschiedenen Analysen anderer Wissenschaftlicher Ausarbeitungen eine Evaluationsmethode gewählt. Diese wird zur Evaluation für die im Kapitel **Literaturauswertung** vorgestellten akademischen Abhandlungen verwendet. Das oben genannte Kapitel **4** beinhaltet die erwähnte Literaturlauswertung, mit Hilfe der erarbeiteten Evaluationstechnik werden diese Arbeiten aufgearbeitet. Das anschließende Kapitel **5** greift die in Kapitel **4** gewonnenen Erkenntnisse auf und es werden darauf basierend Anforderungen für die Darstellung von Netzwerkdaten im Fahrzeug-Ethernet erstellt. Aufgeteilt wird dieses Kapitel in drei Schritte. Zunächst werden allgemeingültige Anforderungen vorgestellt, gefolgt von Anforderungen speziell für den Visualisierungsbereich. Im Anschluss werden konkrete Anforderungen an Visualisierungen von Netzwerkdaten im Fahrzeug gestellt. Das Kapitel **6** beschreibt welche Anwendungsmöglichkeiten sich aus der Anforderungsanalyse ergeben und wie, als auch wo, diese Kombiniert und eingesetzt werden können. Kapitel **7** schließt diese Arbeit mit einer Zusammenfassung und einem Fazit ab.

2 Einführung in das Umfeld

In der **Einleitung** wurde erläutert warum Visualisierungen für die Darstellung der Netzwerkkommunikationsdaten im Fahrzeug wichtig sind. Dieses Kapitel skizziert die Position dieser Arbeit im Bezug auf das wissenschaftliche Umfeld. Der Kernbereich beschäftigt sich mit Visualisierungen aus dem Automotive Bereich. Im speziellen mit Darstellungen für die Kommunikationsdaten und deren Metriken im modernen Automotive Realtime Ethernet. Neben diesem Hauptbereich gibt es mehrere weitere angrenzende Themen, die für die Arbeit von Bedeutung sind. In der folgenden Übersicht 2.1 sind diese Bereiche dargestellt. Das Umfeld ist im Wesentlichen in drei Teile gegliedert, Evaluation, Werkzeuge und Visualisierungen sowie Anforderungen und die Analyse dieser.



Abbildung 2.1: An diese Arbeit angrenzende wissenschaftliche und wirtschaftliche Bereiche

Im Abschnitt 2.1 wird benannt welche grundlegenden Evaluationen es gibt und stellt diese vor. Es wird die Vorarbeit für das Kapitel **Evaluationsbasis** geleistet, welches die geeignetste Technik für die Visualisierung von Netzwerkdaten im Fahrzeugnetz herausarbeitet. Abschnitt

2.2 beschreibt die Notwendigkeit der Betrachtung von Visualisierungen und Werkzeugen nicht nur aus dem Automotive Bereich sondern auch aus angrenzenden Bereichen. Der Abschnitt befasst sich weiterhin damit welches diese Bereiche sind und warum diese in Frage kommen. Der abschließende Teil dieses Kapitels, **Grundlagen für Anforderungen und Analyse**, geht auf die Anforderungen an Visualisierungen ein. Es wird ...

2.1 Grundlagen der Evaluation

Im Bereich der Evaluation beschäftigt sich die Arbeit mit der Frage welche Techniken zur Evaluation sinnvoll in diesem Themenkomplex einsetzbar sind. Es wird herausgearbeitet welche Evaluationsmöglichkeiten es gibt und welche, aufgrund Ihrer Ergebnisse und Anwendungsgebiete, am geeignetsten erscheinen. Dieser Abschnitt zeigt grundlegende Evaluationsmethoden und deren Anforderungen sowie nutzen. Im Folgekapitel wird dann eine der vorgestellten Methoden ausgewählt und als Grundlage für die Literaturanalyse verwendet. Bevor die Techniken zur Evaluation vorgestellt werden, folgt zunächst ein Einblick wie zwischen Evaluationen differenziert wird, weshalb Evaluation verwendet wird und was damit erreicht werden soll.

In der Literatur wird auf verschiedenste Art und Weise zwischen Evaluationen unterschieden. Im folgenden betrachten wir drei verschiedene Klassifikationsarten sowie deren Merkmale, Vorteile und mögliche Nachteile. Kitchenham [15] hat Mitte der neunziger Jahre bei Evaluierungsmethoden zwischen zwei Messmethodiken differenziert. Zum einen Quantitative Methoden und zum anderen Qualitative Methoden. Eine dritte Variante vereinigt beide Messmethoden, diese werden als Hybride Methoden beschrieben. Die folgende Auflistung beschreibt die genannten Methoden kurz.

- Quantitative Methoden, darunter versteht man die Möglichkeit das sich eine messbare Veränderung einer Eigenschaft des zu testenden Softwareprodukts oder Prozesses einstellt. Dabei werden spezifische wissenschaftliche Werkzeuge und Messungen verwendet.
- Qualitative Methoden, meinen Feature/Eigenschaften Analysen welche sich auf die Anforderungen an eine bestimmte Aufgabe bzw. Ablauf fokussieren und die Umsetzung dieser Anforderungen innerhalb einer Methode oder eines Tools bewerten.
- Hybride Methoden, sind beispielsweise *Qualitative Effect Analysis* oder *Benchmarking*. Die erstgenannte nutzt Quantitative Methoden sowie ein oder mehrere Expertenmeinungen. Beim Benchmarking werden ein oder mehrere Tools einer Reihe von Standardtests unterzogen um die Performance dieser Tools vergleichbar zu machen.

Dieser Ansatz unterscheidet also nach der Art der Messmethode. Der Vorteil in dieser Unterscheidungsart liegt darin, dass bereits zuvor identifiziert werden kann, ob messbare Effekte bei der Evaluation entstehen oder es sich um eine Eignungsanalyse von Methoden oder Werkzeugen zur Bestimmung der Nutzbarkeit handelt. Dabei können jedoch wichtige Aspekte außer Acht gelassen werden, wie zum Beispiel der nötige Zeitaufwand oder zu welchem Zeitpunkt der Entwicklung eines Programms oder eines Designs die Evaluation am sinnvollsten eingesetzt werden kann. In der Psychologie wird bei Evaluationen häufig zwischen formativer und summativer Evaluation unterschieden [TODO: cite]. Es gibt für diese Möglichkeit zur Differenzierung jedoch auch Beispiele aus der Informationsvisualisierung [2].

- Formative Evaluation, die formative Evaluation wird meist begleitend zur Implementierung oder zum Projekt durchgeführt. Es wird dabei in Intervallen untersucht und Zwischenergebnisse festgehalten. So können laufende Prozesse angepasst werden und wiederum erneut evaluiert werden.
- Summative Evaluation, als summative Evaluation wird eine das Resultat analysierende Evaluation bezeichnet. Daraus folgt, diese Evaluationen werden nach Abschluss eines Projektes durchgeführt. So ist es möglich das Projekt, Design oder ähnliches zusammenfassend zu bewerten.

Eine Unterscheidung zwischen Methoden, welche während oder nach einem Projekt durchgeführt werden, erscheint besonders dann sinnvoll, wenn vor einem Projekt festgelegt wird, mit welchen Methoden während und abschließend gearbeitet werden soll. Da in dieser Arbeit bestehende Visualisierungen auf ihre Eignung für den Einsatz in einer möglichen Automotive Ethernet Netzwerkvisualisierung hin evaluiert werden, fallen formative Methoden bereits von vornherein aus. So kann also gesagt werden, dass diese Art der Unterscheidung sich nicht für die Ziele dieser Arbeit eignet.

Die letzte Differenzierung, die hier betrachtet werden soll, unterscheidet grundsätzlich zwischen zwei Herangehensweisen für Evaluationen. Zum einen aus der Sicht des Experten in Hinblick auf Softwaredesigns oder der Usability, ohne Einwirkung von Nutzern. Das sind sogenannte Prüfmethoden. Zum anderen Evaluationen, die die tatsächliche Nutzung eines Systems untersuchen, namentlich Testmethoden, in welchen repräsentative Endnutzer ein oder mehrere Interfaces nutzen und Beobachtungen oder Messungen aufgezeichnet werden.

- Evaluation durch Expertenanalyse, dies basiert auf anerkannten Expertenmeinungen mit Hilfe, welcher das Design bewertet wird und die Auswirkungen auf Nutzer analysiert werden.

- Evaluation durch Nutzermitwirkung, lässt den Endanwender das System auf verschiedene Art und Weisen testen. Die Usability steht im Vordergrund.

Ersteres eignet sich insbesondere zur Bewertung von Design, Darstellung und Prototypen, letzteres benötigt normalerweise einen funktionierenden Prototypen oder eine bereits bestehende Implementierung. Natürlich ist das eine sehr grobe Unterscheidung und es kann dabei zu Überschneidungen kommen. Beispielsweise können Nutzer bereits im frühen Entwicklungsprozess mit eingebunden werden oder Experten basierende Analysen werden an bestehenden Implementierungen durchgeführt um günstig und vor allem schnellere Usability Bewertungen vornehmen zu können. Der grobe Unterscheidungsgrad dieser Aufteilung eignet sich gut um ein weites Spektrum an Evaluationen zu betrachten. Diese Differenzierung erscheint, auf Grund der genannten Vorteile beider Evaluationsarten, am geeignetsten für die Arbeit. Bevor die wesentlichen, in diese Aufteilung passenden, Evaluationen vorgestellt werden, werden zunächst noch die allgemeinen Ziele von Evaluationen betrachtet.

2.1.1 Evaluationsziele

Für Evaluationen können, laut Dix [1], drei Hauptziele genannt werden. Zu nächst den Umfang und die Erreichbarkeit der Funktionalitäten des Systems zu bewerten, die Bewertung der Benutzererlebnisses und spezifische Probleme des Systems zu identifizieren. Die Funktionalität ist wichtig, weil sie den Erwartungen und den Anforderungen des Nutzers entsprechen muss und ihn bei seiner beabsichtigten Aufgabe zu unterstützen. Anders ausgedrückt, sollte das Design den Benutzer dazu befähigen seine Aufgabe leichter zu erledigen. Das bedeutet nicht nur, dass die passende Funktionalität vom System her gegeben ist, sondern auch dass diese leicht erreichbar ist. Dies ist gemeint, im Sinne von Aktionen die ein Nutzer ausführen muss um die gewünschte Aufgabe erledigen zu können. Zusätzlich zur Analyse der Funktionalität gibt es die Analyse der Nutzererfahrung bei der Interaktion mit dem System. Das beinhaltet Aspekte wie die Zufriedenheit des Nutzers mit dem System, wie einfach es zu erlernen ist oder eben die Usability. Weiterführende Punkte können auch den Spaßfaktor und emotionale Reaktionen umfassen, speziell bei Systemen welche zur Unterhaltung oder Entspannung gedacht sind. Das dritte Ziel von Evaluationen, ist Probleme mit dem Design zu identifizieren. Das können Ausprägungen sein wie, unvorhergesehenes Verhalten von bestimmten Funktionen des Designs oder Funktionen welche den Nutzer verwirren. Die beiden erstgenannten Ziele, Funktionalität und Usability des Designs, haben natürlich Einfluss auf dieses Ziel. Trotzdem ist dies ein eigenständiges Ziel, da es sich spezifisch auf die Identifikation von Fehlerquellen und Konflikten bezieht, welche dann korrigiert werden können.

2.1.2 Evaluationstechniken basierend auf Expertenanalyse

Idealerweise beginn die Evaluation bei Software, Design oder Projekten bereits bevor jegliche Implementation stattfindet, also in der Design- und Entwicklungsphase. Wenn bereits das Design selbst evaluiert wird, können schwerwiegende und möglicherweise kostenintensive Fehler vermieden werden. Denn so kann das Design noch angepasst werden bevor zu viele Ressourcen darauf verwendet werden. Typischerweise sind Designfehler umso teurer je später im Entwicklungsprozesses sie entdeckt werden und damit die Wahrscheinlichkeit geringer das diese korrigiert werden können. Allerdings sind regelmäßige Usertests während der Entwicklungsphase teuer und es ist kann sich als schwierig erweisen eine akkurate Bewertung der Interaktionserfahrung der User über ein nicht vollständiges System zu bekommen. Die Konsequenz daraus ist, dass einige Methoden entwickelt wurden um Systeme von Experten evaluieren zu lassen. Diese wiederum können nicht nur verwendet werden um während der Entwicklung bewertend einzugreifen, sondern auch nach der Veröffentlichung um eine Qualitäts- oder Eignungsbewertung vorzunehmen. Im Grunde können diese Methoden zu jedem möglichem Zeitpunkt im Entwicklungsprozess eingesetzt werden, von der Spezifikation des Designs über Prototypen sowie eben auch vollständigen Implementationen, was sie zu einem sehr flexiblem Ansatz zur Evaluation macht. Der Aspekt, der Flexibilität und der Möglichkeit vollwertige Implementationen zu untersuchen, ist besonders relevant für diese Arbeit. Es unterstützt den Ansatz der Arbeit bestehende Visualisierungen auf ihre Eignung zur Darstellung von Netzwerkkommunikationsdaten im Automotive Ethernet Netzwerk zu präsentieren. Die Möglichkeit der Evaluation während der Entwicklung hingegen nicht weiter betrachtet, da diese nicht relevant ist. Expertenanalysen hängen davon ab, dass Designer, Analysten oder Domänenexperten, Design und Darstellungen betrachten und bewerten wie sich dieses auf den typischen User auswirkt. Die grundlegende Intension ist Teile der Software oder des Designs beziehungsweise der Visualisierungen zu identifizieren welche sich nicht an Kognitive Prinzipien halten oder bestätigte empirische Untersuchungen ignorieren. Was diese Methoden nicht betrachten ist die tatsächliche Nutzung des Systems, es kann nur analysiert werden ob Usability Standards eingehalten werden. Nachfolgend werden aus dem Bereich der Expertenanalyse folgende Ansätze vorgestellt:

- Cognitive Walkthrough
- Heuristische Evaluation
- Modellbasierte Evaluation
- Verwendung von vorangegangenen Studien

Heuristische Evaluation

Vorgestellt wurde die heuristische Evaluation von Nielsen und Molich [20] 1990. Es handelt sich dabei um eine Usability Untersuchung in welcher jedes Element eines User Interfaces auf bestimmte Usability-Prinzipien hin untersucht wird. Bevor diese Prinzipien entwickelt wurden, wurde die Heuristische Evaluation, im Allgemeinen, basierend auf Intuition und allgemein anerkannten Methoden aufgebaut. Da Heuristische Evaluation bereits auf Design Spezifikationen angewendet werden kann eignet es sich bereits zur Auswertung von frühen Designphasen. Allerdings kann diese Art der Evaluation auch auf Prototypen, Konzepte und bereits voll funktionsfähige Systeme angewendet werden. Daher ist Heuristische Evaluation ein flexibler und günstiger Ansatz. Aufgrund dieser Attribute wird es auch häufig als Discount Usability Methode bezeichnet [13][19]. Generell gibt es bei Heuristischer Evaluation die Vorgabe dass mehrere Auswerter ein System, unabhängig von einander, beurteilen und mögliche Usabilityprobleme aufzeigen. Es ist entscheidend dass es mehrere Auswerter gibt und auch dass diese unabhängig von einander das System evaluieren. Nielsen fand in seiner Abhandlung heraus dass drei bis fünf Auswerter ausreichend sind. Bei einer Anzahl von fünf Auswertern, werden im Schnitt 75% aller Usabilityprobleme gefunden. Wohingegen einzelne Auswerter nur zwischen 20 und 51% finden konnten, bei mehr als zehn Auswertern ist keine Steigerung der gefundenen Usabilityprobleme mehr zu erkennen. Um die Auswerter bei der Evaluation zu unterstützen gibt es zehn vorgegebene Heuristiken [19]. Diese können, wenn nötig, um Domänenspezifische Heuristiken ergänzt werden. Für die Darstellung von Netzwerkdaten im Automotive Ethernet Bereich, wären beispielsweise *Erkennbarkeit einschlägiger Metriken* eine mögliche Ergänzung. Jeder Auswerter bewertet das System und notiert jeden Verstoß gegen eine der vorgegebenen Heuristiken, welcher ein potentiell Usabilityproblem aufzeigt. Zu jedem Fund wird die Schwere des Problems oder des Verstoßes festgehalten. Diese Einschätzung basiert auf vier Faktoren.

1. Wie häufig tritt das Problem auf
2. Wie einfach ist es für den User dieses zu überwinden
3. Tritt es einmalig auf oder ist es reproduzierbar
4. Als wie schwer wird das Problem wahrgenommen

Diese werden auf einer Skala von 0 - 4 bewertet.

0 = Es ist kein Usabilityproblem

1 = Kosmetisches Problem, nicht zwingend zu bearbeiten

2 = kleines Usabilityproblem, niedrige Priorität

3 = großes Usabilityproblem, hohe Priorität

4 = kritisches Usabilityproblem, muss vor Veröffentlichung gelöst werden

Die zehn oben angesprochenen Heuristiken sind:

1. **Simple und natürliche Dialoge:** Dialoge sollten keine Informationen enthalten, welche irrelevant sind oder kaum verwendet werden. Jedes zusätzliche Informationselement in einem Dialog, konkurriert mit den relevanten Informationselementen und mindert deren relative Sichtbarkeit. Alle Informationen sollten in einer natürlichen und logischen Reihenfolge auftreten.
2. **Dem User verständliche Sprache nutzen:** Dialoge sollten klar, in Worten, Sätzen und Konzepten die dem User bekannt sind, ausgedrückt werden, jedoch nicht in eher systemnahen Ausdrücken.
3. **Minimierung der Gedächtnisbelastung:** Der Nutzer sollte sich keine Informationen Dialogübergreifend merken müssen. Anleitungen zur Nutzung des Systems, sollten jederzeit sichtbar oder einfach aufrufbar gemacht werden, dort wo es angebracht ist.
4. **Konsistenz:** Nutzer sollten sich nicht fragen müssen ob mit einem unterschiedlichen Ausdrücken, Situationen oder Aktionen das Gleiche gemeint ist.
5. **Feedback:** Das System sollte dem Nutzer zu jeder Zeit, ausreichend und innerhalb einer angemessenen Zeitspanne, Feedback geben um den Status des Programms einschätzen zu können.
6. **Klar markierte Ausgänge:** Es kommt häufig vor das Nutzer Systemfunktionen ohne Intension auswählen. In diesen Situationen muss es ein klar markierter *Ausgang*, um den ungewünschten Status verlassen zu können, vorhanden sein.
7. **Abkürzungen:** Diese stellen Beschleunigungen für den erfahrenen Benutzer dar. Beispielsweise können Tastenkombinationen für Aktionen bereitgestellt werden, welche den Arbeitsablauf beschleunigen. Diese Funktionen für erfahrenere Nutzer, sollten nicht auf den ersten Blick sichtbar sein, um einen Anfänger nicht zu verwirren.

8. **Gute Fehlermeldungen:** Fehlermeldungen sollten klar in Sprache ausgedrückt werden, nicht in Codes. Es soll das Problem präzise beschrieben werden und konstruktive, im besten Fall, dass Problem lösende, Informationen enthalten.
9. **Fehler vermeiden:** Besser als gute Fehlermeldungen ist, wenn Fehler gar nicht erst auftreten. Ein durchdachtes Systemdesign kann eine Vielzahl an einfachen Fehlern abfangen.
10. **Hilfe und Dokumentation:** Im Idealfall benötigt ein Benutzer keine Hilfestellung für die Nutzung eines Systems. Da das nicht auf jede Person und jedes System zutreffen kann sollte es eine Hilfefunktion sowie eine Dokumentation geben. Die dort enthaltenen Informationen sollten einfach erreichbar, leicht zu durchsuchen, mit der aktuellen Nutzertätigkeit verbunden, konkrete Schritte zur Lösung bereithalten und nicht zu Überladen, also kompakt, sein.

Nach dem Abschluss der separaten Bewertung durch die Experten, werden alle erkannten Probleme gesammelt und die durchschnittliche Bewertung errechnet. Daraufhin wird festgelegt welches das wichtigste Problem ist und somit als erstes der Aufmerksamkeit im Sinne der Problemlösung bedarf. Zusammenfassend beinhaltet die Heuristische Evaluation als eine kleine Gruppe von Auswertern welche das Interface, Design oder Darstellungen untersuchen und diese auf ihre Tauglichkeit mit Blick auf die Usabilityprinzipien (die Heuristiken) bewerten.

Cognitive Walkthrough

Cognitive Walkthrough (CW) ist eine Methode, ursprünglich vorgestellt von Lewis et. al. [18], um systematisch Features eines Interfaces im theoretischem Kontext zu bewerten. Der Ursprung des Cognitive Walkthrough Ansatzes ist der Code Walkthrough, bekannt aus dem Software Engineering Bereich. In Walkthroughs werden detailliert Sequenzen von Aktionen betrachtet. Im Code Walkthrough ist eine Sequenz repräsentiert durch einen Teil des Programmcodes, durch welchen dann schrittweise iteriert wird. Dabei wird auf bestimmte Charakteristiken geachtet, wie beispielsweise Coding Style oder Code Conventions. Beim Cognitive Walkthrough ist die Aktionssequenz beschrieben als, die Schritte die ein Nutzer vollführen muss um im vorgegebenen Interface eine bestimmte Aufgabe zu erledigen. Die Rezensenten gehen dann die Aktionssequenz Schritt für Schritt durch und achten auf mögliche Usabilityprobleme. Hauptsächlich fokussiert sich die CW Methode darauf, festzustellen wie einfach ein System zu erlernen ist. Dabei liegt ein spezieller Fokus darauf wie leicht es ist, dass System explorativ zu erlernen. Diese Methode wird von der Mehrheit von Nutzern vorgezogen, es ist einfacher das System zu testen und dabei zu lernen, als zunächst eine Anleitung

oder spezielles Training zu absolvieren [7]. Für jeden Schritt wird festgehalten warum dieser Schritt gut oder nicht gut für einen neuen Nutzer ist. Diese Rückmeldungen sind sogenannte “success stories“ beziehungsweise “failure stories“. Um einen solchen Cognitive Walkthrough durchzuführen sind vier Anforderungen zu erfüllen.

1. Eine Spezifikation oder einen Prototypen des Systems. Alle möglichen Angaben dazu sollten sehr detailliert sein. Details beim Wortlaut oder der Position eines Menüs oder Steuerelements können große Unterschiede machen.
2. Eine Beschreibung der Aufgabe die ein Nutzer in dem System ausführen soll. Diese Aufgabe sollte repräsentativ gewählt werden, also etwas das jeder Nutzer tun muss.
3. Eine komplette Auflistung aller Aktionen die ausgeführt werden müssen um die Aufgabe im vorgegebenen System zu erfüllen.
4. Eine Angabe über die Zielgruppe. Wer wird das System nutzen und wie viel Erfahrung und Vorwissen kann von diesen Nutzern erwartet werden.

Mit diesen Informationen kann der Experte, die in Punkt 3 beschriebene Aktionssequenz durchlaufen und Anhand dieser die Usability dieses Teils bewerten. Um zu eine Bewertung zu gelangen, werden vier Fragen für jede Aktion der Sequenz beantwortet.

1. Stimmt der Effekt der Aktion mit den Zielen des Nutzers zum gegebenen Zeitpunkt überein? Jede Aktion eines Nutzers hat einen bestimmten Effekt innerhalb des Systems. Trägt dieser Effekt tatsächlich dazu bei das der Nutzer dem Ziel näher kommt? Beispielsweise wenn der Effekt der Aktion ist ein PDF einer Darstellung zu speichern, ist dass auch tatsächlich was der Nutzer in diesem Moment tun möchte?
2. Ist es dem Nutzer möglich, zu sehen das die Aktion verfügbar ist? Können Nutzer den Button oder das Menüelement sehen welches die gewünschte Aktion ausführt? Diese Frage soll nicht beantworten ob der User erkennt das die Funktion diejenige ist die er benötigt, sondern ob diese Sichtbar ist zu dem Zeitpunkt wenn die Aktion ausgeführt werden soll. Menüelemente oder Buttons könnten sich zum Beispiel in Untermenüs befinden.
3. Wenn der Nutzer die korrekte Aktion gefunden hat, wird er in der Lage sein zu erkennen das dies genau die Aktion ist die tatsächlich benötigt wird? Diese Frage schließt an die vorherige Frage an. Die Frage nach der Sichtbarkeit ist unabhängig davon ob der Nutzer auch erkennt das dies auch die Aktion ist die ausgeführt werden muss um die

gewünschte Aufgabe zu erfüllen. Diese Frage zielt darauf ab ob die Bedeutung und Effekt der Funktion klar sind, nicht auf die Sichtbarkeit.

4. Nachdem die Aktion ausgeführt wurde, wird dem Nutzer das gegebene Feedback verständlich sein? Wenn der Nutzer die korrekte Aktion ausgeführt hat, wird er auch darüber informiert das dies die richtige Aktion war? Ist das Feedback ausreichend informativ?

Cognitive Walkthrough wurde auch an komplexe Interfaces angepasst, beispielsweise im Computer Aided Design (CAD) [29] oder bei der Entwicklung von Medizinischen Geräten [3]. Die Methode wurde außerdem in der Evaluation von Betriebsabläufen eingesetzt [21].

Modellbasierte Evaluation

Eine weitere Variante der Expertenanalyse ist die sogenannte Modellbasierte Evaluation [14]. Dabei wird die Evaluation durchgeführt indem ein Modell genutzt wird welches darstellt, wie ein Mensch ein bestimmtes System nutzen würde. Aus diesem Modell werden Usabilitywerte prognostiziert, entweder mit Hilfe einer Simulation oder von Berechnungen. Diese Prognosen können empirische Messungen mit Hilfe von Nutzertests ersetzen oder unterstützen. Zusätzlich enthält das Modell selbst nützliche Informationen die Beziehung zwischen der Aufgabe des Nutzers und dem Design des Systems. Dieser Ansatz basiert auf einer detaillierten Beschreibung des voraussichtlichen Designs und einer detaillierten Aufgabenanalyse. Es erklärt wie die Nutzer Aufgaben erfüllen, wenn sie mit dem voraussichtlichen Interface interagieren. Mit Hilfe von psychologischer Theorie und parametrisierten Daten werden Usability Metriken generiert. Wenn das Modell erstellt ist können die Usability-Vorraussagen schnell und einfach mit Berechnungen oder Simulationen gewonnen werden. Es gibt drei vorherrschende Modellarten, GOMS (goals, operators, methodes and selection) Modelle, Task Network Modelle und Cognitive Architecture Modelle.

1. Das GOMS Modell berechnet die Nutzerperformance für ein bestimmtes Interface im voraus. Über das Akronym lässt sich das Modell gut beschreiben. Der Nutzer kann bestimmte Ziele (**[G]oals**) mit dem System erreichen. Eingabeoperationen (**[O]perators**) sind die Hauptaktionen mit welchen mit dem System kommuniziert wird, gemeint sind hierbei Tastendrucke oder Mausklicks. Methoden (**[M]ethods**) sind Sequenzen von Eingabeoperationen welche, wenn sie ausgeführt werden, ein Ziel erreichen. Auswahlregeln (**[S]election rules**) beschreiben welche Methode in welcher Situation genutzt werden sollte um ein Ziel zu erreichen, falls mehr als eine Methode verfügbar ist.

2. Bei Task Network Modellen wird die Performance eines Tasks mit Hilfe von Netzwerkprozessen modelliert. Jeder Prozess startet wenn seine Vorgängerprozesse beendet wurden und hat eine angenommene Verteilung von Durchlaufzeiten. Dabei müssen Tasks oder Prozesse nicht zwingend vom User ausgeführt werden sondern können auch vom System ausgelöst werden. Die Performancevoraussagen werden mit Hilfe einer Monte-Carlo Simulation der Modellaktivität getätigt. In dieser Simulation werden die Inputevents von Zufallsvariablen oder von Taskszenarios generiert. Eine Vielzahl statistischer Ergebnisse werden dabei produziert, beispielsweise die Ressourcennutzung und die Auslastung des Systems.
3. Cognitive Architecture Modelle bestehen aus hypothetischen interagierenden Komponenten, dabei wird angenommen das diese sich im Menschen befinden. Diese Computersysteme simulieren einen Menschen welcher in einer Simulierten Umgebung, Input von dieser erhält und Output für diese generiert, worauf die Umgebung wieder reagiert. Die Tasks welche von dem Kognitiven Modell ausgeführt werden sollen, werden nach einer Taskanalyse in die Kognitive Komponente integriert. Da diese Systeme versuchen die physiologischen Funktionen des Menschen abzubilden sind diese komplex aufgebaut und werden eher in Grundlegenden Forschungsprojekten eingesetzt.

Verwendung von vorangegangenen Studien

Im Bereich der Human-Computer Interaction und der Experimentellen Psychologie gibt es viele Ergebnisse aus Experimenten und empirische Beweise. Einige davon können spezifisch einer Domäne zugeordnet werden, andere haben einen etwas weiter gefassten Ansatz. Diese bereits gewonnenen Informationen gehen von der Usability von speziellen Menütypen, über die Erinnerung an spezifische Kommandonamen, bis hin zu der Wahl von Darstellungen und deren Farben. Dieser Ansatz macht sich die bereits gewonnen Erkenntnisse zu nutze, indem vorherige Ergebnisse verwendet werden um die Wahl für ein bestimmtes Design oder eine Visualisierung zu stärken oder abzuschwächen. Es ist teurer Experimente zu wiederholen und die Rezension und Analyse bestehender Literatur kann dies obsolet machen. Es sollte dabei bedacht werden das experimentelle Ergebnisse möglicherweise nicht in beliebigen Kontext Bestand haben. Ein Rezensent muss daher die Belege sorgfältig auswählen und dabei das experimentelle Design, die Population der Teilnehmer, die ausgeführten Analysen und gemachten Annahmen beachten.

2.1.3 Evaluationstechniken mit Nutzerbeteiligung

Die im vorherigen Abschnitt beschriebenen Evaluationstechniken fokussieren sich auf die Ausführung durch den Designer oder Experten und nicht mit tatsächlichen Nutzern.

2.2 Einführung in grundlegende Visualisierungen und Werkzeuge

Visualisierungen und Werkzeuge wie zum Beispiel Informationssysteme werden in vielen Bereichen der Wissenschaft und Wirtschaft eingesetzt. Dieser Abschnitt möchte einen Überblick über Methoden und Werkzeuge geben und dabei auch in angrenzende Wissensbereiche vorstoßen.

Um numerische Daten in konkreten Visualisierungen festzuhalten eignen sich räumliche Positionsangaben, laut graphischen Wahrnehmungsexperimenten, am besten [10]. Bereits 1786 entwickelte William Playfair die noch heute weit verbreiteten, positionskodierten, Balken- und Liniendiagramme [22]. Zur Visualisierung von, beispielsweise verfügbarer und/oder genutzter Bandbreite bieten diese Darstellungen noch immer einen angemessenen Informationsgehalt. Diese gehören weiterhin zu den Standarddarstellungsformen, nicht nur für Netzwerkdaten. Ebenso weit verbreitet ist das, ebenfalls von Playfair entwickelte, Kreisdiagramm (auch als Torten- oder Kuchendiagramm bekannt) zur Darstellung von Teilwerten eines Ganzen [23]. Ein Kreisdiagramm kann beispielsweise Verwendung finden um die genutzte Bandbreite verschiedener Links darzustellen.

Weitere naheliegende Darstellungsarten sind die der Netzwerkvisualisierungen. In diesen werden Beziehungen zwischen Daten aufgezeigt. Das kann sowohl in sozialen als auch in physischen Netzwerken sinnvoll sein, aber auch für Datenbanken oder auch Unternehmensstrukturen. Im mathematischen Sinne sind Netzwerke Graphen und bestehen im Grunde aus zwei primitiven Elementen, Knoten und Kanten, wobei die Kanten die Knoten verbinden. Ein zentraler Punkt bei der Visualisierung mit Graphen stellt die Auswahl eines effektiven Layouts sowie der gewünschte Detailgrad der Analyse dar. Typischerweise werden stark verbundene Knoten auch im Bild nah aneinander positioniert, ebenso werden nicht verwandte Knoten weiter voneinander entfernt platziert, um die Beziehungen differenzieren zu können. Im folgenden werden einige mögliche Netzwerkvisualisierungen besprochen.

[TODO: Zwei gängigere Methoden noch.]

Force-directed Layout. Bei diesem Layout wird der Graph als physisches System modelliert. Knoten sind geladene Partikel welche sich gegenseitig abstoßen, die Verbindungen

zwischen diesen fungieren als Federn und halten die Knoten zusammen. Mit Hilfe einer physischen Simulation dieser Kräfte (forces), werden die Positionen der Knoten festgestellt. Hinzu kommt das der Nutzer mittels Interaktivität das Layout ordnen kann wobei die physikalischen Eigenschaften beibehalten werden. Entwickelt wurde dieses Modell von Fruchterman und Reingold [8].

Matrix View. Ungerichtete Graphen können mathematisch als Adjazenzmatrix beschrieben werden. Für ein Netzwerk, im Grunde ein ungerichteter Graph, bietet es also an eine solche Darstellung zu verwenden. Jeder Wert in Zeile i und Spalte j in der Matrix korrespondiert zu der Anzahl der Links von Knoten i zu Knoten j . Ghoniem und Fekete beschreiben beispielsweise in Ihrer Abhandlung die Implementation dieser Technik [9].

Eine noch detaillierte Einführung in die Netzwerk-Analyse mit Hilfe von Graphen bietet die Arbeit von Brandes et. al [4]. [TODO: Details hier her?] ...

Mengen mit Werten die sich über die Zeit verändern, sind eine der häufigsten Formen von aufgezeichneten Daten. Auch im Netzwerk häufen sich Daten welche über die Zeit veränderlich sind, wie die genutzte Bandbreite, die Veränderung der Latenz oder des Jitters. Es gibt einige Visualisierungen die diese Art der Darstellung von sogenannten Time-Series Data/Zeitreihendaten unterstützen.

Small Multiples. Mehrere Zeitreihen können innerhalb von nur einem Diagramm dargestellt werden. Dadurch können allerdings Überschneidungen entstehen wodurch sich die Lesbarkeit verringern kann. Eine Alternative ist die Darstellung in Small Multiples, hier wird jede Zeitreihe Achsen-gleich in einem eigenen Diagramm dargestellt. Durch diese Darstellung können allgemeine Trends sowie individuelle Muster erkannt werden. In Projekt 1 wurde diese Darstellungsform bereits in kleinem Rahmen gewährleistet.

In Burch und Weiskopfs Abhandlung [6] wird diese Darstellungsform zur Verdeutlichung der Dynamik von Graphen genutzt. Im Small Multiple Aufbau werden Node-Link Diagramme verwendet, welche die Veränderung des Graphen Sequentiell vermitteln soll. Die Vorteile eines Small Multiple Aufbaus gegenüber, von den Autoren als sogenannte Large Singles bezeichnete, einfachen Darstellungen wird in van den Elzen und van Wijk's Artikel erläutert [28]. Die Kernaussage besagt, dass mit Small Multiples ein breiterer Umfang an Daten aufgenommen werden kann als mit Large Singles.

Index Charts. Für einige Zeitreihen ist die relative Veränderung ausschlaggebender als die tatsächliche Veränderung der Daten. Für diese Daten kommt ein Index Chart in Frage. Dieser Diagrammtyp zeigt die prozentuale Veränderung für eine Sammlung von Zeitreihen basierend auf einem fixen indexiertem Punkt [10]. Diese Diagrammart kann beispielsweise bei der Visualisierung von Netzwerkdaten eingesetzt werden um bei einer Verbindungen

festzustellen wie sich die Last ab einem bestimmten Zeitpunkt prozentual erhöht. Als Zeitpunkt in einem Fahrzeug beispielsweise das Anlassen des Motors. So lässt sich analysieren welche Verbindungen den größten prozentualen Lastzuwachs, ab diesem Zeitpunkt, erfahren.

...Naheliegender angrenzende Bereiche sind jene aus dem Verkehrswesen. In der Flugzeugtechnik wird bereits mit Ethernet im Flugzeugnetz experimentiert und es ist auch schon im Einsatz, beispielsweise im A380 oder der Boeing 787. Auch hier wurde der erhöhte Kommunikationsbedarf erkannt und mit einem auf die Luftfahrt spezialisiertem und von Airbus entwickelten Datennetz reagiert. Das sogenannte Avionics Full-Duplex Switched Ethernet (AFDX) wird hauptsächlich für sicherheitskritische Anwendungen verwendet [27].

2.3 Grundlagen für Anforderungen und Analyse

Dieser Abschnitt beschreibt grundlegende Aspekte ...

Damit visuelle Analyse Tools effektiv genutzt werden können, müssen diese eine fließende und flexible Nutzung von Visualisierungen ermöglichen. Um das zu ermöglichen wird in [11] eine *Taxonomy of interactive dynamics for visual analysis* eingeführt. Diese ist in drei Kategorien eingeteilt welche die kritischen Aufgaben der visuellen Analyse beschreiben. Die Kategorien sind *Data and View Specification*, *View Manipulation* sowie *Process and Provenance*. Die erste Kategorie beschäftigt sich mit den spezifischen Daten und Views welche für die jeweiligen Bedürfnisse der Analysten von Interesse sind. Es muss eine Steuerung der Applikation ermöglicht werden, so dass die Daten selektiv visualisiert werden können, irrelevante Daten herausgefiltert werden und ebenso müssen die Informationen sortiert werden können um mögliche Muster erkennbar zu machen. Nach der Erstellung einer Visualisierung muss es den Analysten ermöglicht werden die gestellten Views nach Bedürfnis zu manipulieren, dies beschreibt die zweite Kategorie der Taxonomy. Die Manipulation der Visualisierung ist nötig um Muster aufzeigen zu können, Hypothesen zu untersuchen und zur Navigation in den Daten bspw. via Drilldown. Analyse Tools muss es möglich sein multiple verbundene Visualisierungen zu organisieren und zu koordinieren um mehr Einsicht in Multidimensionale Daten zu bieten als es isolierte Views vermögen. Dies wird schon im häufig zitierten Mantra von Ben Shneiderman vorgegeben: "*Overview first, zoom and filter, then details-on-demand*"[24]. Die abschließende Kategorie *Process and Provenance* beschreibt den iterativen Prozess der Datenexploration sowie den der Dateninterpretation. Wenn Analysten Ihre bisher getätigten Aktionen innerhalb eines Programms nachverfolgen können kann die Arbeit überprüft und verbessert werden. Es sollte möglich sein die Ergebnisse zu dokumentieren als auch diese mit anderen zu teilen um eine Diskussion darüber zu ermöglichen. Ferner können Analysewerkzeuge auch Anfänger durch

Ihre Funktionen führen um die Benutzung und den Einstieg zu erleichtern. Die vorgegebene Taxonomie bietet einen guten Einstiegs- und Orientierungspunkt in der Welt der Visualisierung und kann ebenso als Checkliste bei der Entwicklung eines neuen Analysetools verwendet werden. Sie zeigt viele Möglichkeiten auf um den Daten Herr werden zu können.

3 Evaluationsbasis

In den letzten 20 Jahren beschäftigten sich über 80% der Evaluationen bezüglich der Visualisierung und ihrer Werkzeuge mit den sich ergebenden Bildern und mit der Performance der verwendeten Algorithmen. Allerdings lassen mehr und mehr Evaluationen Teilnehmer in ihren Studien partizipieren. Von den Teilnehmern wird dann die Leistungen sowie das subjektive Feedback Evaluiert oder es wird ihre (verbesserte) Performance mit der Visualisierung und in der Analyse, als auch bei der Fähigkeit zur Schlussfolgerung berücksichtigt [12].

Die Wahl der Art der Evaluation ist entscheidend, weil diese vorgibt in welcher Art und Weise die Ergebnisse genutzt werden können. Diese Entscheidung wird von mehreren Faktoren beeinträchtigt. Den Zielen welche mit der Methode erreicht werden sollen, die Umstände und Grenzen die der Evaluationsaufgabe auferlegt sind und der Charakteristik der Objekte welche evaluiert werden sollen. Diese unterschiedlichen Faktoren interagieren miteinander was die Wahl der geeignetsten Evaluationsmethode schwierig gestaltet [16].

Lam et al. haben in ihrer Ausarbeitung [17] 850 Paper hinsichtlich möglicher Evaluierungsszenarien untersucht. Sie haben sieben verschiedene Szenarien ermittelt, welche sich in zwei grundlegende Kategorien aufteilen. Zum einen gibt es Szenarien, welche mit dem Verständnis der eigentlichen Visualisierungen beschäftigen, zum anderen solche die zum Nachvollziehen der Datenanalyse dienen. Besonders das Szenario aus der Datenanalyse **evaluating visual data analysis and reasoning (VDAR)** sticht für die angestrebte Arbeit heraus. Evaluationen in diesem Bereich fokussieren sich hauptsächlich darauf ob und wie ein Visualisierungswerkzeug visuelle Analysen und Schlussfolgerungen über Daten zulässt. Mögliche Ergebnisse sind hierbei sowohl Metriken (bspw. Anzahl der Analyseerkenntnisse), als auch subjektives Feedback (bspw. Umfrage über Datenanalysequalität). Meist werden in diesem Bereich Feldstudien genutzt, häufig in der Form von Fallstudien.

[TODO: Korf: Anforderungen an diese Methoden und was kann man daraus gewinnen]

3.1 Häufig verwendete Evaluationen im Bereich Visualisierung

Visualisierung/Automotive/Angrenzende

3.2 Faktoren zur Abgrenzung der Evaluationstechniken

3.3 Entscheidung für Evaluation

dieser oder die Kombi aus diesen ist für Automotive am besten weil! Und dieser wird für die Literaturanalyse bzw. Analyse von bestehenden Visualisierungen und Werkzeugen verwendet.

4 Literaturauswertung

Dieses Kapitel beleuchtet die für diese Arbeit relevante Literatur. Das Kapitel ist in zwei Bereiche gegliedert, zum einem werden theoretische Abhandlungen betrachtet welche sich mit Anforderungen an vornehmlich Netzwerk bezogenen Visualisierungen beschäftigen. Zum Anderen...

4.1 Verwandte Arbeiten

4.2 Visualisierungen und Werkzeuge

Es wurde bereits eine große Zahl an verschiedensten Analyse-Werkzeugen entwickelt. Dieser Abschnitt geht auf die Funktion, die Nutzbarkeit und das Anwendungsgebiet der für diese Arbeit relevanten Programme ein. Die Programme kommen aus den verschiedensten Bereichen, der Automatisierungstechnik, der Flug- und Fahrzeugindustrie so wie weiteren kleineren Bereichen. Es handelt sich dabei sowohl um proprietäre, als auch um wissenschaftlich erarbeitete Anwendungen.

5 Anforderungsanalyse

5.1 Allgemeine Anforderungen

Dieser Abschnitt soll die unterschiedlichen Anforderungen welche an Visualisierungswerkzeuge, ebenso wie an Visualisierungstechniken gestellt werden, herausarbeiten. Im folgenden werden dann die essentiellen Anforderungen hervorgehoben um in einem letzten Schritt diese auf ihre Anwendbarkeit im Automotivebereich geprüft.

5.1.1 Anforderungen an Visualisierungen

Anforderungen an Visualisierungen:

- Verständlichkeit und schneller Zugriff auf Daten
- Zeitersparnis bei der Analyse
- Kollaboratives Arbeiten
- Datenabstraktion, Überblick über Daten

5.1.2 Anforderungen an Werkzeuge

Spez. Anforderungen an Werkzeuge:

- Große Datenmengen verarbeiten/Datenexploration
- Neue Erkenntnisse schaffen
- Visuelle Analysen mit Schlussfolgerungen über Daten

5.2 Automotive spezifische Anforderungen

Visualisierungen Automotive spezifischer:

- Verbesserte temporale Struktur und Orientierung
 - schnellere Übersicht
 - bessere Navigationsmöglichkeiten als in Liste
 - grobe sowie feine Zeitinformationen nötig
- Zugriff auf Rohdaten muss gegeben sein
 - Signale, Nachrichten und deren Werte müssen abrufbar sein
 - Abstraktion kann helfen, aber nicht um Werte aus Rohdaten anzuzeigen

Werkzeuge mögliche Evaluierungsrichtungen:

- Verständnis der Umgebung und Arbeitspraktiken (In welchem Kontext wird es verwendet? Welche Vis?)
- Evaluation der visuellen Datenanalyse und Schlussfolgerungen (Entscheidungsfindung, Hypothesengenerierung, Wie wird das unterstützt?)
-

6 Anwendungskonzepte und -bereiche

Würde man es nach dem und dem Ansatz machen wäre das ergebnis so und so
kombiniert man den und diesen Ansatz dann wäre das so
daraus ergibt sich (hoffentlich) ein völlig neuer Ansatz

7 Zusammenfassung und Fazit

Literaturverzeichnis

- [1] Alan Dix, J. Finley, G. D. Abowd, and R. Beale. Human-Computer Interaction. In *Encyclopedia of Database Systems*, pages 1327–1331. Springer US, 2004. DOI: 10.1007/978-0-387-39940-9_192.
- [2] K. Andrews. Evaluating Information Visualisations. In *Proceedings of the 2006 AVI Workshop on BEyond Time and Errors: Novel Evaluation Methods for Information Visualization*, BELIV '06, pages 1–5, New York, NY, USA, 2006. ACM.
- [3] L.-O. Bligård and A.-L. Osvalder. Enhanced Cognitive Walkthrough: Development of the Cognitive Walkthrough Method to Better Predict, Identify, and Present Usability Problems. *Adv. in Hum.-Comp. Int.*, 2013:9:9–9:9, Jan. 2013.
- [4] U. Brandes and T. Erlebach. *Network analysis: methodological foundations*, volume 3418. Springer Science & Business Media, 2005.
- [5] M. Broy, S. Chakraborty, D. Goswami, S. Ramesh, M. Satpathy, S. Resmerita, and W. Pree. Cross-layer Analysis, Testing and Verification of Automotive Control Software. In *Proceedings of the Ninth ACM International Conference on Embedded Software*, EMSOFT '11, pages 263–272, New York, NY, USA, 2011. ACM.
- [6] M. Burch and D. Weiskopf. A flip-book of edge-splatted small multiples for visualizing dynamic graphs. In *Proceedings of the 7th International Symposium on Visual Information Communication and Interaction*, VINCI '14, pages 29:29–29:38, New York, NY, USA, 2014. ACM.
- [7] M. Franzke. *Exploration and experienced performance with display-based systems*. PhD thesis, University of Colorado, 1994.
- [8] T. M. J. Fruchterman and E. M. Reingold. Graph drawing by force-directed placement, 1991.

- [9] M. Ghoniem and J.-D. Fekete. Matrix view of graphs and direct manipulation of cluster hierarchies. In *Proceedings of the 15th Conference on L'Interaction Homme-Machine, IHM '03*, pages 206–207, New York, NY, USA, 2003. ACM.
- [10] J. Heer, M. Bostock, and V. Ogievetsky. A Tour Through the Visualization Zoo. *Commun. ACM*, 53(6):59–67, June 2010.
- [11] J. Heer and B. Shneiderman. Interactive Dynamics for Visual Analysis. *Queue*, 10(2):30:30–30:55, Feb. 2012.
- [12] T. Isenberg, P. Isenberg, J. Chen, M. Sedlmair, and T. Möller. A Systematic Review on the Practice of Evaluating Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2818–2827, Dec. 2013.
- [13] D. Kane. Finding a place for discount usability engineering in agile development: throwing down the gauntlet. In *Proceedings of the Agile Development Conference, 2003. ADC 2003*, pages 40–46, June 2003.
- [14] D. Kieras. The Human-computer Interaction Handbook. pages 1139–1151. L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 2003.
- [15] B. A. Kitchenham. Evaluating Software Engineering Methods and Tool Part 1: The Evaluation Context and Evaluation Methods. *SIGSOFT Softw. Eng. Notes*, 21(1):11–14, Jan. 1996.
- [16] B. A. Kitchenham. Evaluating Software Engineering Methods and Tool—Part 2: Selecting an Appropriate Evaluation Method—Technical Criteria. *SIGSOFT Softw. Eng. Notes*, 21(2):11–15, Mar. 1996.
- [17] H. Lam, E. Bertini, P. Isenberg, C. Plaisant, and S. Carpendale. Empirical Studies in Information Visualization: Seven Scenarios. *IEEE Transactions on Visualization and Computer Graphics*, 18(9):1520–1536, Sept. 2012.
- [18] C. Lewis, P. G. Polson, C. Wharton, and J. Rieman. Testing a Walkthrough Methodology for Theory-based Design of Walk-up-and-use Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '90*, pages 235–242, New York, NY, USA, 1990. ACM.
- [19] J. Nielsen. *Usability Engineering*. Elsevier, Nov. 1994. Google-Books-ID: DBOowF7LqIQc.

- [20] J. Nielsen and R. Molich. Heuristic Evaluation of User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '90, pages 249–256, New York, NY, USA, 1990. ACM.
- [21] D. G. Novick. Using the Cognitive Walkthrough for Operating Procedures. *interactions*, 6(3):31–37, May 1999.
- [22] W. Playfair. Playfair's Commercial and Political Atlas and Statistical Breviary, 1768.
- [23] W. Playfair. *The statistical breviary shewing, on a principle entirely new, the resources of every state and kingdom in Europe ... to which is added, a similar exhibition of the ruling powers of Hindoostan*. For J. Wallis, London, 1801.
- [24] B. Shneiderman. The eyes have it: a task by data type taxonomy for information visualizations. In *Visual Languages, 1996. Proceedings., IEEE Symposium on*, pages 336–343, Sep 1996.
- [25] B. Shneiderman and C. Plaisant. *Designing the user interface: strategies for effective human-computer interaction*. Pearson/Addison-Wesley, Boston [u.a.], 4 edition, 2005.
- [26] T. Steinbach, F. Korf, and T. Schmidt. CoRE Group, 2017. <http://core.informatik.haw-hamburg.de/en/>, Zugriffsdatum: 17.02.2017.
- [27] W. Steiner, P. Heise, and S. Schneelee. Recent IEEE 802 developments and their relevance for the avionics industry. In *2014 IEEE/AIAA 33rd Digital Avionics Systems Conference (DASC)*, pages 2A2–1–2A2–12, Oct. 2014.
- [28] S. van den Elzen and J. J. van Wijk. Small multiples, large singles: A new approach for visual data exploration. *Computer Graphics Forum*, 32(3pt2):191–200, 2013.
- [29] C. Wharton, J. Bradford, R. Jeffries, and M. Franzke. Applying Cognitive Walkthroughs to More Complex User Interfaces: Experiences, Issues, and Recommendations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '92, pages 381–388, New York, NY, USA, 1992. ACM.

Glossar

HCI Human Computer Interaction, Seite 14

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe.

Hamburg, 19. März 2017

Benjamin Burchard