RWTH AACHEN UNIVERSITY

Analyzing Net

A User Interface for Complex Systems Based on Mapping Principles

Diploma Thesis at the
Media Computing Group
Prof. Dr. Jan Borchers
Computer Science Department
RWTH Aachen University

by
Monika Wernerus

Thesis advisor:
Prof. Dr. Jan Borchers

Second examiner:
Prof. Dr. Thomas Rose

Erklärung

Hiermit versichere ich, dass ich die vorliegende Diplomarbeit selbständig und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt habe. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Schriften entnommen sind, wurden als solche kenntlich gemacht.

Aachen, March 25, 2008

Contents

List of Figures

List of Tables

Abstract

This thesis describes the design of a new interface for a control center in the domain of power transmission. The interface is based on visualization of associations between information, the basic principle of mapping. The associations and additional visual clues guide the operator to information, he needs to fulfill a task at hand.

Before starting the design process, I did intensive research about the domain of power transmission and the working context of a control center, by visiting a demonstration room, interviewing experts, and reading reports about site visits. During a demonstration I identified problems occurring in the user interface of a control center. I made a task analysis of the contingency analysis, which is used as example task.

I developed the interface in three cycles of an iterative design process. In the first cycle I used two paper prototypes to compare two different design approaches and to check for violated constraints violations. The evaluation of the second prototype, implemented as screenshot prototype, was used to strengthen my assumptions. The final prototype was implemented in C# using GoDiagram and XML. Parts of the evaluation took place at a control center of a power grid.

Finally I developed approaches to evaluate a new control center interface and identified major evaluation problems in the domain.

The work done for this thesis results in an interactive prototype, implementing associations for navigation purposes in a control center, an example for structuring the information, and suggestions for further evaluation.

Überblick

Diese Arbeit beschreibt das Design einer neuen Benutzerschnittstelle für ein Kontrollzentrum im Bereich Stromübertragung. Es basiert auf der Visualisierung von Verbindungen zwischen Informationen, dem Grundprinzip von Mapping. Die Verbindungen und weitere visuelle Hinweise führen den Operator zu den Informationen, die er während der Bearbeitung einer Aufgabe benötigt.

Vor dem Beginn des Designprozesses habe ich intensive Untersuchungen der Domäne Stromübertragung und des Arbeitsumfeldes in einem Kontrollzentrum betrieben, indem ich einen Demonstrationsraum besucht, Experten interviewt und Berichte über Site Visits studiert habe. Während einer Demonstration habe ich Probleme identifiziert, die mit einer Benutzerschnittstelle in einem Kontrollzentrum auftreten können. Ich habe eine Task Analyse einer Ausfallvariantenrechnung durchgeführt, die als Beispielaufgabe dient.

Ich habe die Benutzerschnittstelle in drei Zyklen eines iterativen Designprozesses entwickelt. Im ersten Zyklus habe ich zwei Papierprototypen benutzt, um zwei verschiedene Designansätze zu vergleichen und auf Verletzungen der Nebenbedingungen zu überprüfen. Die Evaluierung des zweiten Prototyps, ein Screenshotprototyp, wurde benutzt, um meine Annahmen zu festigen. Der letzte Prototyp wurde in C# unter der Verwendung von GoDiagram und XML implementiert. Ein Teil der Evaluierung fand in einem Kontrollzentrum statt.

Zum Schluß habe ich einige Ansätze zur Evaluierung von neuen Benutzerschnittstellen eines Kontrollzentrums entwickelt und größere Probleme bei der Evaluierung in der Domäne identifiziert.

Das Ergebnis der Arbeit ist ein interaktiver Prototyp, der Verbindungen zu Navigationszwecken in einem Kontrollzentrum realisiert, ein Beispiel zur Strukturierung der Informationen und Vorschläge zur weiteren Evaluierung.

Conventions

Throughout this thesis I use the following conventions.

The whole thesis is written in American English.

Instead of avoiding the singular pronoun, I decided to use 'he', whenever I am referring to the operator, in other cases I use 'she' as generic term. This should not be taken to to imply anything about the composition of any actual population.

Chapter 1

Introduction

1.1 Motivation

About ten million people were affected by the major europeen blackout in November 2006. During the visit at a control center of a power grid an operator referred to the blackout [Siemens AG, 2007]:

> Die Netzanalyse braucht eine bessere Oberfläche![1]

This example shows, that the user interface of a control center needs to be improved. Nowadays, most of the interfaces are designed by merging all applications in one interface. Information and applications, which are needed to perform one single task, are spread all over the system. Thus, the operator spends a lot of effort on navigating to the desired information.

A reorganization of the user interface is required. To fully support the work of an operator, the information should be ordered depending on the task at hand. By making the relations between the different types of information visible, the operator may be guided to important information.

[1] translation: Grid analysis requires a new user interface!

According to a model of the human memory new information is remembered by creating associations to existing knowledge. The associations are also used to recall information. Mapping techniques, like mind maps and concept maps, use associations to visualize information and their relationships.

An advantage of using such a natural way of connecting information is quick learnability and orientation by visualizing the context. The idea of using associations to visualize relationships between information is transferred to a control center user interface. Visualizing the context of information, should help the operator to navigate in the information space.

The goal of this thesis is to develop a way to use mapping principles for guiding the operator during a task completion. During the design process it is examined, how associations can be integrated in a control center user interface to support the operator in finding information. In addition, it is investigated, how visual clues help to guide to important information.

1.2 Structure of This Work

Chapter 1 - Introduction: Chapter one gives the motivation for this thesis and an overview of the structure of this thesis.

Chapter 2 - Domain: The second chapter covers the domain of power transmission. First, I will explain power transmission and the main characteristics of a power grid. Then a control center of a power transmission grid and its working context is described. The state-of-the-art of operating systems in a control center is introduced, before their main interface problems are identified. As example task the contingency analysis and its handling is presented and analyzed.

Chapter 3 - Theory: Chapter three introduces the theory of two main concepts of my work. Mapping, in particular mind maps and concept maps, uses the visualization of associations to match a model of human memory—the semantic network. Zooming user interfaces face the opposition of a big information space and limited screen space by adding the information on a infinitely large plane and the access to the information by panning and zooming.

Chapter 4 - Related Work: Several projects, which have influenced my design, are presented in chapter four. A project of the same domain is presented: A visualization of the contingency analysis data. CounterPoint and Fly, two presentation tools, are described as examples of information presentation with zooming interfaces. The use of associations to organize information is further demonstrated with The Big Picture and Personal Brain. With OZONE a project, using semantic zooming, is depicted.

Chapter 5 - Low-fidelity Prototypes: In chapter five the first two design cycles will be described. Two paper prototypes were created to compare different design aspects. An expert review helped to decide, which alternatives were integrated into the design. The paper prototypes, the results and the decisions will be presented. An intermediate screenshot prototype and its evaluation was used to strengthen some assumptions, before realizing in a further prototype. I will describe the prototype, the evaluation, and the results. In addition, I will introduce the content structure used in the design.

Chapter 6 - High-fidelity Prototype: Chapter six covers the software prototype. I will describe the design, the implementation and the evaluation. The prototype is implemented in C# using a library called GoDiagram and XML. The results of two evaluation rounds—with engineers and at a control center—will be presented.

Chapter 7 - Evaluation: In chapter seven I will introduce possibilities to evaluate the design against existing interfaces, and explain the occurring challenges.

Chapter 8 - Summary and future work: The eighth chapter summarize the thesis and gives a perspective for future work.

Chapter 2

Domain

In this chapter I will introduce the domain of power transmission. Therefore, I will first explain the term power transmission and all technical facts, that are important for this thesis. The next sections describe the working context in a control center and the operating system with its problems. The last section introduces a typical task during a day of an operator and describes the work flow of this task.

2.1 Power Transmission

Transmission vs. distribution

The transportation of electric power from a power plant to a consumer is separated into two parts. The part from the power plant to a substation is called energy transmission and the part from the substation to the consumer is called power distribution. The distribution networks are smaller than the transmission networks, which span over long distances [von Meier, 2006]. Figure 2.1 is an abstraction of this transportation.

Grids in Germany and Europe as example

An example for the size such a grid may have: In Germany four companies are responsible for energy transmission and even their grids are interconnected to one big grid, which itself is connected to the European network. Figure 2.2 gives an impression on the size of a grid in Germany by showing the four parts of the grid in Germany.

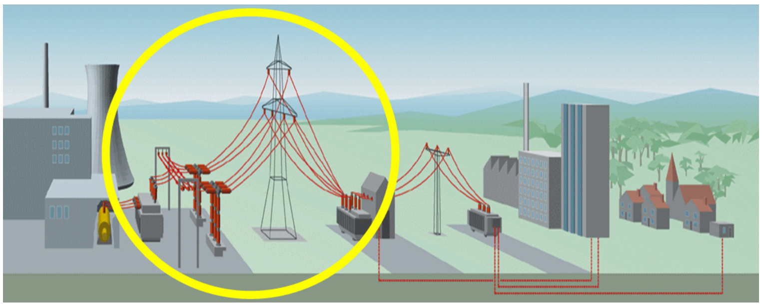

Figure 2.1: Power supply from power plant to consumer; The part of power transmission is circled in yellow.

Figure 2.2: Germany's power transmission is split in four parts

To keep the power loss low over such long distance, power transmission takes place at high or highest voltage, i.e., 110kV or higher (in Europe 220kV or 380kV).

High to highest voltage

Although the network is often called power grid, it is not a real grid. It consists of several redundant pathes, which transport the power from power plants to substations, where it is distributed to the customer. The pathes have to be redundant to keep the grid in a N-1 security. That means the system must keep running, even if any single item fails and every customer must be provided at any time.

Redundant pathes for N-1 security

The grid is maintained by operators in a control center. The system must be kept in a state of equilibrium. For all power, that is taken out of the system, the same amount has to be put into the system. It is the operator's task to keep the equilibrium. He has to monitor and direct the ongoing reconfigurations, that means diagnostics, switching, and coordination of reparations. The grid is controlled in real time.

Grid runs in an equilibrium

In fulfilling his tasks the operator has to take several challenges. External factors must be taken into account, because they can influence the system. The weather can influence the amount of power, which is consumed. During cold periods for example more power is needed for the heating, or on cloudy days the use of more light causes a higher consumption. On the other hand, the limit of loads at a transmission line changes changes with the humidity and the temperature. During storms or thunderstorms the devices are in danger of getting damaged. The characteristics of the point of time also changes the amount of consumption. It depends on the season, on the hours of work, on the hours of light and so on. In case of constructions, elements of the grid may be switched off and it is impossible to activate them. The operator needs to know about these factors and take them into consideration, when maintaining the grid.

Challenges for an operator

Another challenge are uncertainties in the real-time system states, that means system parameters may be unknown or not known exactly.

Uncertainty of system state

A change in the grid at one point may have consequences

Operations may have unwanted effects

at a different and possibly far away point and every disturbance may cause a cascade of effect with possibly huge consequences.

Time pressure

But the hardest challenge is, that the operator has to cope with these challenges in real time, and thus, always working under time pressure.

The working context of an operator will be described in more detail in the next section.

2.2 Working Context in a Control Center

When designing a new interface it is important to take the working context into consideration [Dix et al., 1997]. The following description of the working context in a control center is based on site visits in control centers of power transmission in Switzerland and the USA. During these site visits the operators were observed and questioned while working [Siemens AG, 2006a,b].

Crew in a control center

In a control center several people are working at the same time. They either fulfill different functions, like operator, operational planner, or they are responsible for different areas, which are bundled in one control room. Nonetheless, they are responsible for the grid altogether and need to communicate and help each other in case of emergencies

Several large and small screens

There is at least one wall-sized screen for everyone working in the control room. In addition every operator on duty has several displays at his work place, as in figure 2.3.

Same system used by everyone

The work is done in shifts to keep the control center manned every day for 24 hours. Some workers, like the operational planner, are working only in day shifts, but an operator must be present all the time. Different people have to work with the same system, either at the same time or in different shifts. That means during a shift change information about important incidents has to be transferred between the operators. At the different work places in the control center the same system may be used with different

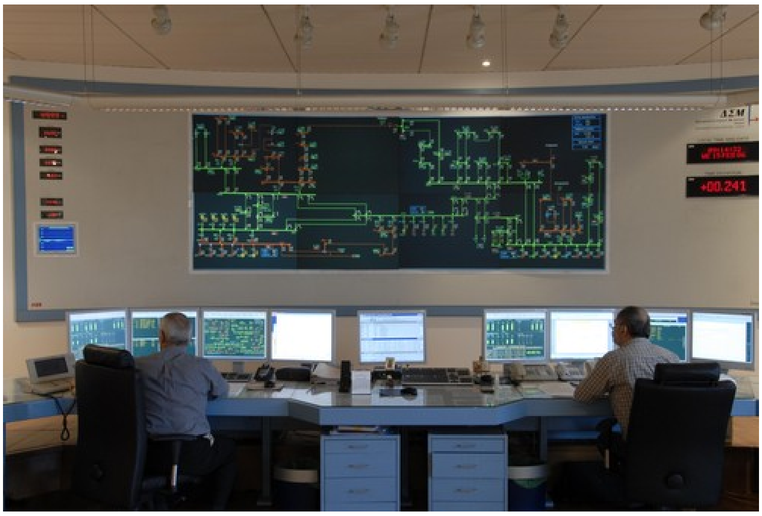

Figure 2.3: Control center of Cyprus as typical but relatively small control center

focuses. The operator and the operational planner use the same systems and a lot of similar applications, but while the operator is always working in the real time mode the operational planner is always working in the study mode.

Communication is crucial

In addition to the computer system, there are two main facilities the operator may work with. There is a telephone at each work place, which is important for the external and internal communication, for example with field crews, customers, or operators of neighboring grids. A good communication is essential for the work in a control center.

Documents for additional information

Documents, like manuals, schedules, or external information, either in printed form or on a different computer system, are essential for the operator's work. (See later in this section.)

Operator has high responsibility

An operator bears a big responsibility during his work. He has to keep the grid stable at any time. Therefore, he needs to react to changing conditions. But at the same time each single operation, he executes, may have repercussions at any point in the grid. The consequences may be of no further importance, but can result in violations of the N-1 security or even in a loss of power at some point and in the worst case of cascading failures at several points.

Problems cause stress

In case of a power loss or just a limit violation, which may cause a power loss, the operator's job gets stressful. He has to get everything running as quick as possible and to avoid further failures. His goal is to get the system stable again. At the same time he probably has to deal with ringing phones, complains by the customer or his anxious supervisors.

No incident counts as success

For the operator it is a success to maintain the grid without incidents. He makes changes in the grid only if absolutely necessary. But as the energy market is changing, economical reasons for routing become more important.

No right solution

There is not one right solution to solve a problem or stabilize the grid. There are actions, which can be taken and should improve the situation. But there are also actions, which can make the situation worse and in the worst case cause a black-out.

Every operations needs to be simulated

Except for economical reasons, there are two reasons for executing an operation: Maintenance and problem correction. Some operations are scheduled for maintenance or other reasons. Other operations occur during the shift and are needed to correct problems and to keep a secure state. Regardless if planned or not, operations have to be simulated and verified directly before execution. Planned operations are simulated additionally during one or several planning phases.

Decisions by intuitive reasoning

To act quick enough, operators often decide on intuitive reasoning, especially in cases of insufficient data or missing time for all analytical procedures. Operators normally work in a control center for a long period of time and gain a lot of experience. They know their grid by heart, and know, for example, how the grid would react to changes or events, or if an element can take more load as the official limit. If a helping, but not essential information is too complicate to access, they trust their intuition and do not use the additional information.

There are manuals with suggestion for operation procedures to handle problems. The grid is highly complex, and it is not possible to foresee all situations in full detail. The manual only assists the operator in finding a suitable action. Rather most operators trust their experience and recall similar situations from the past to make a decision [von Meier, 2006].

Predefined procedures for problems, but not for every situation

The main tools, an operator works with, are the operating system controlling the grid, the telephone for internal and external communication and (printed) documents, like schedules, manuals or reports. The computer system will be described in more detail in the next section.

2.3 Current Operating Systems

Each transmission grid is individual, and no two of them are similar to each other. Different geographical conditions, different elements and techniques constitute the individuality of a grid. The operating system of a power grid has to be adjusted to the corresponding grid and technology. Thus, there does not exist a universal system. But there are common characteristics and problems, which will be described in the following section.

The operating system is used by different people with different functions at several work places. As mentioned before all of them use the same interface, which comprises two modes: real time and study. As the name suggests, the real time mode monitors and controls the grid. In the study mode the grid is simulated and it is used to plan and create schedules for the future, e.g., next hours, days, or weeks.

One user interface for all functions

The grid is monitored and controlled by several smaller systems and not only by one. It consists of many tools and applications. As everything is combined and used with one user interface, it is sufficient to consider a single system in the context of this thesis.

Several small systems are combined to one

The combination of different systems to one, causes an order of information by domains, applications, tools, and functions.

One-line diagram

One important representation tool is the one-line diagram, a simplified notation of the technical grid structure. It displays in which way the different components, like lines, generators, and busses are connected, without accounting for geographical positions or distances. Data and states of the components are represented here as either values, icons or colors. The kind of data and the size of the section presented depends on the purpose of the one-line diagram and by which application it was launched (see figure 2.4).

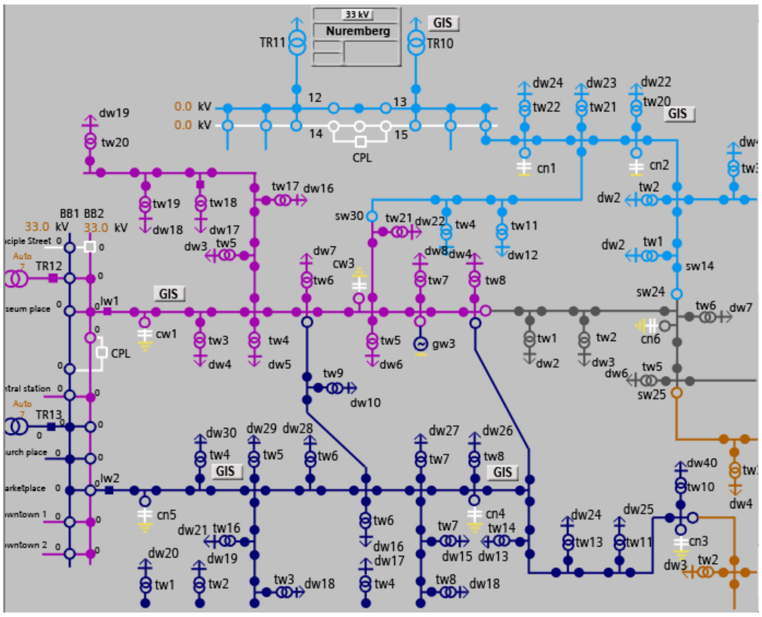

Figure 2.4: An one-line diagram

2.4 Problems while Using the Interface

As with every other interface, problems caused by the interface may occur, while working in a control center. To learn more about their nature I took part in a demonstration of an alarm and its clearance. I used this demonstration to identify the typical problems with the interface of an operating system.

In figure 2.5 you can see an abstraction of the applications used during this task and the steps taken within these applications respectively between different applications.

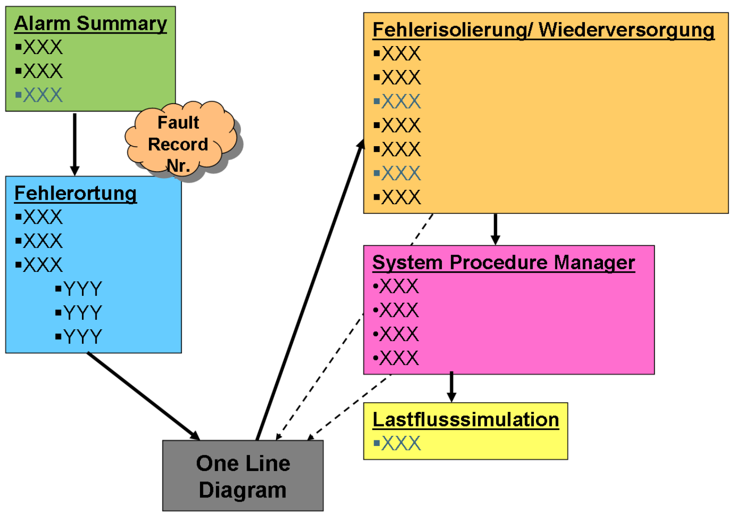

Figure 2.5: Abstract view of the applications used during a typical task in a control center

Every box in the figure represents an application, which has to be used and each of them is opened in a new window. The name of the application is written in the box and was kept unchanged. The solid arrow shows that one application is started out of the other. The broken arrows represent the need to look at a previous used application, which is probably still open. Each three Xs stands for a step, which has to be done in that window, like pressing a button, or changing the view. The Xs written in blue denote, that the operator has to make a decision, remember something or check some data. The cloud between the green and the blue box represents a number, which has to be looked up in the first application and searched in a list in the second application.

In the following, I will identify problems with the user interface and give examples of their occurrences. Some of the problems are not identified in the demonstration, but in the reports of the site visits.

Problem: Memory in the head

- The memory load for a user should be kept as small as possible. Knowledge is better 'saved' in the world instead of in the head.

  In the demonstration task a number had to be remembered in one application and chosen in a list in a different application. Concentrating on that number takes the focus of attention from the task at hand and the operator has to refocus to the task. In addition the operator needs time to find the right number. The computer probably could do so faster.

Problem: Multiple applications

- A high number of running applications, each with its own window, makes it difficult to find the desired one.

  At the end of the task, there are six windows opened, and the operator has to search for the right window, if he needs to go back to a previous used application. No obvious associations between the different applications help to orientate.

Problem: Different views in one application

- Different views in one application can confuse the user. He has to remember to change the view and could be confused by a different look than expected.

  Even though the operator has chosen the right application, he still needs to find the view with the information or action he needs. In the window 'Fehlerortung' he has to change to another overview right after starting the application.

Problem: Information overflow

- The structure of information by domains, and applications causes an information overflow. The complexity of many applications, views, and information demands to much attention, which instead could be used for problem solving.

  The information presented is typical for the application or domain, to which the application belongs, but not adapted to the task at hand. Thus the operator has to search the needed information and filter it mentally to get the desired one. This was the case for nearly all applications, used during the demonstration.

Problem: Scattered information

- Scattered information, outside of the system, cause breaks in tasks to access this information. There is helpful information, that is currently not integrated in the system, and therefore, in some cases not used.

The prepared procedures with suggestions, how to solve the problem mentioned in 2.2—"Working Context in a Control Center", are often kept as printed documents. There are some more documents helpful in decision making, like reports about older cases, data about neighboring systems, or manuals.

- Giving essential information at later point, forces backward steps.

  In the 'Fehlerisolierung/Wiederversorgung' (orange box) a suggestion is chosen by the operator and passed to the 'System Procedure Manager' (pink box) for further examination. Afterwards it is passed to the 'Lastflusssimulation' (yellow box), where it is checked for limit violations. In case of violations another suggestion has to be chosen in 'Fehlerisolierung/Wiederversorgung' (orange box) and the process has to be repeated with the new suggestion.

Problem: Backward steps

Summarized the operator needs to concentrate much on navigating to, searching, and filtering information. A part of his actions is only used to find the desired information, instead of concentrating on the task itself. Different types of breaks in his workflow slow him down in his work

2.5 Example Task: Contingency Analysis

As the entire interface of a control center is way too complex to be considered completely in this thesis, I chose the contingency analysis as example task.

The contingency analysis calculates for each device of the network, what would happen, if the device fails. It identifies violations at other parts of the networks caused by a failure of the device and reports the violations. As the network should be running with a N-1 security, there should not be any violations.

Contingency analysis calculates consequences of possible failures

Consequences should be kept low

If there are any critical contingencies—causing violations—it is the operator's task to solve them. He analyzes the state of the grid under the conditions of the contingency and the surrounding conditions, e.g., time and weather.

Preventive or corrective action

He has two possibilities to solve the task, depending on several different aspects, like the severity of the violations, the time to react after the failure and the probability of that failure: A preventive action or a reactive plan. The preventive actions are immediate changes in the grid, so that the failure of the initial device won't cause any violations or additional failures. The reactive plans are executed after the initial device failed. They comprise procedures, which avoid violations and additional failures.

Periodic execution

The contingency analysis is started automatically, triggered after a defined time period or after a switching operation in the grid. If there are new cases with violations a visual alarm occurs, so that the operator is directly informed about potential problems.

The contingency analysis is used in real time operating mode and in study mode. Both modes are similar for the contingency analysis. I concentrated on the real time mode for this thesis.

The choice of the contingency analysis and its handling as example task had several reasons. It is one of the most frequently performed task during an operator's work day. The calculations take place on a regular basis, and the operator has to solve the critical contingencies as soon as they occur, which happens several times during a day.

A lot of different types of interaction with the interface or even with additional tools, like documents, telephone are part of the contingency analysis. Thus, the main part of interactions in a control center are covered.

2.5.1 Task Analysis

In interface design it is important to know more about the task, for which the interface is designed. A task analysis helps to understand the way people are doing their job and contributes in finding the requirements for a new interface [Dix et al., 1997].

A typical task analysis considers also aspects outside of the system. As mentioned in chapter 2.2—"Working Context in a Control Center" there are some important facilities outside the system. Thus the task analysis includes the interaction with this facilities and later on it is possible to consider their integration in the new design.

Decomposition into subtasks

One possibility to conduct a task analysis is to decompose the task into subtasks. For the task 'contingency analysis' I conducted a task decomposition on a low level to get a first structure (see figure 2.6).

The flow of the task is as follows: The operator starts with gaining an overview on the new critical contingencies and choosing one to solve. Most of the times, he chooses the worst one, which is indicated by a performance index. The parameters used to calculate the performance index depends on the company, responsible for the grid, and can be influenced, for example, by the severity of the violations, the number of violations, or the importance of the affected elements.

For the chosen case he analyzes the state of the grid, especially of the violated elements, under the assumption of the failed element(s). If he has knowledge about surrounding conditions, like the weather or the environment, he takes that information into account, as it may have an influence on the grid's behavior (see chapter 2.2—"Working Context in a Control Center"'). Knowledge about earlier failures of the contingent element can provide information about the probability of failing and the reactions, of the grid as well as the counteraction of the operator on duty.

On the basis of situational analysis the operator can check for predefined procedures, which propose actions to solve the problems. The possibilities for actions also depends on the responsible company. Possible actions are changes in the grid, e.g., activating a line, or changes at power plants, e.g., energy feed-in. A predefined procedure has to be adjusted to the current situation. The operator can, instead of

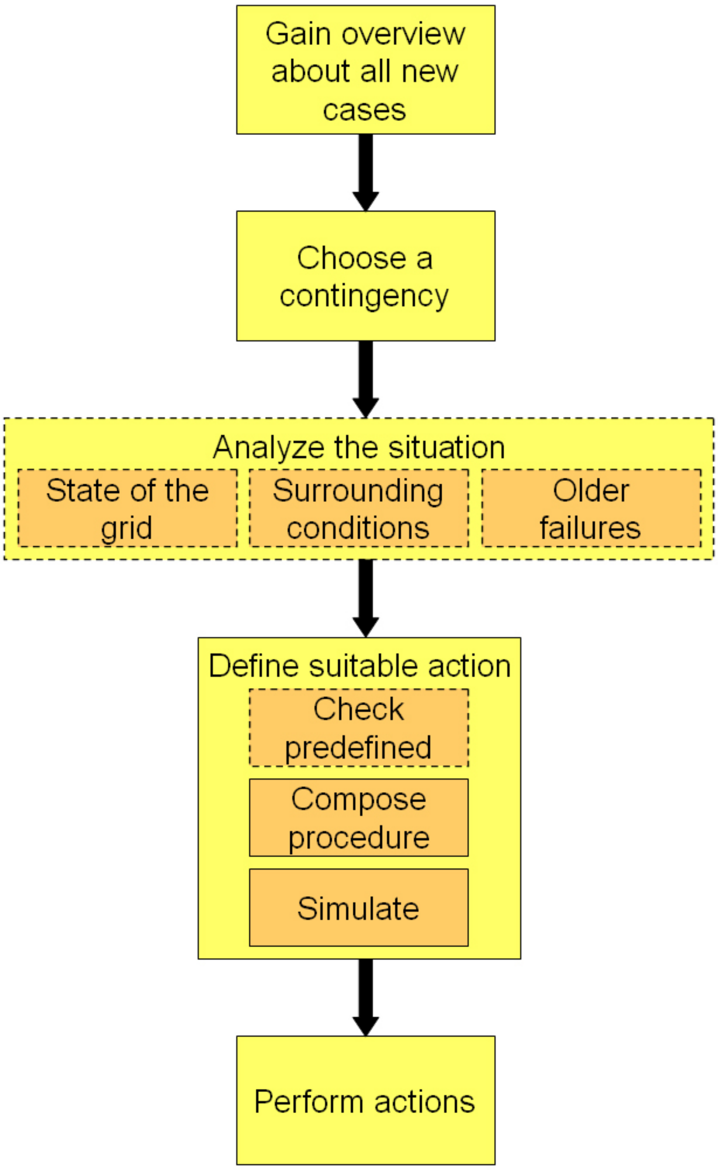

Figure 2.6: Task analysis of the contingency analysis and its handling; The boxes represent the subtasks, the orange boxes are subtasks of the yellow ones. Horizontally ordered tasks can be performed in any order, while vertically ordered have to be performed in this order. Tasks, surrounded by a broken line, need not to be executed.

using a predefined procedure, define one himself. Whatever action he chooses, he has to simulate the state of the network under consideration of the planned actions before executing them.

The new design should guide the operator through the process of choosing a case to work with and finding a solution to the problem. But it should only support the operator with his decision and should not give a final decision.

Chapter 3

Theory

After introducing the theory of the domain power transmission in the previous chapter, the following chapter will give the theoretical background of mapping and of zooming user interfaces.

Two main ideas for the new user interface are the concepts of mapping and zooming user interfaces. Therefore, I will first give an overview about mapping, more precisely about mind maps and concept maps and their psychological background. Afterwards, I will introduce the idea of zooming user interfaces and different attributes to characterize them.

3.1 Mapping

The term 'mapping' is used in different domains with different meanings. In this thesis I relate to mapping as a method to visualize information with its structure and context. It is based on a theory of cognitive psychology.

Associational organization in the head

Cognitive psychologists suppose that information is 'stored' in the human brain like a semantic network [Dix et al., 1997]. New information is integrated in this network and relationships to existing information are built. Every human constructs her individual network with individual associations. Associations are similarities of various types between two or more items or thoughts. To recall information from the brain the associations are used and the related information is remembered. For example, Garfield's name may be remembered by thinking about a cat and a comic figure, or while eating some lasagna, his favorite food. It is assumed, that a presentation of facts as a visual network supports understanding and remembering these facts [Mandl and Fischer, 2001a].

Many different types of mappings and methods to create maps have been developed, e.g., mind maps, concept maps, semantic net, or knowledge maps. For my thesis I concentrated on the most common approaches: mind map and concept map.

Similarities: nodes, links, spatial layout

Both are developed in pedagogical contexts to support the process of understanding. Common to both types of maps is the symbolic infrastructure of nodes and links and the organizational principle, which is a spatial layout [Dansereau, 2005]. Both techniques are used to represent knowledge in a visual form [Goldstein, 2001].

3.1.1 Mind Maps

Spidermaps as parents of mind maps

Even if Mind Maps are claimed to be invented by Tony Buzan [2002], the first step in the development was done by Marilyn B. Hanf [1971]. She developed a map to support students' understanding of texts, which was later called *spidermaps*.

Creation of a spidermap

Instead of making linear notes, while reading a text, the students should create a map to rethink and structure the content of the text. Therefore the main topic is written in the center of a paper and the succeeding thoughts are added around the main topic. Thoughts should be written with as few words as possible and organized hierarchically. Thus the students' critical thinking should be fostered and the maps are easier to repeat. See figure 3.1 for an example of a spidermap.

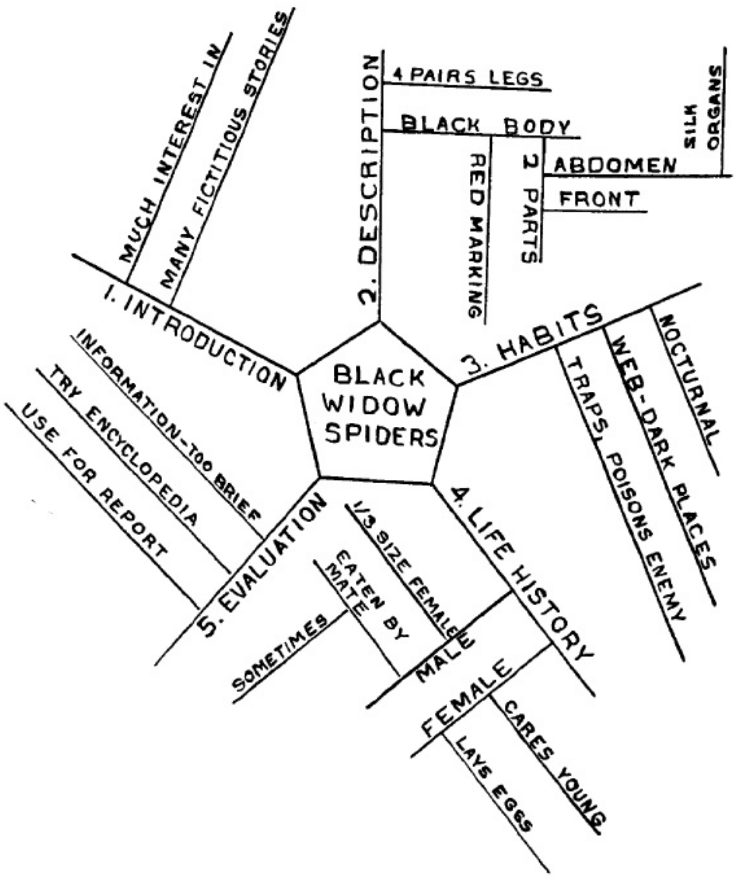

Figure 3.1: Spidermap

Tony Buzan: inventor of mind maps

Tony Buzan started with a technique called *pattern noting* or *brain pattern*, which he later enhanced to *mind mapping* [Buzan and Buzan, 2002]. First developed for educational purposes, today mind maps are used in different contexts, e.g., schools, universities, business companies, or even in private context. Tony Buzan suggests to use mind maps for several tasks like brainstorming, decision making, structuring ideas, creativity, or learning.

Mind maps activate more cortical abilities

As Buzan first used mind maps as note taking tools he compared it to other note taking methods. He criticized, that only a few cortical abilities were used in standard note taking. Especially, visual clues, colors, spatial sense, or associations were not used, although their use could enhance the creators or readers attention.

Use associations

In addition, he claimed that the use of association is similar to the way of human thinking (see chapter 3.1—"Mapping"). Thus, mind maps are simplified visual representation of thoughts and should support learning and understanding.

A mind map has four main characteristics [Buzan and Buzan, 2002]:

Subject in the center

1. The subject of the map is placed in the center of the map. It could be written as text, presented with a picture, or a combination of text and picture.

Thoughts radially arranged

2. The main thoughts are arranged radially around that main topic—like branches.

Keywords and -pictures

3. Keywords and -pictures are written on the hierarchically arranged branches.

Linked nodes

4. All branches build a complete structure of linked nodes.

In figure 3.2 a mind map with the main topic 'Mind Map Guidelines' is shown. Even though Buzan prefers mind maps painted by hand this computer painted one is a good example as it includes all characteristics.

Pictures augment memorability

Even if the main concept of mind maps is similar to spidermaps there are some important differences, which make them more powerful. One big difference was already mentioned above: the use of pictures. Buzan highly recommends to use pictures and colors, when creating mind maps, because pictures are better remembered than pure text.

Colors and dimension too

For the same reason Buzan suggests to make a high use of colors and dimension. A creative and individual layout augments the memorability of the map and its content by appealing different brain regions.

One line per thought

For simplicity the user should reduce the thoughts to the minimum at best one keyword, -picture or a combination of both. Each thought should be written on a single line with the exact length of the keyword or -picture.

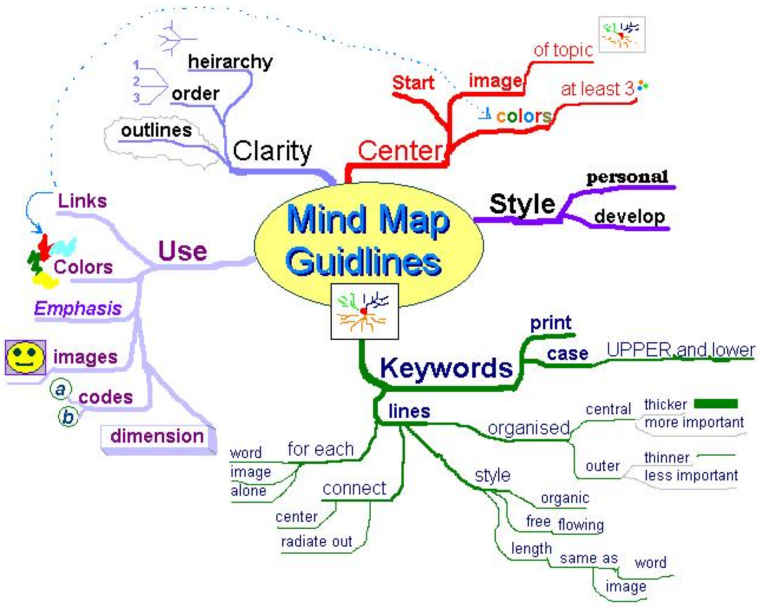

Figure 3.2: Mind map on the subject: 'Mind map guidelines'

The main structure of a mind map is a hierarchy, but in difference to the Spidermap cross-links are permitted and recommended. The hierarchy should be clearly demonstrated by using thicker lines or bigger letters.

Hierarchy with cross-links

To make high use of the power of the brain, Buzan recommends some general techniques to use in a mind map. To ease the recall of thoughts, the main characteristics of memory should be used. As stated above one of these characteristics is association. In mind maps the branches are automatically associations. The user benefits of the basic structure of mind maps as it is similar to the natural structure. In addition she can add cross-links, arrows, or color to augment the number of associations.

Associations support memory

The second characteristic of memory is emphasizing. Important and high-level thoughts should be highlighted, so that they are easier seen and remembered. Size of pictures, letters, or lines is one possibility to emphasize important aspects. But also color, use of pictures in general,spatial

Emphasize important thoughts

layout, and even free space are ways to emphasize certain aspects.

Altogether the design of a mind map should support the user to perceive and filter information according to a special subject, so that he could concentrate on the content.

3.1.2 Concept Maps

Also in the seventies Novak developed *concept maps* [Novak and Canas, 2006]. Like mind maps concept maps have been developed in a psychological and pedagogical context. It is based on Ausubels assimilation theory, which describes learning as connecting some new information with existing knowledge. New information is integrated and connected to knowledge, which is already in the head.

First developed for children

Novak, a researcher of human learning, educational studies, and knowledge creation and representation, examined the understanding of children. With the requirement to make the children's knowledge and understanding visible he developed concept maps.

Concepts, relationships and propositions

A concept map is a network to represent knowledge consisting of concepts and their relationships. A concept consist of words, symbols, or a combination of both. Two concepts and their relationship are called proposition.

Boxes, circles, lines and arrows

In the graphical presentation concepts are framed by boxes or circles, and relationships are demonstrated by lines or arrows, which are mostly labeled. The basic structure of a concept map is hierarchical. The most global concept is placed at the top of the map, and the more detailed the concepts are, the lower they are arranged. The hierarchy is not strict and cross-links augment the interconnections between the concepts and give more possibilities to express ideas in a concept map. See figure 3.3 for an example of a concept map.

Focus question

For easier creation of a concept map Novak and Cañas [2006] suggest to use a focus question, which should be answered by the map. Thus, the building process of the map is supported, and the map helps to answer the question or solve a problem.

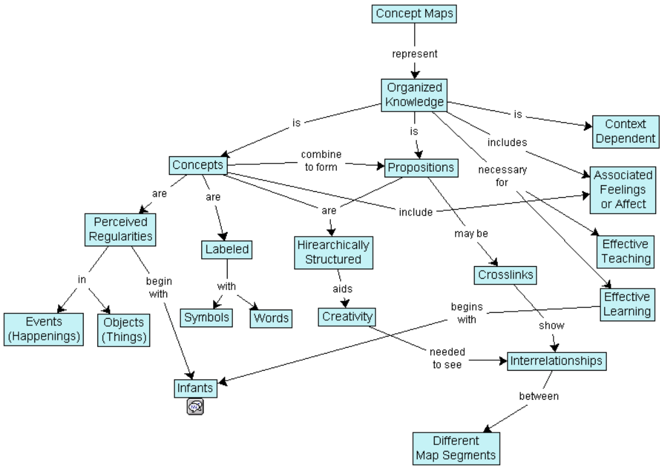

Figure 3.3: Concept map on the subject: ‘Concept Map’

There are two main differences between mind maps and concept maps, except the obvious visual differences. While a mind map has one main thought a concept map can contain several main concepts, even if it answers only one focus question. Thus the structure of a mind map is like a tree structure, while a concept map is more comparable to a network.

One main thought vs. several main thoughts

Different authors already suggested to use mind maps or concept maps to support navigation in information spaces like web sites [Canas et al., 2005, Alpert, 2005]. See also chapter 4.3—“Organizing tools” for some examples.

3.2 Zooming User Interfaces

Zooming user interfaces (ZUI) are used to cope with the problem of presenting a big information space on a limited screen space. An infinitely large plane is used to present all information by organizing in space and scale [Hornbæk et al., 2002].

Infinitely large plane

Interaction by panning and zooming

To get access to all information the user has two ways to interact with the information space: panning and zooming. The impression of physical movements should ease the interaction with the plane. Good and Bederson [2001] compare the action of panning with moving a piece of paper and zooming with looking closer or further at it.

FisheyeView, BiFocal Lens

Early techniques of zooming are the Fisheye View [Furnas, 1986] and the BiFocal Lens [Spence and Apperley, 1999]. Either of them showing the information in focus zoomed in and detailed and the information less important—the context—zoomed out and more abstract.

Degree of interest (DOI)

The degree of interest (DOI) determines in which detail objects are presented. For objects in the focus the DOI is high and for objects in the context it is low. Both techniques use distortion for the transition between focused and context information. The balance between local detail and global context has to be kept [Furnas, 1986]. The context should help to find and process the information the user is interested in.

Pad/ Pad++

One of the first zooming user interfaces was Pad [Perlin and Fox, 1993], which was further developed to Pad++ [Bederson and Hollan, 1994]. It is an indefinitely large information plane, where the objects are organized spatially. Portals are used like magnifying glasses to zoom to documents, images etc. The level of detail, at which objects are shown depends on their current size. For example, a book is represented only with its title. Zooming in to the next level, first, shows the abstract and then the table of contents, till reaching the point, when the user is able read the text of the book.

Zoomworld/ Archy

Another zooming user interface was developed by Jef Raskin [2000]. He even introduced the zooming interface paradigm (ZIP) for the idea of working on a infinitely large plane by zooming and panning . While Pad(++) was really implemented, Raskin's Zoomworld has first been only thoughts, which were partly implemented later on in Archy (see Raskin Center[1]). Zoomworld consists also of an infinite information plane with the possibility to zoom in

[1] http://rchi.raskincenter.org

and out, but with no need of portals or something similar. Raskin also compares the panning and zooming to real world movements by using the expressions 'flying over' and 'diving in'.

Similar to real world navigation

Both, Pad(++) and Zoomworld, are based on the same idea. They are using spatial layout to support the user orientation and navigation. In real world people find routes or things by orientating with the help of relative positions and landmarks.

On a noticeboard full of notes, papers, and pictures a typical sentence would be: "the document I need is next to the picture XY". Looking at the board from a distance allows to find desired objects by an information like the cited sentence. Moving closer to the board gives access to the detailed information, as the observer is able to read it now. This approach is transferred to ZUIs. In the context view the abstracted shapes can help to find something, while in the detailed view all information are perceivable.

Geometric zooming

The appearance of the objects in the information space depends on their level of scale [Hornbæk et al., 2002]. But the difference depends on the kind of zooming. For geometric zooming the only difference is the size of an object in different level of scales. The zooming is linear and the size of the objects is proportional to the zooming distance. Geometric zooming is used in CounterPoint and Fly presented in chapter 4.2—"Presentation Tools".

Semantic zooming

In semantic zooming the objects are shown with a different level of detail. Zoomed out only the most important facts are shown. While zooming in, the amount of visble information increases. Semantic zooming is often used in maps—the closer you look at it the more details are shown. For another example see chapter 4.4—"Semantic Zooming".

Constant density

A third way of zooming is called constant density. The scale depends on the number of objects currently visible. During zooming the number of objects remains constant, but the visible objects and their appearance changes [Woodruff et al., 1998a].

Goal-directed zooming

The panning and zooming is usually controlled by input devices like mouse or keyboard, so that it is linear to a change in these devices. But also non-linear control is possible. Hornbæk et al. [2002] mention three forms of non-linear zooming. In goal-directed zooming the user choose an object and the representation. The interface zooms and pans accordingly [Woodruff et al., 1998b].

Combination of panning and Zooming

The second form of non-linear zooming is a combination of panning and zooming. That means extensively panning automatically causes zooming [Igarashi and Hinckley, 2000].

Automatic Zooming

Using automatic zooming, a click on the object causes an automatic zooming and panning, so that the object is centered. In this form of zooming the zoom level is predefined for the objects. This form is used for the the presentation tools in chapter 4.2—"Presentation Tools".

Jump zooming

Another difference in the use of zooming is the transition between different levels. On the one side there is jump zooming, where the representation in one level is instantly replaced by the representation of the new level of zooming.

Animated Zooming

The opposite is a smooth transition between two zooming levels, which is called animated zooming [Hornbæk et al., 2002].

Chapter 4

Related Work

The related work, which I will present in this chapter, can be grouped in three categories. Existing tools for contingency analysis demonstrate current work of visualization techniques in the domain of power transmission. Associative organizational tools illustrate the use of associations to find information. Zoomable and mapping interfaces demonstrate the interaction techniques I will employ in my system.

4.1 Contingency Analysis Visualization

Three-dimensional, interactive

With such a huge amount of information in a power system, some research about visualization is done. One work is an interactive three-dimensional visualization for the results of a contingency analysis by Sun and Overbye [2004]. The contingency analysis calculates the consequences of a possible failure of each element in the power grid (see chapter 2.5—"Example Task: Contingency Analysis"). They wanted to connect the information about contingencies and their violations, which is mostly presented in tables or matrices, with the technical position. Therefore, they integrated the information visually in a third axis of a one-line diagram.

Two different views

As there are two concepts of looking at the contingency analysis, they decided to separate these two concepts into two different views. One of them emphasizes the contingencies and their severity and the other one the elements in the power system, which would be affected by contingencies.

Overview with all cases

For both views there are three levels of detail. The overview displays the system security state and presents either all contingent elements causing a violation as cylinders or all violated elements as reversed cones (see figure 4.1).

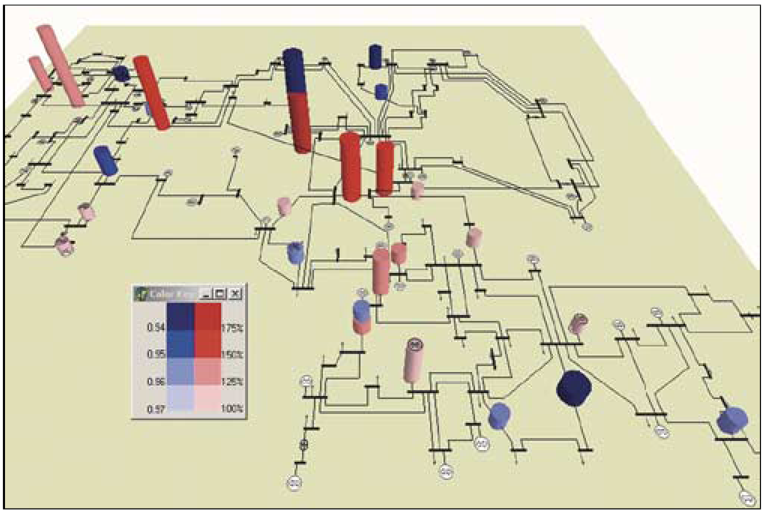

Figure 4.1: Visualization of the contingency severity: The xy-plane is a one-line diagram. The cylinders present the contingent elements, where blue marks low voltage violation and red transmission violations. The height of a cylinder is proportional to the number of violations, and the radius and brightness is proportional to the magnitude of the worst violation.

Colors indicate types

The authors use different visual clues to give more information about the marked elements. Different types of violations are distinguished by color: red marks transmission violations (overloads), and blue marks low voltage violations.

Brightness and size for severity

Severity of the worst violation is indicated by the markers' radius and brightness. This double-coding aims at balancing distortion effects caused by the perspective in the 3D-representation. Objects, which are further away, seem to be thinner than objects in the front, even if they have the same size.

Few cases with more information

The middle detailed view reduces the amount of information to a handful of elements—again either the contingent or violated ones. As additional information a line demonstrates the connection to the respectively other, i.e., the resulting or evocating element, depending on the choice of presentation (see figure 4.2).

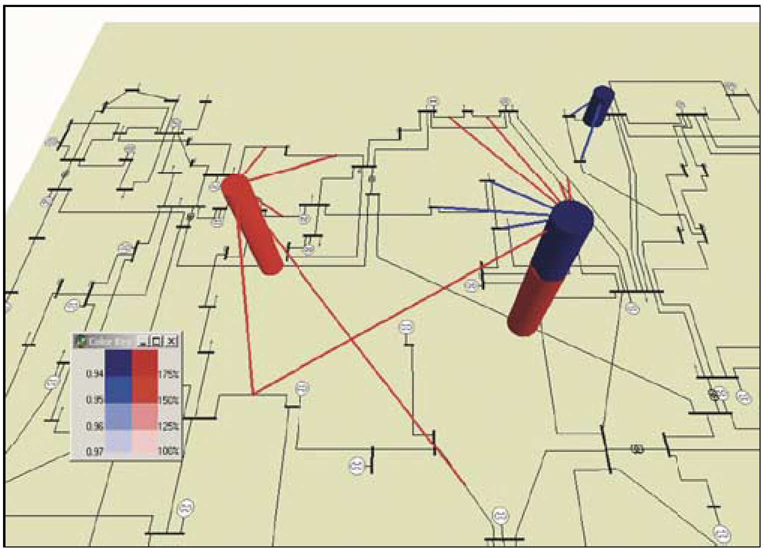

Figure 4.2: Middle-level visualization of the severity of a contingency: A few contingencies are presented similar to figure 4.1 plus lines pointing to the violated elements

One Element

The third view is the most detailed one, which takes only one contingent or one violated element into consideration. As additional information the location and magnitude of the violated elements are visualized.

As the authors' focus lay solely on the visualization, the interaction with the system is not described any further.

The authors themselves state, that this approach to visualize the contingency analysis should not replace the numerical charts used at the moment, but could supplement the

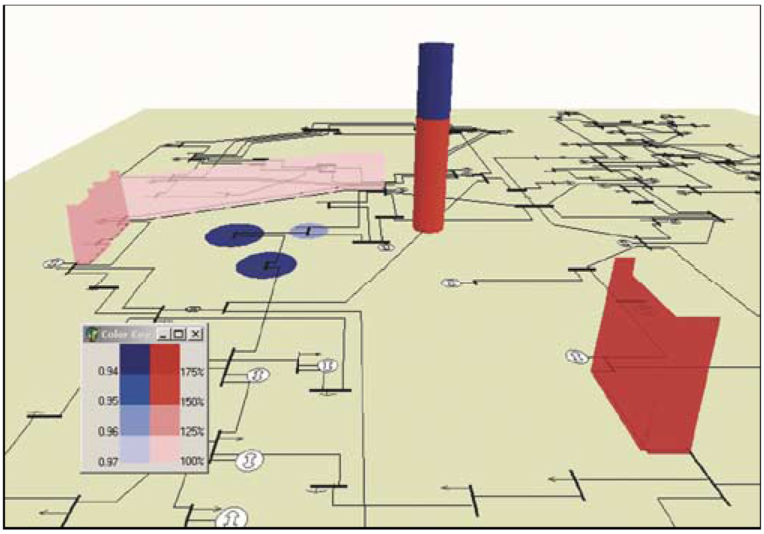

Figure 4.3: Detailed visualization of the contingency severity: Violated transmission lines are extended into the z-axis. Violated buses with low voltage are marked with blue circles. The size, either radius or height, and shading represent the severity

information. Conventional representations are still needed to give concrete values.

4.2 Presentation Tools

After describing an existing approach to visualize the results of contingency analysis, I will now describe several projects using technologies, which I will integrate into my design.

Zooming presentation tools

I will introduce two presentation tools with zooming techniques. CounterPoint as well as Fly use spatial arrangement to support the audience's orientation and understanding similar to mind maps and concept maps.

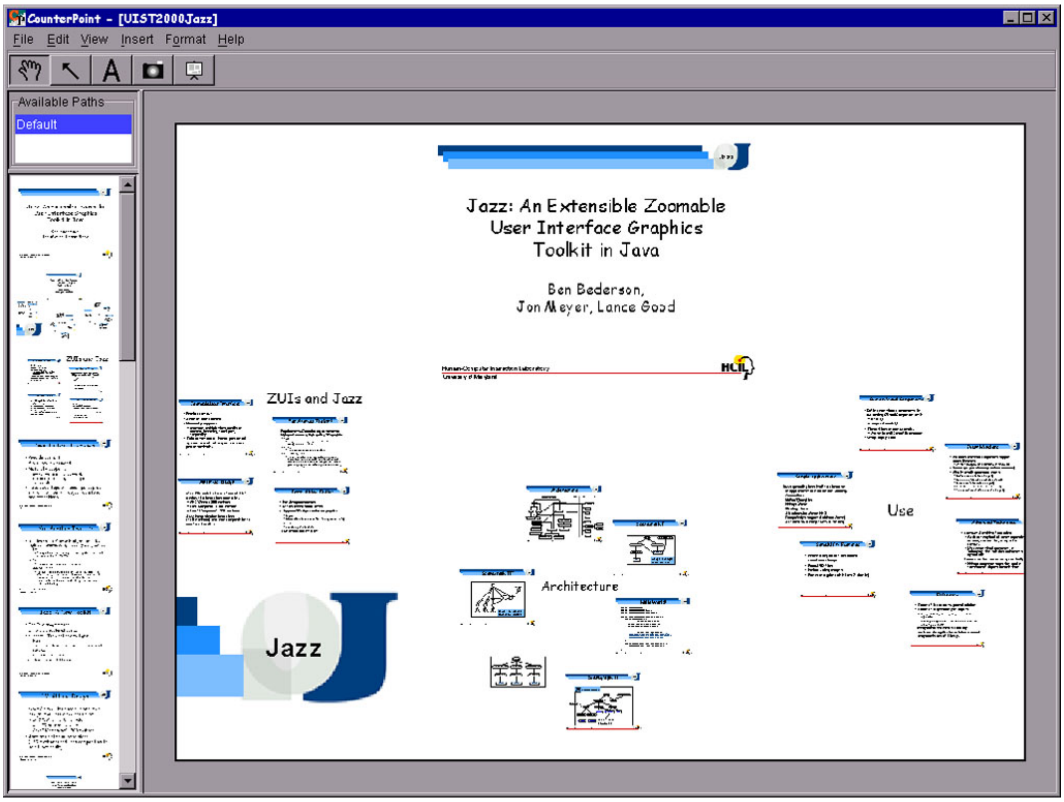

Figure 4.4: Screenshot of CounterPoint

4.2.1 CounterPoint

CounterPoint was developed by Good and Bederson [2001, 2002]. The user can import slides, create a spatial layout and define a path for her presentation.

Spatial layout and presentation pathes

When creating the spatial layout, the user can position the slides freely, group associated slides together, and define the size of the slides in the overview (See figure 4.4).

Flexible layout

Good and Bederson identified two kinds of users: the presenter and the audience. For both groups CounterPoint offers advantages.

Two user groups: presenter and audience

The presenter is flexible in her presentation as she can reuse the same slides for a presentation with a different purpose by defining another path, e.g., a shorter one or in a different

Reuse of presentation, flexibility during presentation sequence. During the presentation, it may be necessary to spontaneous revisit slides to answer questions, or to skip some slides caused by a lack of time. It is easier to choose an arbitrary slide in the spatial layout than in normal presentation. In the overview she can choose a slide by seeing it and clicking on it, while in a linear presentation she has to flip through all slides between the current one and the desired one.

Spatial layout supports memory

For this thesis the advantages for the audience are more interesting. The spatial encoding should helps the listener to remember and recall the content of the presentation. In addition, the structure of the presentation can be encoded in the spatial layout, so that the user learns the structure implicitly during the presentation. This approach follows he principles mentioned in chapter 3.1—"Mapping".

Animated zooming against disorientation

To avoid disorientation, the transitions between slides are animated. This way, the listener sees the context of both slides. The animation of zooming out to an overview and zooming in to a detailed view is a natural behavior, which every person uses in real life, e.g., in front of a noticeboard (see chapter 3.2—"Zooming User Interfaces").

4.2.2 Fly

Fly by Holman et al. [2006] is also based on spatial arrangement and zooming in and out of slides. There are however some remarkable differences.

Based on mind maps

In Fly the slide layout is based on mind maps with the main subject in the center and the subsequent levels of ideas located towards the exterior connected by links (see figure 4.5).

Semi-automatic layout

Fly's import process is augmented by semi-automatic spatial arrangement of the imported slides. The user inserts a slide on a node and Fly integrates it as a child node, positions the new slide, and connects both nodes.

Animated zooming

Transitions of slides during a presentation is composed of zooming out of the current slide, panning to, and zooming

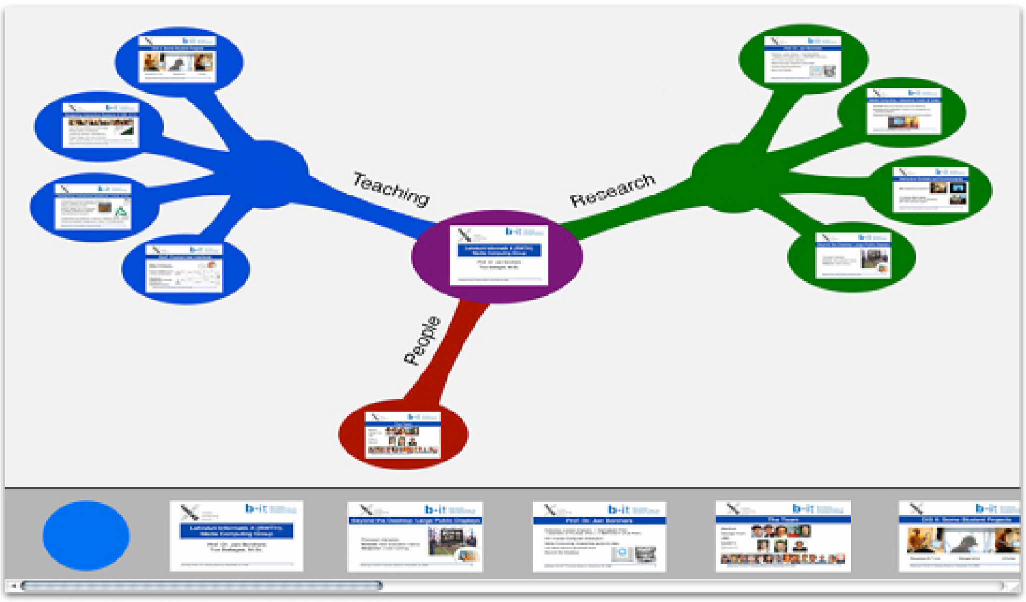

Figure 4.5: Screenshot of Fly

in the next one. Panning and zooming take place at the same time. Slides are visited in a clockwise order beginning from the center node. When entering a subgroup of nodes the animation follows the line from the parent node to the first child (see figure 4.6).

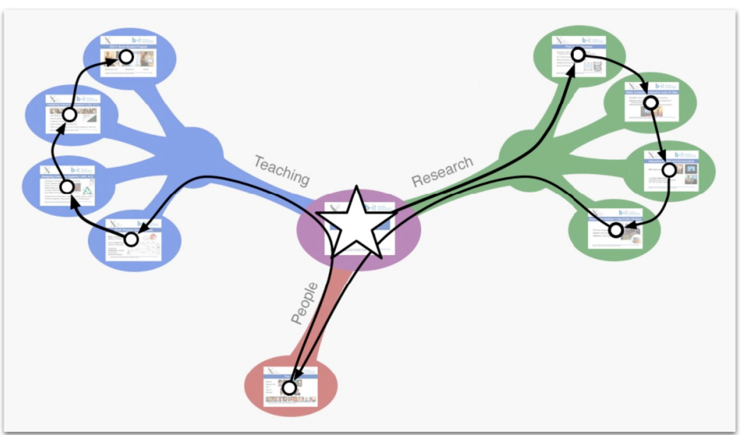

Figure 4.6: Order of slide transition in Fly. Beginning at the star, the slides are traversed in the order of the arrows.

Stronger order

When creating a presentation the user can choose between the semi-automatic order or a free order. Random access to slides, e.g. to answer questions, is possible by clicking on the slide, which is then panned to and zoomed in.

Links increase visibility of the structure

For the audience the advantages are similar. The visual structure of the presentation is used for navigation and makes relations more explicit.

For the new design the concepts of spatial arrangement and zooming in both tools are interesting. Choosing random slides by clicking on them is intuitive. The automatic zooming supports the user's orientation, and is handled similar in the new design, but it stops before the content fills the screen completely. Like in Fly there will be links, which strengthen the associations between nodes.

4.3 Organizing tools

Associations ar not only used for presentation purposes, but also to support a user in finding information. There are several tools using associations, based on the assumption, that humans recall knowledge with association (see chapter 3.1—"Mapping").

4.3.1 The Big Picture

Web site search by associations

The World Wide Web is a huge information source. You can search the content of some web sites by using associations. Similarities of items are used to get from one of these items to another one.

An example for navigating items by associations is 'The Big Picture' [CNET Networks, Inc and Liveplasma.com, 2005]. News are associated to topics and companies (see figure 4.7). Nodes represent the articles, companies, or topics.

Color and tool tips for meta information

Visual clues are used to add some information to the nodes. Color coding is used to differ the types of nodes. Black

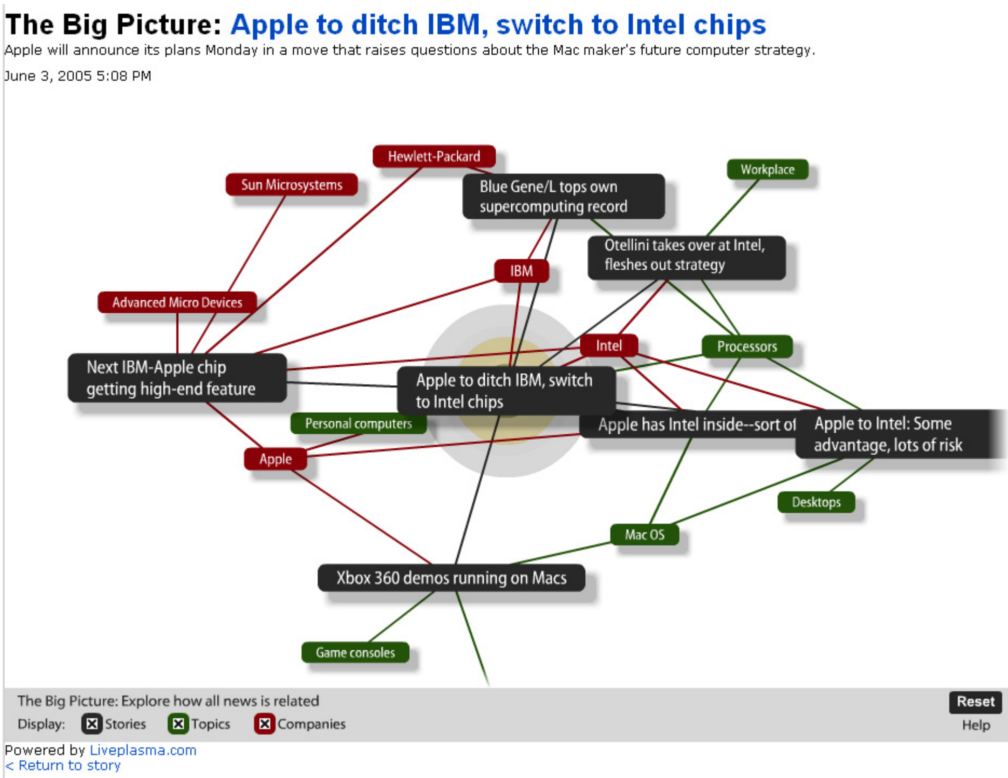

Figure 4.7: The Big Picture: News articles associated by topics and companies

marks articles, red marks companies and green marks topics. Tool tips give some extended information.

Focus is centered

Clicking on a node selects it and centers it. The connected nodes are rearranged according to the centered node. It is easier to find articles with the topic of interest or related to one special company. To help the user keeping the orientation the movements of nodes are animated and can be tracked.

The article could be opened and read with a right-click on the node, representing it.

4.3.2 Personal Brain

Desktop search by associations

Instead of searching content on the web you may need to find something on your own computer. Personal Brain

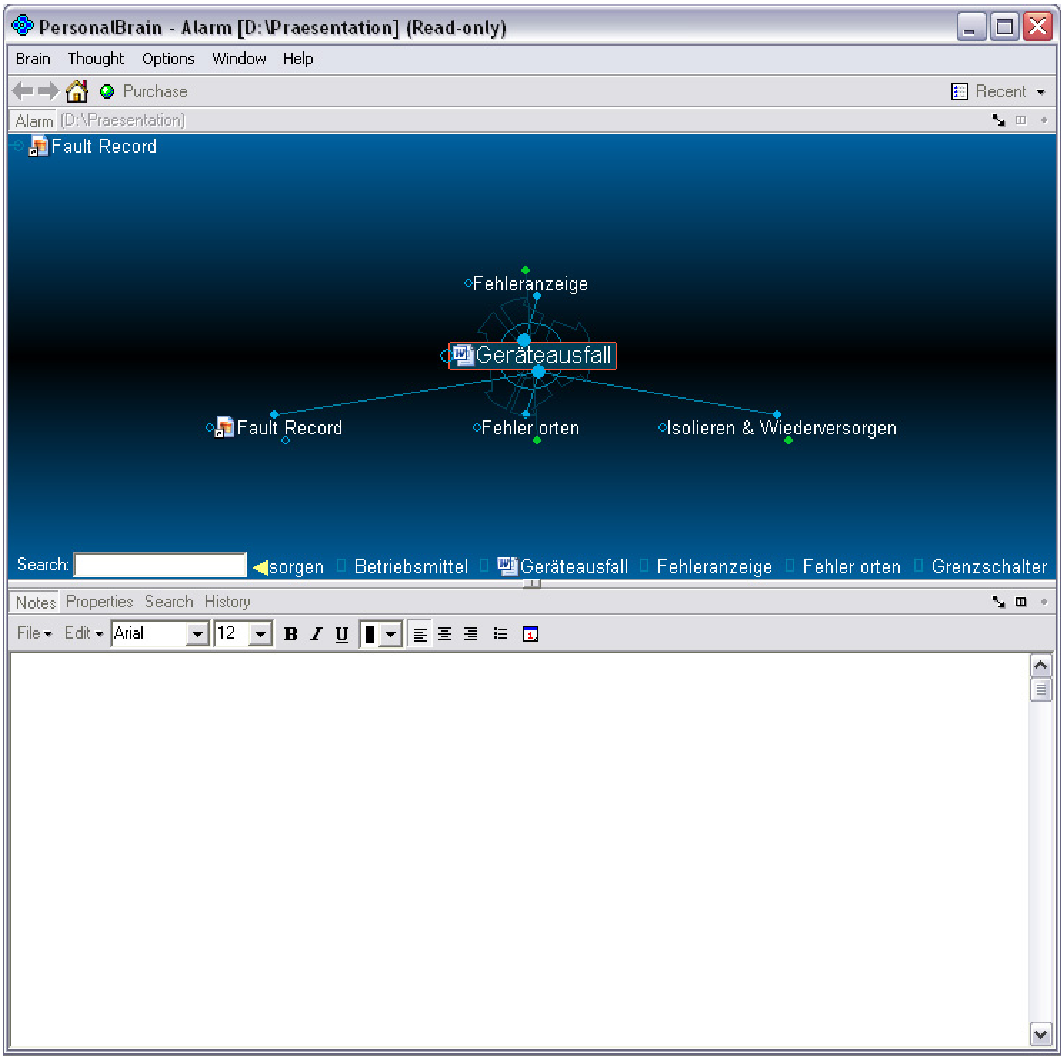

Figure 4.8: Screenshot of Personal Brain

helps you finding information by associations on your computer [TheBrain Technologies LP, 2005]. It does not provide the associations by itself. The user has to construct them. He can connect different types of files, e-mails, or applications or add notes in the tool itself (See figure 4.8).

Hierarchy with cross-links

Personal Brain uses a hierarchical organization with parents, children and siblings. Additional cross-links can be added to 'jump' to an associated node, without hierarchical connection.

With the tool it is possible to create a network representing your own information space. The network is never finished and can always be extended.

Spatial layout

Like in the web tools an item—also called thought—is centered, when clicking on it. The associated thoughts are connected by links and positioned accordingly to the type of their relationship, parents, children, siblings, or cross-links.

Separation of content and structure

The content of the node is not opened at the same time as it is centered. A click on the focused node opens the content in an additional window, the default application of the file. Only if the node corresponds to a note made in Personal Brain, it is shown in the lower half of the Personal Brain window.

Animated transitions

Like in all presented tools the transitions are animated. It prevents the user from loosing the orientation and the context of the new focused item. But with the content of an item opening in another window the connection between the piece of information and the associated items is less obvious.

Additional navigation like a web browser

As Personal Brain has some properties of a hypertext it offers some navigation possibilities known from web browsers. There are buttons to navigate forward or backward and a 'home' button, which leads to the starting thought. Another navigation tool is a history bar, similar to breadcrumbs on a web page. It shows all lately visited items in the visited order.

Similar to these tools my design will allow finding the desired information by following associations. Visual clues, like color or size, provide meta information.

4.4 Semantic Zooming

Most zooming user interfaces like CounterPoint and Fly (see 4.2—"Presentation Tools") scale the objects when zooming in or zooming out. Another possibility is to use semantic zooming and use representations of the same object, with different levels of detail.

Semantic zooming

OZONE [Suh and Bederson, 2001] is a tool for navigating in ontological information, which uses semantic zooming. The classes of the ontology are represented by nodes, and the queries followed to find the information are represented as labeled links (See figure 4.9).

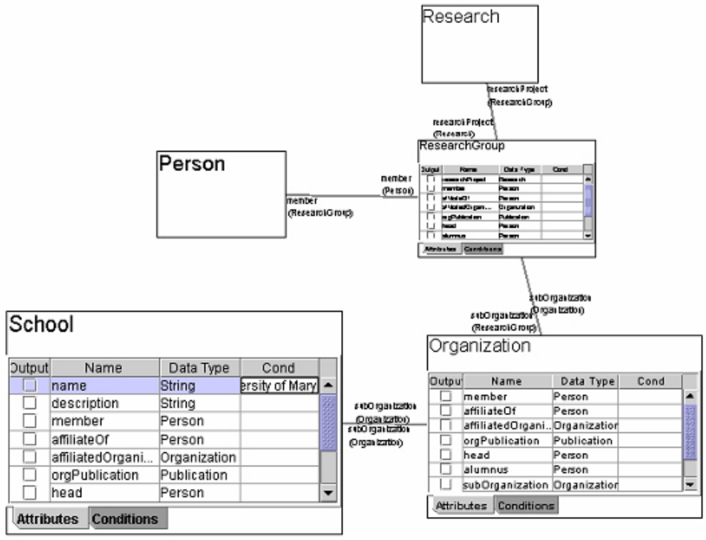

Figure 4.9: OZONE: The focused node 'School' is presented as biggest node with all properties. The next two subsequent nodes are presented smaller, but still with all properties. The nodes 'Person' and 'Research' are presented only with a box and their titles.

Different levels of detail

Similar to the tools presented before, it is possible to center a node and zoom to the detailed presentation. But this tool does not zoom from a scaled version to the full view. There are different representations depending on the actual size of the node. The smallest version of a node consists only of a rectangle with a title. The full view shows a rectangle with the title and all properties of the class.

Group nodes

Creating a query gives the possibility to construct a group node. The group nodes have additional representation. The smallest view is again a rectangle plus title. The full detail view includes showing all nodes belonging to the group node plus their queries.

Constant density

Nodes are zoomed out automatically, if more space for other nodes is needed. Which nodes are zoomed out depends on the connection to the focused node. The further away a node is, the less detailed it is shown. This principle is called constant density zooming (see chapter 3.2—"Zooming User Interfaces").

Animated transitions

The movement of a node to the center and the zooming of nodes are animated.

The semantic zooming, like it is used in OZONE, is a reasonable way to make information accessible even in a view zoomed out. The most important information can be given and the user gets a first impression of the detailed content. For my design it makes sense to use semantic zooming. The distance to the focused node is a natural factor for the size of the associated nodes.

Chapter 5

Low-fidelity Prototypes

Nielsen [1993] proved in case studies, that iterative user interface design improves the usability of a new or a reengineered user interface. As nobody can design an interface without errors in the first attempt, by iterative design the errors could be reduced at each step.

Iterative user interface design

Iterative design is described by the design-implement-analyze(DIA)-cycle. It is a continuous repetition of the three phases. In each iteration the design becomes more precise, the implemented prototype more detailed and the evaluation focuses more and more on details. During the design process big problems are solved first, smaller ones later.

DIA-cycle

During the DIA cycle prototypes are used to implement and analyze the different stages of design. Benyon et al. [2005] state one purpose of building prototypes is the clarification and verification of "requirements, which will usually need adjustment once clients and users have a realistic design to review and explore."

Prototypes for communication between designer and user

The form—paper versus digital—and the grade of detailing of a prototype depend on the time in the design process. At the beginning less detailed prototypes are used to verify basic concepts. The later in the design process, the more detailed and the more similar to the final product the prototypes become.

Type of prototype depends on the intended use

I conducted three phases of the DIA cycle. After a preliminary task analysis (see chapter 2.5.1—"Task Analysis") I developed two paper prototypes to decide between some basic ideas. The results with the paper prototypes inspired me, to build an additional screenshot prototype. At the same time I did more research on the desired information for the contingency analysis and developed an according content structure. After the evaluation of the paper prototypes and the content structure I developed a more advanced software prototype, which will be described in chapter 6—"High-fidelity Prototype".

5.1 Paper Prototypes

Paper prototypes are hand drawn sketches

Paper prototypes are easy and quick to build. They are pencil sketches of the interface, detailed enough to convey the basic concepts [Benyon et al., 2005].

Two approaches to compare

In the first stage two different approaches of the design have been developed. Different alternatives of various aspects were realized in the approaches. The main distinction between both approaches is the combination of content and associations. In one approach the content is visualized with its associations, while in the second approach the content is visualized in a separate part of the window, detached form the associations. Additional other smaller aspects have been realized differently in both approaches.

Avoid impossible design solutions

In an evaluation with experts of the domain and in usability, the different approaches should be rated and checked against constraints in a control center. Impossible design solutions, in the context of a control center, could be avoided in the first place.

Both approaches have been visualized and evaluated with paper prototypes. The evaluation phase was used to decide some design issues.

Main concept: Nodes and associations

The design consist of nodes and associations between them. The nodes contain the information or actions for the contingency analysis and are connected to each other by as-

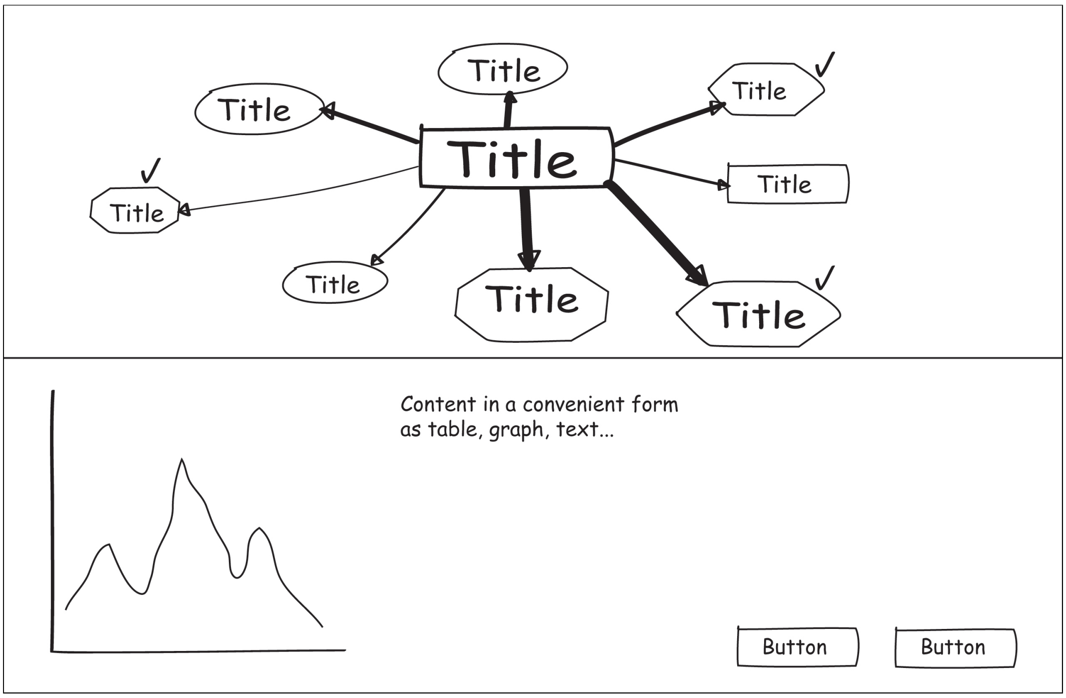

Figure 5.1: Paper prototype A

sociations.

The associations help the operator to find desired information and to integrate them into the context of the current situation.

Associations support orientation

5.1.1 Approach A

The window is horizontally separated into two parts, like in Personal Brain (see chapter 4.3—"Organizing tools"). In the upper part the nodes and their associations are visible. The focused node is centered and the connected nodes are arranged around it. The lower part is used to present the content of the focused node (see figure 5.1).

Separation of content and associations

Each node is represented by a geometric form, like an ellipse, rectangle, or octagon, and its title. The form of the node indicates the type of the content, like grid data, external information, or reports.

Form indicates content type

Two levels visible

For screen space and clarity reasons, only the focused node and the directly associated nodes are visible. The remaining nodes are not visible.

Size of a node as indicator of importance

Some nodes are more important than others. For example a contingency may cause more severe violations than others, or the weather data may be critical, because of extremely low temperatures. The size of a node indicates the importance of the contained information.

Marking of visited nodes

Once visited, a node is labeled by a check mark.

Arrows for associations

Associations between nodes are represented by arrows. The directions of the arrows give a sense of hierarchy between the nodes. The thickness of an arrow is a second indicator for the importance of the node, the arrow is directing to.

Selecting a node shows its content and centers it

To access the content of a node, the operator can select it. The selected node moves to the center and the associated nodes move to their places around the new focused node. Nodes, which have not been visible, before appear, and nodes, which were visible, but are not neighbors of the focus node disappear. At the same time the content in the lower part of the window changes to the content of the newly focused node.

5.1.2 Approach B

Content and associations in one window

In the second approach there is no separation of the window. The content of the nodes is connected with its associations.

Focused node shows all content, other nodes only title

The focused node is centered and fully zoomed to. It consists of a surrounding box, which contains the title at the top and the content of the node. Every other node is represented by its title and a surrounding box (see figure 5.2).

Color of a node indicates content type

The content type is indicated by different use of colors. The size of the node is inversely proportional to the distance to the focused node, i.e., how many nodes are between both nodes.

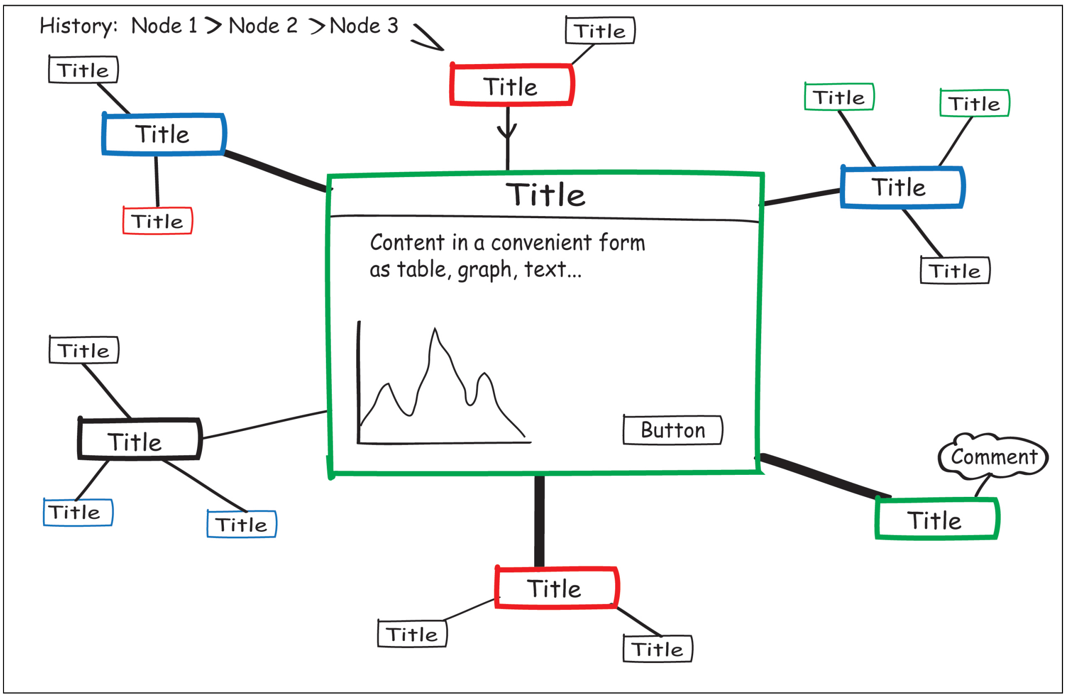

Figure 5.2: Paper prototype B

For screen space and clarity reasons, only the focused node and the next two levels of nodes are visible. The remaining nodes are not visible.

Three levels visible

Lines represent the associations between nodes. The thickness of the lines represents the importance of the corresponding node.

Lines for associations

The operator can add a comment to a node. It is included with a cloud associated to the node.

Clouds for comments

To keep track of the already visited nodes a history is included. It looks similar to breadcrumbs, which are often used in web design. The titles of the nodes, that has been visited lately, are written in the visiting order, separated by arrows. This history bar is placed at the top left of the window and an arrow points at the last visited node, which is still visible. Another arrow is placed on the line between this node and the focused node.

History bar

To access the content of a node, the operator can select it.

Selecting a node zooms to it and centers it

It is moved to the center and 'grows' to its full size, so that the whole content is visible. The old focused node 'shrinks' to the size, depending on the distance to the focused node. The visibility of associated nodes change correspondingly, i.e., nodes with one or two level distance to the new focused node appear—if they had not been visible—and all other visible nodes disappear. In addition, the size of the visible nodes change to the size according to the distance of the focused node.

Aspect	Approach A	Approach B
Content and associations	Separated	Combined
Indicator for content type	Geometric form	Color
Visible levels	Two	Three
Indicator of importance	Size of a node Line thickness	Line thickness
Associations by	Arrows	Lines
Marking of visited nodes	Check marks	None
Comments	None	Cloud
History bar	None	Yes
Selecting a node	Centers it Opens content separate	Centers it Zooms to content

Table 5.1: Summary of differences in the two paper prototypes

Suitable representation of content is required

For both approaches the content of the focused node is presented in an appropriate way—either in the lower part of the window or in the node itself. The content may consist of data, represented in lists, tables, diagrams etc., or possibilities to interact with the system, like fields,or buttons. It is supposed, that the content itself is designed best under usability considerations.

5.1.3 Evaluation

Early evaluation to avoid big problems

Early evaluation is necessary, even before any implementation is done. Big problems can be detected early and the designer is prevented of reimplementing a whole design, caused by bad design decisions [Dix et al., 1997].

Expert review

As it was difficult to get real users—operators—for a user test (see chapter 7—"Evaluation"), an expert review was used to evaluate the two paper prototypes. As the experts know the domain and the working context in a control center, but cannot execute the handling of the contingency analysis, a review is more suitable than a user test. The evaluation helped to decide between the different aspects realized in the two paper prototypes. In a review the comparison of the differences in both approaches is easier. A user test is better suited to detect problems.

Expert reviews can be conducted at any stage of the design process. Van Duyne et al. [2003] state them as "effective technique for evaluating [...] without the need to involve customers". They are an alternative, if no users can or should be integrated in the evaluation. But expert reviews should not fully replace evaluation with real users during the entire design process.

There are different forms of expert reviews, like heuristic evaluation, guideline reviews, or consistency inspections. Which one is suitable for the evaluation at hand, depends on the goal and the main question it should answer.

Single aspects have been discussed

The review started with an explanation of both prototypes and their functionality. The overall function and all details were described in their different implementations. Afterwards, the prototypes were discussed with the participants. I wanted to know, which aspects should not be implemented, because they were inappropriate for the conditions in a control center. For the aspects, implemented differently in both prototypes, it was discussed, which suggestion is more suitable in the context of a control center. During the discussion no decision for one approach as a whole was made.

While discussing the participants were free to express their opinion about the design. They were encouraged to tell, what they think about the design, and what could cause problems in a control center. Thus, influence on the answers by the interviewer was reduced. Participants were further questioned by the interviewer to clarify aspects not mentioned by themselves.

Reviewed aspects

The following aspects have been discussed, during the review, and were addressed, if not mentioned by the participant:

- Separation or combination of content and associations
- Indicator of content type
- Indicator for importance of a node
- Associations by arrows or lines
- Marking of visited nodes
- Inclusion of comments
- History bar

Experts know power transmission context

Three experts have participated in the reviews: A product lifecycle manager for a power transmission operating system, working for several years in the context of power transmission; a usability expert, working in the context of power transmission for more than a year and before working in the similar area of automation; the chief of the architecture and usability department, working several years in the context of power transmission. To receive independent answers, the discussion took place with each expert one after another.

Two of the experts have also participated in the site visits of two power transmission control centers (see chapter 2.2—"Working Context in a Control Center").

5.1.4 Results

Qualitative feedback to refine the design

The review produced qualitative data in form of comments by the experts. Qualitative results are typical for the evaluation of paper prototypes, as they are not that detailed. In such an early stage this feedback gives important hints to refine the design.

In general, all experts liked the idea of helping the operator to navigate by associations and by presenting only the information, he needs for the task at hand. They understood the concept of selecting nodes, to get to see their content and the principle of repositioning of nodes after selecting it.

In the following I will describe the different aspects, which have been discussed, and the impacts on the design.

Combination of content and associations

- The approach of the combination of content and associations in one window, like in approach B (figure 5.2), were preferred.

  The combination of content and associations in one window reduces the need to change the focus of attention. The context of an information is visible without changing the focus. It helps the operator to orientate and to see the information in the context. The associations, directly connected to the content, help the operator to navigate to related information or actions.

  A change of the focused node is more visible to the operator, if the content and the associations are visible in the same window. While changing the focused node, it is visible, that another node opens. In contrast a focus change in separated windows is executed by two different changes—the change of the focused node in the upper part and a change of the content in the lower part. The connection of content and structure is not that obvious.

Showing the content type is useful for orientation

- The concept to make the types of a node visible was rated positive, as it helps the operator to know, for example, if a node gives information about the grid state, information about external factors, or a procedure to solve the situation.

  No use of color for content type

  The use of color for differentiating between content types was criticized. In the user interface of an operating system colors are used to get attention for alarms or warnings. Especially, during the contingency analysis possible violations are highlighted with colors, marking the type of violation. In addition different states are often indicated by colors. The colors already used for these purposes cannot be used to mark different content types, like in approach B (figure 5.2), or otherwise the operator may get confused. But it is not suitable to use additional colors as the screen may get too colored and critical situations, like alarms and warnings, could be overseen more easily.

Geometric forms indicate content type

Different geometric forms for the surrounding boxes, like in approach A (figure 5.1), are more appropriate to mark the different types. The operator is used to differentiate by forms. In one-line diagrams he recognizes different devices by their symbol. What different types of content exist, will be explained in chapter 5.3—"Content Structure". If it is really possible to use forms, depends on the number of types. There is only a limited number of geometric forms, which may be used for this purpose. They have to be easy to differentiate and suitable to contain enough content. The use of forms is further discussed in chapter 6.1—"Software Prototype".

Three level of nodes

- To allow a fast access to the content of nodes, the direct associated nodes and their associated nodes are visible. In approach B the distance of nodes to the focused node is indicated by the size of the node (see figure 5.2). With a greater amount of visible nodes the screen would be too crowded, and the map would produce more confusion than orientation. The amount of visible nodes is further discussed in chapter 6.1—"Software Prototype".

Thickness of associations indicate importance

- As the size is already used to indicate the distance to the focused node, it cannot be used to indicate the importance of a node. The thickness of the association lines—in both approaches (figure 5.2 and 5.1)—remain as indicator for the importance. It is a suitable element to highlight a node, but was questioned to be sufficient to direct the operator's attention to an important node.

Brightness of nodes indicate importance

During the discussion the brightness of a node was suggested as second indicator for the importance. A darker or brighter node—depending on the brightness of the background—raises the operator's attention to that node. See chapter 5.2—"Intermediate Prototype" for further results.

- The use of arrows as associations, like in approach A (figure 5.1), was preferred for two reasons. The task proposes a natural sequence between the direct associated nodes—proposed actions are not interesting before looking at the contingency itself. Arrows represent a workflow better than simple lines.

  The thickness of an association line indicates the importance of a node. But there are two nodes connected to the line. With the arrow pointing to a node, the operator knows, which node is highlighted by the thicker line.

Arrows as associations

- Knowing, whether he has already visited a node or not, helps the operator to judge the situation. But it is not an essential information to solve a task. The checkmark, used in approach A (figure 5.1), is too prominent to use for marking visited nodes. A more subtle clue is to change the color of the arrow pointing to the node, in the same way, like marking visited links on a web site. As mentioned before, the use of too many colors is critical, but a change in the brightness of the link does not overwhelm.

Visited nodes indicated by change of arrow brightness

- As the relative position of nodes keep constant, it is not suitable to include an arrow from the history bar to the last visited visible node and another arrow on the association to the focused node, like in approach B (figure 5.2). Either the position of the last visited node or the position of the history bar needed to be changed, when changing the focus. The history bar should be placed all time at the same position. As it is not the main instrument to access nodes, it is placed at a not that prominent place on the screen—the lower left corner.

History bar lower left corner

- The integration of comments, like in approach B (figure 5.2), was appreciated by the experts, especially for the situation of a shift change. It was questioned, if the possibility to add a comment to every node is necessary. This aspect is analyzed in chapter 5.3—"Content Structure".

Comments helpful, specially for shift changes

Aspect	Preferred	Approach A or B
Content and associations	Combined	B
Indicator for content type	Geometric form	A
Visible levels	Three	B
Indicator of importance	Line thickness Node Brightness	Both None
Associations by	Arrows	A
Marking of visited nodes	Arrow brightness	None
Comments	Unclear	
History bar	Yes but differently	

Table 5.2: Summary of decisions about different aspects of the two paper prototypes

5.2 Intermediate Prototype

As new indicator for the importance of a node, it is planned to use its brightness and the thickness of its association. Even though Ware [2000] states, that differences in brightness guide the users attention, I decided to test it, before implementing in a more advanced prototype.

Screenshot prototype

It is easier to realize different brightness with screenshot prototypes, than with hand drawn paper prototypes. A screenshot prototype is a digital drawn picture of the screen or of several screens and can be the next step in the DIA-cycle.

Adequate use of screen space

The main reason to build this intermediate prototype was to test, if it is possible to direct the attention of the user by the combination of node brightness and arrow thickness. But the creation of the prototype gave the possibility to also test other aspects. With a digital prototype I could get a first impression of wise use of screen space. The focused node must have a minimum size to present its content in an appropriate way. But if the node is to big, there is not enough space left to adequately display the associated nodes, and especially not in different sizes for different levels. Reasonable sizes and number of nodes were found.

User expectations

With a digital picture of the screen it is easier to ask for the expectation of the user. What does he think, can be done, with such an interface? How does he expect the interface to react? A screenshot looks more like a real application than a paper prototype and it is more natural to build realistic expectations.

The paper prototypes and the results of their evaluation (see chapter 5.1—"Paper Prototypes") were used as basis for the screenshot prototype.

Main color: blue

As mentioned in chapter 5.1.4—"Results" some colors are already preoccupied in their meaning. It is not possible to use red, yellow, green or orange as main color for the nodes. There remain only blue and grey as neutral clearly differentiable colors. For both it is possible to change the brightness to attract more attention. Blue was chosen as main color.

Dark blue nodes = more important

The background of the design was chosen to be white. A higher contrast directs the attention and a darker color for a node catches the attention more than a lighter one. A node in darker blue is more important than one in lighter blue (see figure 5.3). Under the consideration that differences are more distinguishable with discrete color coding, blue in three different levels of brightness is used.

Thicker arrow = more important node

The thickness of an arrow to a node is proportional to the importance of the node, it is directing to. The arrow and the brightness indicate the same importance. A discrete range is also chosen for the thickness of the arrows, as it is considered to be easier distinguishable than a continuous range.

5.2.1 Evaluation

The screenshot prototype was mainly built to check, if the combination of a node's brightness and its arrow's thickness indicate the importance of the node. In addition the first digital image of the design could be used to test the

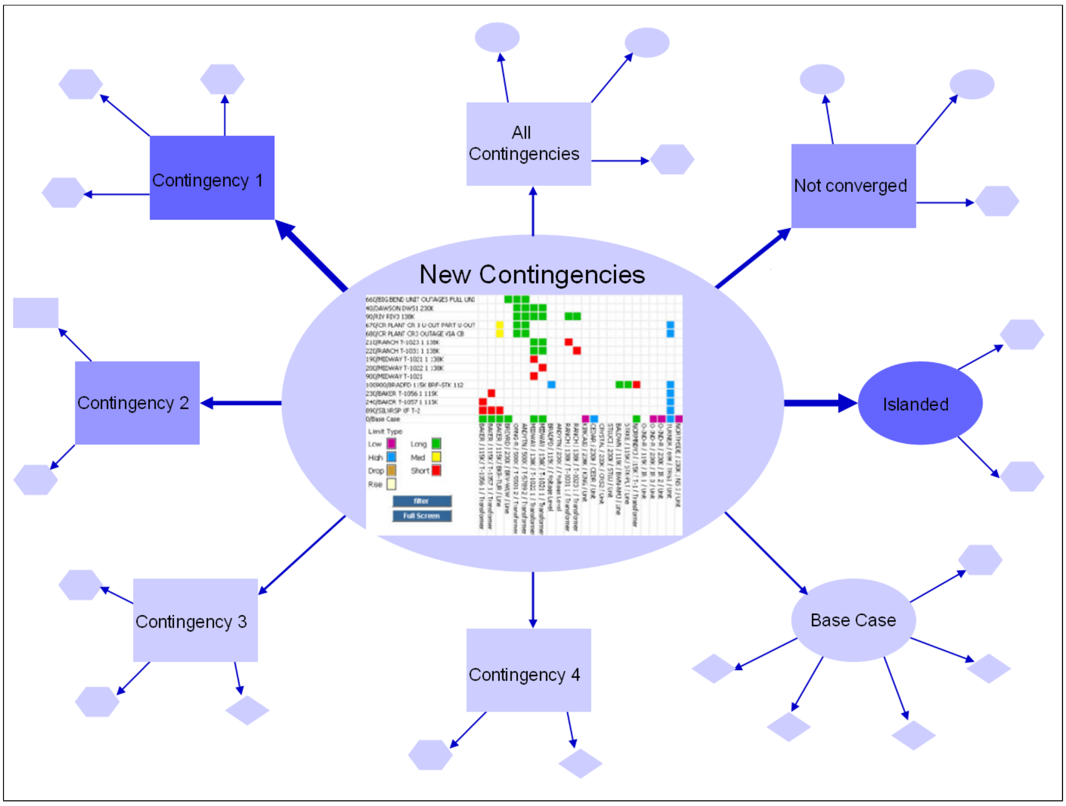

Figure 5.3: Screenshot prototype as intermediate prototype

users expectation about interaction possibilities and reactions of the system.

To confirm the assumption a short, informal evaluation was done. For the evaluation the prototype was shown to several test participants, and they were questioned about their expectations.

Engineers as participants

Test participants have been four engineers, working in the context of power transmission. All participants are working in the development of power system related software and have a degree in a technical subject, either electrical engineering or computer science. Their educational background is similar to the educational background of a typical operator.

In the introduction the participants were told, that they will see a prototype of an interface in a control center to analyze and solve a problematic situation in a grid of power transmission. Then they were asked several questions:

Interview guideline

- How do you think, does this interface work? What do you think, you can do?
- What do you expect to happen after clicking on a node?
- Which node would you choose first?
- Why did you choose this node?

The questions have only been a guideline. The interviewer reacted to the participants and used the terms, the participants used in their answers, to reduce confusion by different wording.

5.2.2 Results

Qualitative feedback

The results of the evaluation are of qualitative nature. The participants' comments give helpful feedback on the design.

For each aspect asked in the test, I will summarize the answers and discuss the conclusions:

Information expected in the nodes

- All participants supposed, that the interface can be used to get more information about the problematic situation. They noticed, that they can click on the nodes to access their content. It is supposed, that the interaction is fast learnable with the first use—also an important usability factor.

Clicking on a node = access to content

- With the knowledge, how to interact with the interface, all participants expected, that clicking a node gives access to the content of that node by opening it. There is no exact expectation, how that happens, and how the focused node reacts. But it is sufficient to know, how to get access to the content of the other nodes.

Brightness of nodes and thickness of arrows

- The choices of nodes by the participants suggest, that the brightness of nodes and the thickness of links focus the attention to the important nodes. It has to be analyzed, if this also holds when the content is taken more into consideration.

Because of the low number of test participants, no general conclusions about the use of brightness or thickness to indicate importance can be made. But the approvement of all participants strengthens my assumption. The brightness of a node and the thickness of the corresponding arrow will be used in the next prototype of the design and further examined.

Limited number of nodes for clarity

As mentioned before the production of the screenshot prototype was a possibility to test the use of screen space, i.e., the number and sizes of nodes and visible levels. Including too many nodes, the screen would be too crowded and the probability of disorientating the operator increases. The amount of eight second level nodes, used in the screenshot prototype, did not overwhelm the participants.

The number of necessary nodes depend on the content structure. To decide, if the number of displayed nodes can be limited, further research of the underlying content structure is required.

Semantic Zooming

In the screenshot prototype it is obvious, that the second level nodes are big enough to already contain some information. Using semantic zooming for the design allows to present the nodes in the second level with their most important information. The operator gets an first impression of the content of that node. The presented information helps him to decide about the relevance of a node and thus to orientate in the information space (see chapter 6.1—"Software Prototype" for more information about semantic zooming in the design).

5.3 Content Structure

Detailed view of content necessary

Parallel to the creation and evaluation of the first prototypes, I conducted a research about desired information and subtasks for the contingency analysis. The task analysis (see chapter 2.5.1—"Task Analysis") already gives an overview of the content. But for designing the interface a more detailed view is necessary.

It is important to know, which information and subtasks are helpful and which are absolutely necessary. It is not sufficient to know, that the weather can influence the operators reaction, but which weather data, like temperature, humidity, or forecast. The detailed contents need to be combined and structured.

Content map of the interface

The detailed analysis of the content is used to build the map building the interface. It describes the content integrated in each node, the type of the nodes, and the associations between the nodes.

With the information I collected during the task analysis I built a map describing the structure. An additional list describes the detailed content summarized under each point. Figure 5.4 shows the final map. The entry for one contingency, for example, on the detailed list looks as following:

Single Contingency

- Performance indices
- Time to react
- Outaged equipment
- Violated equipment
- Type of violation
- Values: base case (current data) and contingency
- Link to one line diagram
- Contingency identification (number)

Refinement with engineers and experts

With the map I talked to several engineers, working in the development and test of control center operating systems, and the experts, that evaluated the paper prototype (see

Figure 5.4: Content structure of the contingency analysis

chapter 5.1.3—"Evaluation"). During that interviews I continually refined the map and the list in collaboration with the interviewed persons. This approach guaranteed a constant evaluation of the map and list.

Four content types

With the final map it was possible to identify different types of nodes, which should be distinguished in the design (see chapter 5.1.4—"Results"). In figure 5.4 dark blue nodes contain data about the system state, green nodes about external factors, orange nodes older, archived information, and yellow nodes allow actions, e.g. changes in the grid or comments.

The starting node of the interface is the node 'New Contingencies'. The node 'Single Contingency 1' and its children is an example for all nodes describing a single contingency. The nodes surrounded by the broken lines stand exemplary for all nodes containing different contingencies and their associated nodes.

A short description of the nodes:

- New Contingencies: Overview of new calculated contingencies with violations
- Single Contingency: Data for one contingency with violations
- Geography: Geographical data connected to the contingency
- Weather: Current weather situation and forecast
- Clients affected: Clients, that would be affected by an outage and the possible consequences
- Older Failures: Summary of older failures comparable to the contingency
- Report: Data about the older failure
- Actions: Proposed actions for the contingency, either preventive or reactive
- Pass to Study: Pass the contingency to the study mode to develop a solution
- Comments: Possibility to leave a comment about the contingency for later
- Cases, that did not converge: Contingencies, where errors in the calculation occurred
- Cases, that islanded the system: Contingencies, that separates the grid, i.e., creates islands

- Dummy: Place-holder for contingencies without their own node
- All Contingencies: Overview of all, also earlier calculated, contingencies with violations
- Solved cases: Contingencies, that do not violate the grid
- Base Case: Current grid state, base for calculation of the contingencies

Chapter 6

High-fidelity Prototype

Dynamic behavior for impression of 'look and feel'

In the next step of the iterative design process a dynamic behavior is included in the prototype to get the "look and feel" of the interface. A mock-up simulates and evaluates main aspects of the user interface, like content, visuals, interactivity or media [Benyon et al., 2005]. The emphasis lies on the the user interface and not on the underlying functionality of the system.

Feedback about details

The evaluation of such an interface produces feedback on the details of the interface and the interaction. Its appearance and behavior should be similar to the appearance and behavior of the final product. Participants of an evaluation typically focusses on smaller problems than on bigger design issues. They are afraid to criticize a finished looking prototype and to cause too much change. The user feedback is about the interaction with the system and its "feel".

My goal was to get feedback by real users, i.e., by operators with a convincing prototype. I implemented a software prototype and went to a power transmission control center. This chapter describes the design, implementation, and evaluation of the software prototype.

6.1 Software Prototype

In the prototype the findings of the first prototype evaluations were considered (see chapter 5.1.4—"Results" and 5.2.2—"Results"). The content, based on the structure found in chapter 5.3—"Content Structure", is included in the prototype.

Besides including the content, semantic zooming was also incorporated in the prototype. Three levels of zooming are used (see figure 6.1):

Focused node in full zoom

- First level: The focused node is the biggest one and approximately half the screen size. It consist of the title, the content in full detail, and the surrounding geometric form. It is always placed in the center of the screen. Other nodes than the starting node are slightly shifted, so that there is enough space to keep the staring node always visible.

Associated nodes with important facts

- Second level: The nodes directly connected to the focused node are smaller, approximately a quarter of the focused node. They consist of the title, important facts, which give the operator a first orientation, and the surrounding geometric form. For a contingency, for example, it is the types of violations and their number, plus the time to react in case of the failure.

Next level only the title

- Third level: The next level of nodes is presented only by their title. There is neither additional facts, nor the surrounding form.

Start node always visible

The nodes with a distance to the focused node higher than two are not visible with one exception. When navigating to a deeper level, the start node, which gives an overview of all new contingencies stays always visible together with its directly associated nodes. It allows the operator to return to the beginning at any time, and it can be used to notify about new contingencies by changing the color to an attention catching color, like red or orange. When not in the first

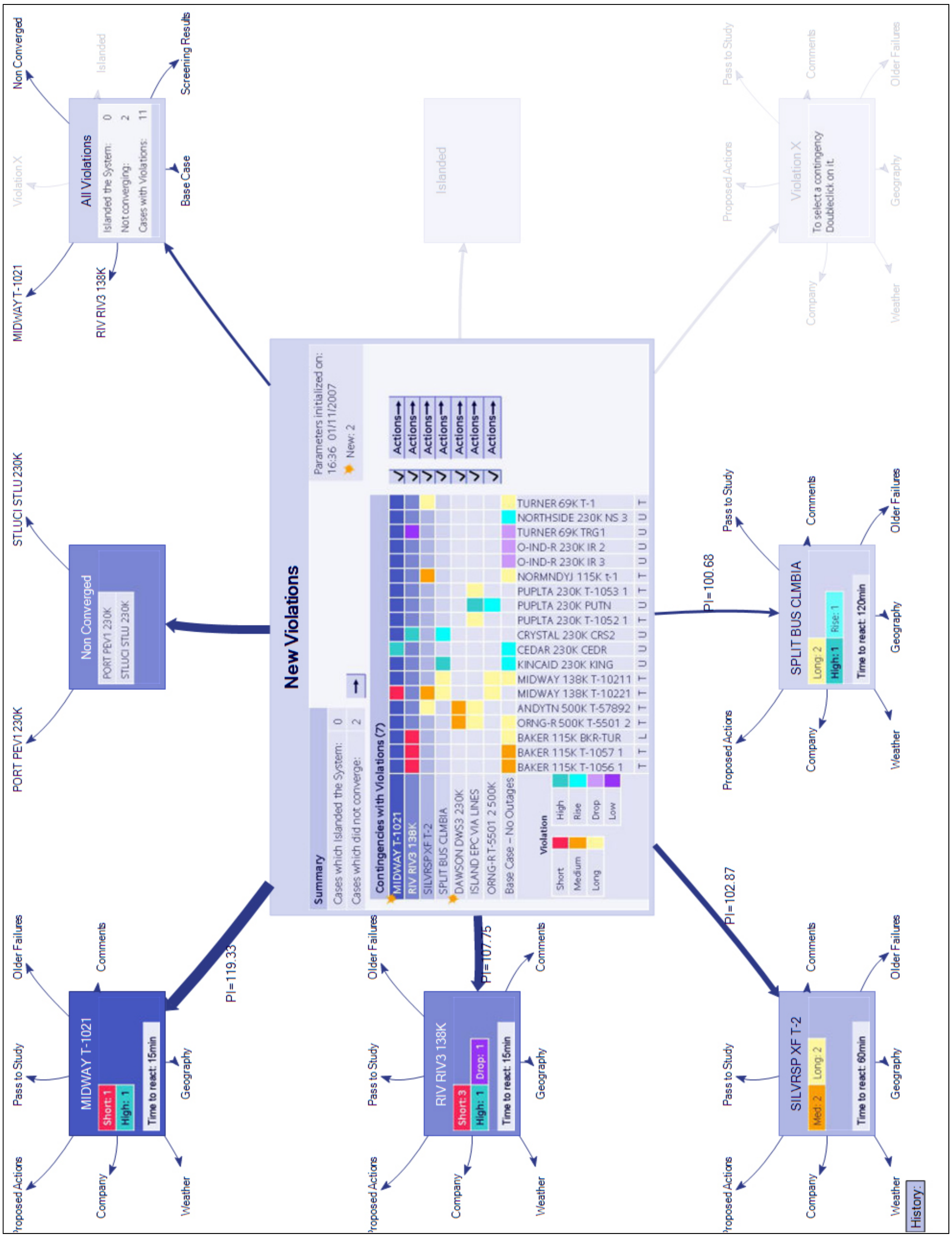

Figure 6.1: Starting screen of the software prototype

or second level the node 'New Contingencies' is presented with its title and a box—smaller than the second level box.

Automatic zooming
automatic zooming

The design uses automatic zooming (see chapter 3.2—"Zooming User Interfaces"). By clicking on a node, the node is selected and zoomed to. The node moves to the center and grows to the biggest size with fully detailed content. All associated nodes, in second or third level, become visible, if they had not been, and change to the according size. All other visible nodes, except the starting node and its associated ones, become invisible.

Relative positions of nodes is constant

With a focus change every node gets a new place on the screen. While doing that, the relative position to the other nodes stays consistent (see figure 6.2). The screen is comparable to a magnifier moving over a map, showing only the nodes in its range, with the node in the middle in full size.

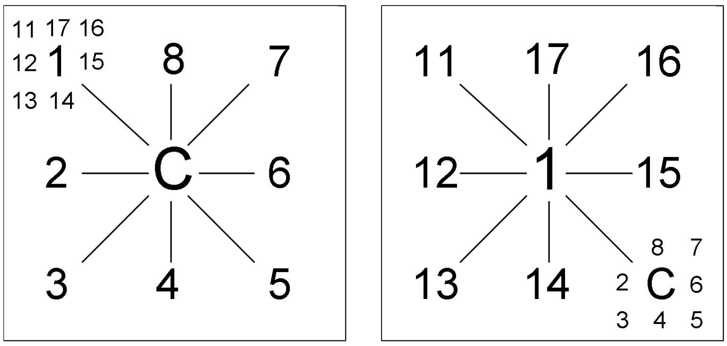

Figure 6.2: Example for the stability of the nodes' relative positions; left: starting constellation, right: after zooming to node 1

Animated transitions

The transitions, changing size as well as changing position, are animated. It supports the operator in keeping the orientation, because he does not loose the context of the nodes (see appendix A—"Transition between two nodes in the focus"). For the animation of the transition in this design a time of one second is used. An argument against using animated transition would be, that time is wasted for the transition. But without animation a user needs at least the same amount of time to reorientate.

Importance indicated by brightness of nodes and thickness of associations

The importance of nodes is indicated by their brightness and the thickness of the pointing arrow (see chapter 5.2.2—

"Results"). A discrete range with four levels is used for the importance. The reason for the level of importance may be written in a label next to the arrow. For example the performance index of a contingency, an index for the severity, is written next to the arrow to the according contingency.

In chapter 5.3—"Content Structure" I identified four types of content. The types are indicated by different forms:

Form of node indicates its content

- Rectangle: Data about the current grid state
- Octagon: Older, archived information, like reports
- Hexagon: External data, not directly concerning the state of the grid
- Rounded rectangle: Possibility for actions and changes

In some cases there may be no data for a node. For example contingencies, which produce islands in the system, are displayed in an extra node. If there is no such contingency, there is no data, which can be shown in the according node. In this case the node is presented transparent, instead of fully deleting this node. If the node would not be there, the operator may miss it and search for the information. With the transparent version the node is visible, but it is also fast recognizable, that there are no data contained and there are no contingencies islanding the grid.

Transparent nodes contain no data

Arrows to an already visited node are painted in a lighter blue, than the unvisited ones.

Visited nodes indicated by lighter arrows

The number of nodes, visible at one time, is limited to eight of level two. But it is possible, that there are more contingencies with violations, than representable. A 'dummy' node without a predefined contingency is integrated. Like the other empty nodes, the dummy node and its dummy children are transparent as long as they are empty.

To choose such an contingencies, without an own node, the operator doubleclicks on the name of the contingency in the overview. To visualize, which contingency will be presented in the dummy node, the title of the contingency 'flies' to the dummy node. The dummy node is then zoomed to the center, like every other node.

As seen in the previous example, there are other interactions, than choosing a node. To avoid confusion with zooming interaction, other interactions only exists in the focused node. In the focused node The interactivity is similar to an application window.

6.2 Implementation

Implementation in C# using GoDiagram and XML

The prototype was implemented in C# using the library GoDiagram. The content structure and the initializing properties of the nodes and links are stored in a XML-file, while the contents are realized as images.

GoDiagram for visualizing nodes and links

GoDiagram[1] is a library to visualize and manipulate diagrams. It offers various graphical objects, which can be combined to adjust their appearance, properties, and behavior as desired. Its concept is based on nodes and linking of nodes, and thus, is well suited to implement my design, which is based on a similar concept. In addition GoDiagram offers classes to transform XML data into objects.

Easier integration

To support the integration of the new interface in existing systems, I used the .NET framework and C# to implement the prototype. Although GoDiagram is available for several programming languages, the implementation for C# offers most of the desired functionalities.

Initialization with XML-data

On initialization specialized classes (TransformContentNode, TransformContent, TransformImage, and TransformLink) are used to transform the XML elements into objects and set their variables to their initial values. The starting screen is painted.

Animated zooming by depth-first search

Zooming to another node is triggered by clicking on any visual node. A focus change is realized by changing the sizes, positions, and visibility of the current visible and the

[1] http://www.nwoods.com

newly visible nodes. A depth-first search traverses the visible nodes, starting at the current focused node, and sets their new properties. The animated transitions are implemented by repainting the visible nodes every tenth of a second with a modification of their sizes and positions. For each iteration—in total ten—a tenth of the difference between old and new position, respectively old and new size, is subtracted. A second depth-first search, starting with the newly focused node, updates the properties of the nodes, including changes of the visibility of nodes.

Structure stored in XML-file

The XML-file stores the information about the nodes and their associations, needed for the initialization (see figure 6.3). XML offers an easy possibility to define the nodes and their structure outside of the implementation. In addition, XML can be used for communication between the interface and other applications in a later implementation.

```
<node id="00" type="0" position="0" title="MIDWAY T-1021"
      priority="1">
   <content small="false">
      <image path="con1.jpg" size="590 355">
      </image>
   </content>
   <content small="true">
      <image path="con1small.jpg" size="140 65">
      </image>
   </content>
</node>
<link from="0" to="00" priority="1" label="PI=119.33">
</link>
```

Figure 6.3: Extract of the XML-file with the elements: node, content, image, and link

`<node>` is used to create a new node. The attributes are:

- *id*: a unique number to identify the node
- *type*: numerical identifier of the content type (0: system state; 1: external factors; 2: archived information; 3: actions)

- *position*: identification for the relative position (to its parent node), used to assign the absolute position
- *title*: title, displayed in the node
- *priority*: importance of the node

`<content>` is a child of `<node>`, exists twice in each node. The attribute *small* marks the level of zooming, at which this content is displayed.

`<image>` is a child of `<content>` and its attributes define the *path* to the content image and its *size*.

`<link>` defines a link between two nodes:

- *from*: defines the outgoing node, it is similar to the attribute *id* of `<node>`
- *to*: defines the ingoing node, see above
- *priority*: importance of the ingoing node
- *label*: text for an optional label next to the link

6.3 Evaluation

Two evaluation rounds

Like the prior prototypes the software prototype was evaluated. The evaluation was carried out in two parts. During the first part the participants were experts in the domain of usability or of power transmission. The second evaluation took place in a power transmission control center with employees of the control center.

Presentation of prototype with discussion

The procedure of both parts of the evaluation was more or less the same. The participants got an introduction to the main idea of the design and to the contingency analysis. Afterwards the prototype was presented in its full functionality. In a discussion, feedback about positive and negative aspects, additional questions, and constraints were collected.

Participants first round: Experts in domain and usability

The evaluation with the experts was accomplished in three rounds, i.e., there have been three presentations and discussions. In total 23 people have participated in the first part of evaluation, all of them working in a company developing systems in the domain of power transmission. They work in different functions, like developer, tester, manager, architect or usability engineer. They know the domain, but not necessarily the work in a control center or the interface of an operating system. The introduction included a description of problems with an interface in a control center and of the contingency analysis.

Participants second round: three employees of the control center

During the visit in the control center there have been three employees, the head of grid operation, the head of IT-systems and another person working in IT-systems. They know the work in a control center and the system for several years. The introduction consisted only of a description of the main idea of the new interface.

Presentation of the prototype along the workflow

In all cases the presentation of the prototype itself was done in the same way. The prototype was explained along the workflow of the contingency analysis (see chapter 2.5.1—"Task Analysis"). For each step, that an operator would perform, was shown, how it was implemented in the prototype. Starting with the 'New Violations' node the most urgent contingency was chosen and its content and the associated nodes demonstrated. After the presentation along the workflow, additional functionalities were presented, like the interaction possibilities in the focused node and the dummy node.

The feedback round was a free discussion. A few questions guided the participants to helpful feedback:

- What do you think about the idea of connecting the content by associations?
- Is there anything missing? Is there anything needless?
- What do you like? What do you dislike?
- Are there any constraints, associated to a control center, violated? Are there other constraints, which have to be taken care of?

Next to the questions the participants were asked to give free comments about the design.

Limited time at the control center led to group evaluation

Busy times at the control center, caused by internal changes in the structure of the control center, reduced the available evaluation time to one and a half hour. Because of insufficient time to make individual evaluations, a group evaluation was necessary.

Limited number of persons led to discussion

To get the opinions and impressions of each individual person, I prepared a questionnaire. Several questions about the interface and its behavior should have been answered. Before the meeting in the control center it was not known, how many people would be present. With only three persons present, of whom only one was an operator, it was decided to cancel the questionnaire and answer the questions in a discussion. Questions by the participants were answered and more details about the participants' answers were collected.

6.4 Results

The results of the evaluation are qualitative feedback in form of comments by the participants. Following, I will summarize the feedback separated for each evaluation round, first the engineers, then the control center employees. The distinction is important, because the engineers give feedback as experts in the domain, but the feedback gained at the control center is given by real users, working with a control center interface. Between the two parts of the evaluation, I made small design changes, based on the comments of the first evaluation round.

6.4.1 Expert Feedback

The majority of the participants liked the idea of navigating via associations. They understood the principle of the interface fast and had no problems to get the functionality. The access to information, necessary and helpful for the task at hand, was said to be very valuable.

The following points were rated important for a control center and the implementation in the design seem to support the operator.

- In the contingency analysis it is important to know, if a failure in the grid causes violations and thus additional failures. The main task of an operator is to keep the grid in a N-1 security. That means he has to clear the contingencies with violations.

  Worst contingencies are visible fast

  The design gives a good overview of the contingencies and the consequences they may have. The type and number of violations is visible fast, especially in the second level presentation of the contingencies. The ranking of contingencies was rated positive. The redundant marking with the brightness of the nodes and the thickness of the arrows was mentioned to be helpful.

- One-line diagram opening in an own window is necessary

  It is not only important to see, which elements of the grid may be affected, but also to know where these elements are. Their technical position is of interest—the connection to neighboring elements. The one-line diagram (see 2.3—"Current Operating Systems") was seen as absolutely necessary in this context and the possibility to invoke it form the overview is essential. It was appreciated that the one-line diagram opens in its own window, as it may be necessary to keep it, even when changing the nodes. It was suggested to open it on a different screen, when working with several screens. In the node, representing a single contingency, the possibility to view the affected elements in the one-line diagram is not obvious enough. Especially at there, this information is of interest. A button to view the one-line diagram should be integrated in the first level presentation of a contingency node.

- Geographical information is useful

  But not only the technical position is of interest. Also the geographical positions of the elements and their surrounding may hold important information about the urgency and the possibilities to handle the case.

The operator can judge the cases easier. To integrate a map with additional information and the association to the respective contingency was seen as good approach to include such information.

Further information support the decision

- The integration of new information like the weather, reports of passed problems, and comments was judged as positive aids. Only the information about affected companies are not necessary and cannot be taken into consideration during the handling of a contingency.

Criticized aspects, missing information and what has been done, before the second evaluation round, or could be done about them is described in the following.

Time to react more obvious

- An important factor in the handling of contingencies with violations is the time between the failure of an element and its consequences, failures of additional elements. The time to react is essential for the operator to decide, wether and how to solve the contingency. Contingencies with a short reaction time should be handled with an preventive action, while for cases with a long reaction time reactive actions may be sufficient. In the design the cases, with a critical reaction time are not obvious enough. A better highlighting of time critical contingencies was demanded.

  The time to react was included in the first level as well as in the second level representation of the node of a single contingency.

Suggested actions were not expected under the term 'actions'

- The suggested procedures for a contingency are a starting point to develop an action. Some participants did not expect them in the node 'Proposed Actions'. They expected the procedures at a more prominent place with the possibility to access them faster.

  Wording right?

  If the suggested procedures are expected under the node 'Proposed Actions' is a matter of wording. My initial research recommended, that the term would be understood by operators. Right usage of terms should be further investigated.

Two possible ways direct to the node, one button next to the contingency in the overview and the node connected to the contingency node. Thus there is already a prominent access from the start. Is the contingency node focused, the action node is by default rated as quite important and therefore highlighted by a darker node color and a thicker arrow.

Two ways to access suggested actions fast

- It is important to notice, if there is a new contingency with violations, especially if it is a time critical one. The operator should see out of the corner of an eye, that incidents occurred and that a reaction is required. It must be made visible, when there are new cases. In particular if the node 'New Violations' is not the focused node, there should be an mechanism to get the operator's attention.

Mandatory: warning in case of new critical contingencies

In the case, that the node 'New Violations' is not in the focus, new contingencies with violations may be alerted by changing the color of the node. A change to a color known for warning, like orange or red, should direct the operators attention to the overview of new contingencies. Depending on the importance a warning could get even more attention, with an additional acoustic signal or a blinking of the node.

Color change of overview node in case of new contingencies

The general priority of the overview can be set higher, if there is any critical case, possibly to the same priority as the worst case. Then, the node is more visible to the operator, if there are still cases to solve. He gets an easy impression, if he needs to do something at any time.

In the overview new cases with contingencies are marked with an orange star—realized in the prototype before the second part of evaluation. An orange or red border around the node could signal new cases.

- At some points the participants wished more descriptions included in the design. They wanted to insert the meaning of colors, brightness, form or thickness of lines for example. Another example were explanations for the types of violations or the like. A tooltip was suggested to include explanations.

No inclusion of descriptions

Although the operator should know the types of violations, that may occur a context-sensitive online help

Online help

could be useful. This online-help should also contain information about visual clues, used in the design.

- A connection to the documentation may be useful.

  Integration of documentation

  Like mentioned before the integration of the documentation connected to the single cases does not make sense, as their is no logical connection between the single contingencies and the documentation. On the other hand a general integration of the documentation can be helpful for the operator.

Marking visited nodes support the operator

- The marking of visited nodes was rated as helpful tool to orientate. But it was also noted, that an additional marking could make the difference between visited and unvisited nodes more obvious.

  The information, if a node has been visited, is helpful but not essential. Therefore it is sufficient, to mark visited nodes by brightening the arrow, pointing to them. Otherwise there may occur confusion caused by an information overflow.

Overview design

- The starting node with the overview of new contingencies was too confusing and too colored. It was discussed, if this representation is known by the operators and therefore understood.

  Further research about a suitable representation of the starting overview should be done.

6.4.2 Feedback at the Control Center

Innovative concept

The basic feedback of the participants during the evaluation in the control center was positive and they commented the main idea as innovative. Nevertheless, they gave several comments, about what must be considered in a control center.

Comparison to well ordered desk

- The design was compared to a well ordered desk. There is no chaos caused by many open and overlapping windows. It allows good access to the desired information.

- The possibility to move contingencies from the 'New Violations' node to the 'All Violations' node is critical. The operator should always have the possibility to see every critical case, as long as they are unsolved.

  Do not remove unsolved cases

  In best case there should be no cases with violations and normally there are only of few of them. It should be further researched, if it is necessary to use the two nodes or if one overview node is sufficient.

  Two overview nodes necessary?

- The visualization of information was mentioned to be most important. It was stated, that the operator wants to read as few as possible. The highlighting with color is very useful to reduce reading as it guides the operator's attention to important information. Anyhow, they wished less text and more graphical visualization.

  Graphical visualization better than textual one

  Especially the visualization of problems in a one-line diagram is essential and must be available all the time and everywhere.

  Access to one-line diagram is essential

- With such an design the possibility of overlapping windows with other application—no matter if started from within or independent of the interface—should be avoided. Other applications should be launched on a second screen.

  Avoid overlapping windows

- A clear signaling must inform the operator of new important information, such as new contingencies with violation or a worsening of an already existing contingency. Specially if the operator has focused another node than the starting overview.

  Clear warnings in case of new critical contingencies

  A solution to this aspect was already discussed in chapter 6.4.1—"Expert Feedback".

  The overview node should be always presented at least in the second level size and with second level information, i.e., with the number of critical cases visible at every time.

- A consistency in the layout of the screen was mentioned as important factor, for not confusing the operator by changing positions of contents. It was explicitly stated, that in periodic running applications the presentation of results does not change after each run. For example the position of a contingency in a

  Consistency in layout

list should not change after a new run of the calculation. In this context it was stated, that the overview should always be at the same position, if not focused.

The wish to keep the position of the nodes on the screen clashes with the idea of keeping the nodes relative position, while moving in the map. If the overview node is shaped prominent enough, changing its position should not be a big problem, as it stays in the operator's eye. The change to a warning color, in case of new critical contingencies, catches the eye.

The question remains, where to place new critical contingencies after a calculation. To order the contingencies by their severity, gives the operator the opportunity to easily access the worst cases first. He does not need to search the worst case as it is placed at the top of the list and in the left, high corner. Ordering the contingencies by their appearance—or another strict order—helps the operator in orientation to find a case, he has already analyzed.

After a new calculation the changes are not only new critical cases, but cases may also disappear, when they no longer cause violations. Thus, positions could also change without new critical cases, making it impossible to keep the position consistent.

In further research it should be clarified, which order of new critical contingencies support the operator more without causing confusion.

Parallel representation of current and simulated situation

- After a simulation of possible changes in the grid the results are compared to the current state of the grid. A parallel representation of the simulation and the current situation is helpful to decide, if the situation improves to due the changes. It must always be recognizable, which one is the real and which one is the simulated situation.

Two first level nodes for comparison

A parallel representation of both situations could be handled with two big nodes. To solve the problem of insufficient screen space the second level nodes could be reduced to the size normally used for the third level.

The participants of the second evaluation round were really interested in the design and wanted to know more about the further development. They could even imagine to participate in a pilot phase of a further development. In such an open feedback round with the developers present, the comments about the design in question tend to be more positive than with anonymous feedback. Nobody likes to criticize other persons or their work. The offer to take part in further development of this design approach and the willingness to use it in a pilot project, proves real interest in the design.

Interest in pilot project

Chapter 7

Evaluation

When designing a new user interface or a new system, the improvements over existing systems should be investigated. A user test executed with both systems allows comparison of their performance.

User test

In a user test, the participants are asked to perform typical tasks with the system. Participants should be real or potential users of the system. Only with such users the problems, which may occur while working with an interface, could be revealed. A typical task is a task, which is regularly executed using the system. In a user test the situation should be as close to the real situation as possible. The test can either take place in a laboratory or in the field—the actual environment of the system.

Same conditions for both interfaces

In a comparing user test for both interfaces the same tasks are tested, but in an alternating order. Every participant would test both interfaces with different tasks. The tasks and interfaces must be combined in a way, that each scenario is tested with one interface as often as with the other one. The order, in which one participant tests the interface, changes with the participants, to reduce influence by learning effects.

Method depends on the goal

Several methods are used to gain results during user tests, e.g., observing, think-aloud, constructive instruction, or controlled experiments. The choice of a method depends on the goal of the user test. The results are used to compare the systems.

7.1 User Test

Conducting a user test on my new design and on an existing interface can be used to compare the performance of both interfaces.

Participants: Operators

The users of both interface are operators working in a control center of power transmission. Therefore, they are the optimal test participants.

Setting: Demonstration room

As test setting, which is close to the real situation, a demonstration room, imitating a real control center, could be used.

Task: Analyze the results of a contingency analysis

The task, the participants need to fulfill during the test, is an analysis of the results of the contingency analysis, as described in chapter 2.5—"Example Task: Contingency Analysis". At the end of the task they should have developed an action, which they would perform. The performance of both designs is compared by the time to develop such an action, the correctness of the action, or the stress during the task.

Different scenarios

As the task is quite short different scenarios are used for one participant. A scenario consists of all data, describing the state of the grid and the additional information, which are contained in the prototype. Thus, in each task completion a solution to a different sets of critical contingencies needs to be found.

Realistic data is needed

To test my design realistic data needs to be imported. The prototype could be connected to the system used in the demonstration room. It controls a simulation of a power grid. Thereby, the simulated grid could be based on an existing grid or on an imaginary grid.

7.1.1 Challenges

No data about an existing grid

Integrating data about an existing grid, could cause a problem. The grid owning companies do not hand out information about their grids, which is detailed enough to simulate them. The data is kept secret by the owning companies to prevent abuse. Thus, it is difficult to simulate an existing grid.

Missing knowledge of imaginary grid

Operator knows his grid by heart

The alternative, to simulate an imaginary grid, also causes a problem. The operator does not know the simulated grid. As written in chapter 2.2—"Working Context in a Control Center" operators often decide by intuitive reasoning. They know their grid by heart. Even seemingly cryptic element names are immediately recognized. By long working experience they know reactions and real limits of elements by heart. The operator, participating in this test, would not know the grid and its characteristics and an important factor would be missing. Problems during the test could be caused by the interface or by the lack of knowledge about the example grid. Thus, the results may not be representative.

Low number of potential participants

Another, smaller problem is the difficulty to find suitable participants for a user test: the operators themselves. Only with real users the problems, which may occur while working with an interface, could be revealed. A transmission grid spans over long distances, thus, there do not exist many control centers for power transmission close-by. The visited control center, for example, controls the entire transmission grid in Switzerland. The low number and the spreading of possible users reduces the chances to find participants, which have the time and are willing to take part in a user test.

7.2 Testing at a Control Center

Test in a control center

To avoid these problems, the user test could be conducted in a real control center. It is easier to get the real users as test participants, because they do not need to travel to the test setting. The problem of not knowing the grid, is also avoided, as the operators are working in that control center. As the study would be performed in the real working environment, there is no influence by an artificial situation. Both interfaces could be compared in a real working context.

7.2.1 Challenges

Long time study

In such a setting it is still difficult to compare both interfaces. The operators probably work in the control center for several years and know the interface quite well. To reduce the effects of comparing a well known interface to an unknown, the study should be a long time study. Later retrieved results should be weighed higher.

Security concerns

Security reasons cause some difficulties to connect such a new interface to the real operating system. If connected to a real operating system, a problem with the prototype could cause an error in the system with unforeseeable consequences. As described in chapter 2.2—"Working Context in a Control Center" minor errors or problems could cause a failure with cascading consequences, up to black outs. Two constraints must be fulfilled to meet the security concerns.

On one side, there is the danger of interfering the operating system. As the contingency analysis is a purely analyzing task, no changes of the grid would be necessary. The communication between the prototype and the system could be reduced to sending the relevant information to the prototype. This one-way communication would prevent from forbidden interference of the operating system.

On the other side the unknown interface could cause problems in situations under time pressure. Any confusion of the normal work flow caused by the new interface must be avoided. Therefore, the possibility to switch to the normal interface must be given at any time.

7.3 Change to Planning Context

Test in planning context

To reduce interference of the work flow in the control center, the prototype may be tested in the context of planning instead. As mentioned in chapter 2.5—"Example Task: Contingency Analysis" both modes are similar for the contingency analysis, but not exactly the same.

Less security concerns

In the planning context no changes of the grid are possible. It is only possible, and for the evaluation sufficient, to get data about the state of the grid. In the planning system an assumed system state is calculated for a given time and analyzed by the operational planner. There is no danger of interfering the grid by using the prototype on the planning system.

After further research of the contingency analysis in the planning context, the prototype could be adjusted accordingly. Testing the new interface in the planning context, should also be done in a long time study, to reduce the influence of familiarity with the interface.

7.4 Conclusion

Changing to the planning context is probably the best method to start an evaluation of the new design. Even though it does not fully replace a study in the real time mode. But the interface can be improved to the point that it may be tested in the real time mode with less concern about possible confusion or disturbance.

With the current state of the prototype an evaluation was not possible. Next to the already mentioned challenges the following problems occurred.

As mentioned above, security issues make it difficult to connect the prototype to a real operating system. Therefore, I could not evaluate the prototype at a control center.

Simulating an imaginary grid was not possible, as I had no access to data needed to model an imaginary grid. As a power grid is highly complex, and the needed data would include several complicated calculations, it is not possible to make up an imaginary grid by oneself.

In the new design there are information integrated, which do not exist in the current interface, and also not in the system in the demonstration room. At the moment, there is no technology to integrate the information in the prototype. Sources and communication channels have to be developed first.

In addition, the communication between the operating system and a user interface is highly complex. To integrate a new user interface, or even a part of it, into an operating system would imply extensive knowledge about the data and their representation. To develop the communication to an operating system, is too complex to be implemented in the scope of this thesis.

Chapter 8

Summary and Future Work

In this chapter I will summarize the work, done for this thesis, and the achieved results. Afterwards I will propose ideas, which could be pursued in future work.

8.1 Summary and Contributions

I started with an intensive research of the domain of power transmission and analyzing the working context of a control center. With a demonstration of an example I identified problems in the current user interface. I chose the contingency analysis as example task, to implement in the design, and conducted a task analysis. Additionally, I deepened my knowledge of mapping and zooming user interfaces, to better understand the usage of associations and zooming and panning technologies.

In an iterative design process of three cycles, I have developed and refined the initial design by evaluation with expert reviews. A final review with real users in a control center provided further information on the applicability of the design in a control center. Finally, I suggested possibilities to evaluate the design with real users and to compare its performance to current systems.

Till the Interface can be used in a control center, a lot of work has to be done. The feedback at the control center, especially the willingness to participate in a pilot project, proves that the concept of associations to navigate is the right approach and should be further pursued. The main parts of the design can persist, and the next steps should concentrate on integrating suitable content, implementing a secure communication to an operating system, and performing a user test under realistic conditions, possibly by transferring the design to a planning task.

8.2 Future Work

During the evaluations several issues occurred, which I had not the possibilities to pursue. In this section I will discuss some of them and introduce possible solutions.

8.2.1 Comparison of Two Nodes

In some situations it could be useful to compare the content of two nodes, e.g., two simulations. Therefore, it should be possible to focus two nodes at the same time.

With a slight reduction of the size, two nodes could easily be represented at the same time. To spare screen space, the nodes directly connected could be represented in third level detail instead of second level detail.

The remaining question is: How can I zoom in on a second node without zooming out the currently focused node? As comparing two nodes makes only sense for a few contents, the comparison could be integrated as association. A meta node 'compare' is associated to both nodes and clicking on this meta node zooms in on both nodes.

8.2.2 Extension of the Interface

Frequent changes of main concepts, while working with one interface may cause confusion and mishandling. Further extending the design beyond the example task of contingency analysis should be considered.

Therefore, it should be analyzed, if the content could be structured in a way, such that it can be visualized with nodes and associations.

In the context of extending the interface, another issue comes to mind. At some points it may be necessary to transmit data from one node to another. A way to visualize such an transmission between two associated nodes should be developed.

Appendix A

Transition between two nodes in the focus

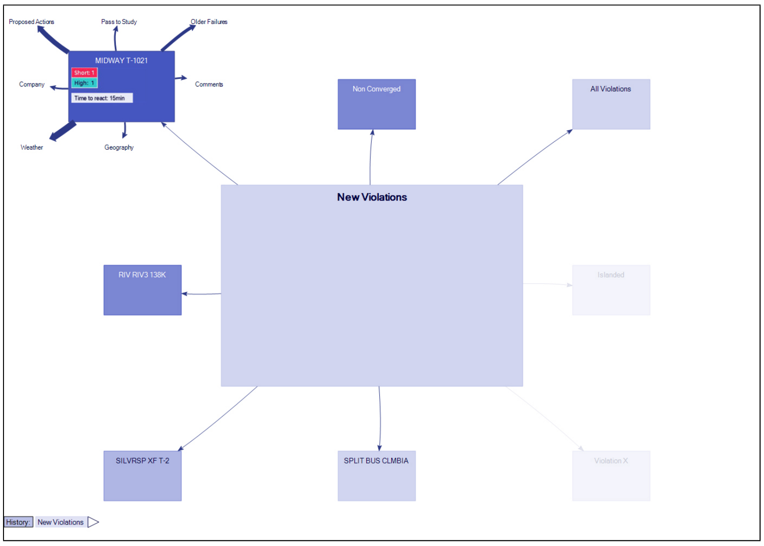

Figure A.1: Transition from the overview node in the focus to the one single contingency; 1 of 3

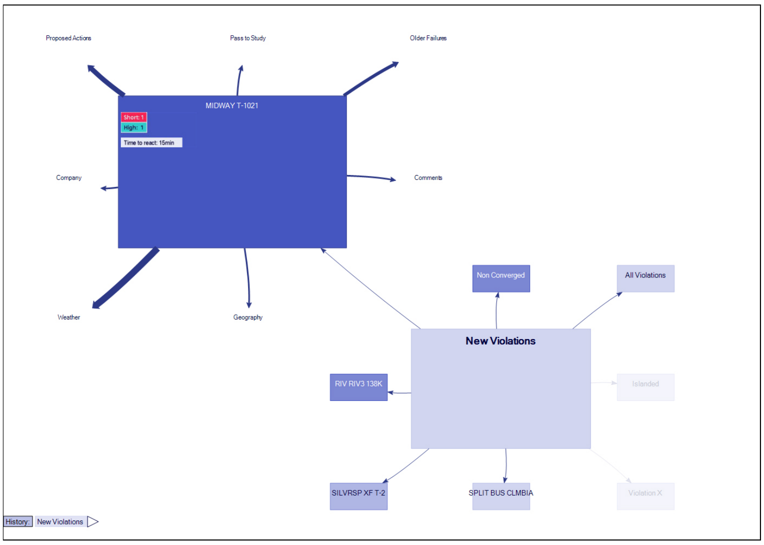

Figure A.2: Transition from the overview node in the focus to the one single contingency; 2 of 3

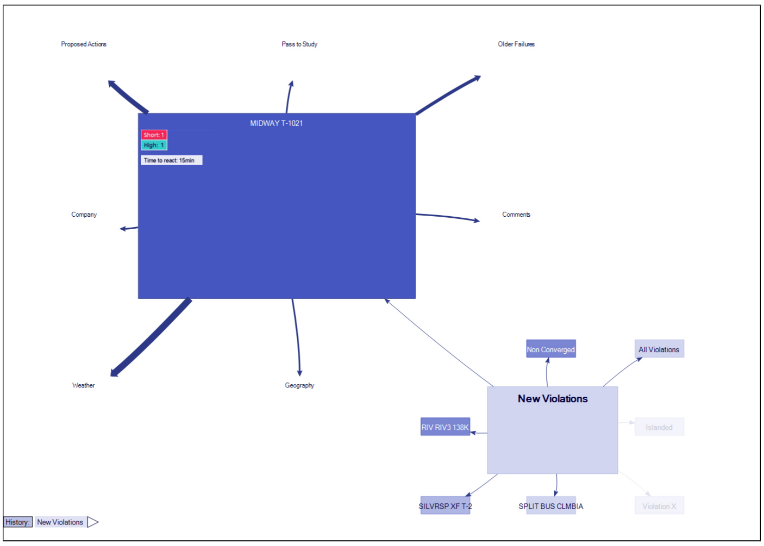

Figure A.3: Transition from the overview node in the focus to the one single contingency; 3 of 3

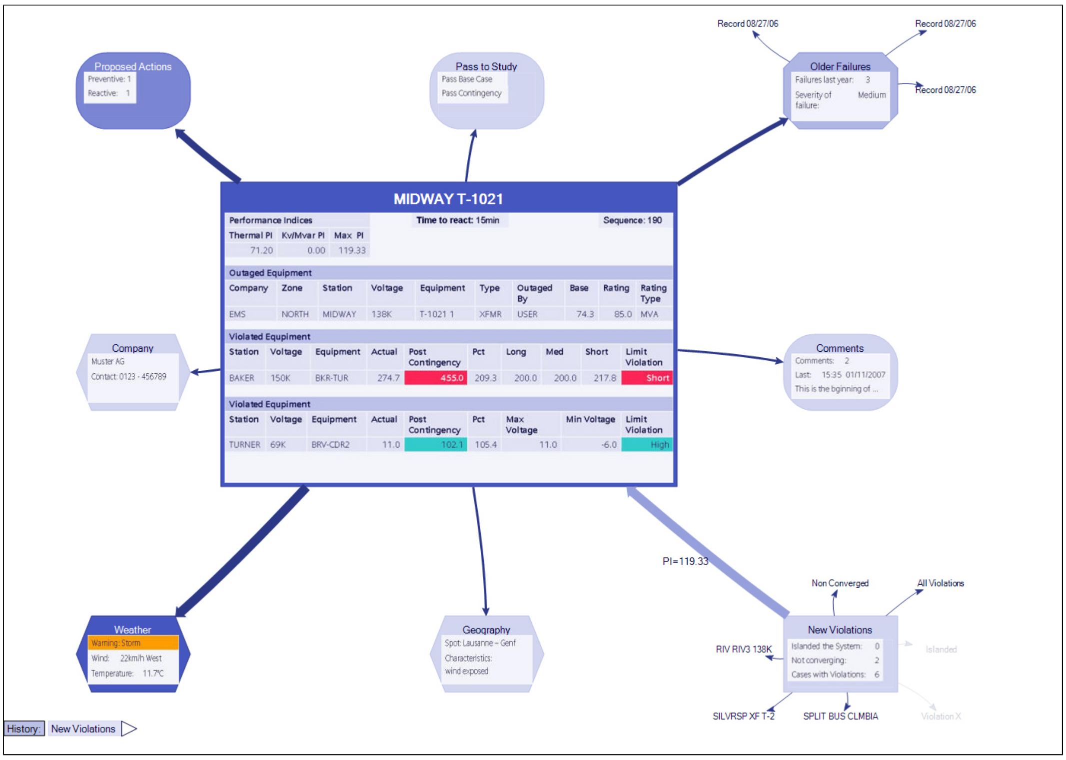

Figure A.4: A contingency in the focused node

Bibliography

Sherman R. Alpert. Comprehensive Mapping of Knowledge and Information Resources: The Case Webster. In *[Tergan and Keller, 2005]*, 2005.

Benjamin B. Bederson and James D. Hollan. Pad++: A Zooming Graphical Interface for Exploring Alternate Interface Physics. In *UIST '94: Proceedings of the 7th annual ACM symposium on User interface software and technology*, pages 17–26, New York, NY, USA, 1994. ACM. ISBN 0-89791-657-3.

David Benyon, Phil Turner, and Susan Turner. *Designing Interactive Systems. People, Activities, Contexts, Technologies*. Pearson Education Limited, Harlow, England, 2005. ISBN 0-321-11629-1.

Tony Buzan and Barry Buzan. *Das Mind Map Buch*. mvg, Landsberg am Lech, 2002.

Alberto J. Canas, Roger Carff, Greg Hill, Marco Carvalho, Marco Arguedas, Thomas C. Eskridge, James Lott, and Rodrigo Carvajal. Concept Maps: Integrating Knowledge and Information Visualization. In *[Tergan and Keller, 2005]*, 2005.

CNET Networks, Inc and Liveplasma.com. *The Big Picture*, 2005. `http://news.com.com/The+Big+Picture/2030-12_3-5843390.html` Last visited: 07/03/2007.

Donald Dansereau. Node-Link Mapping Principles for Visualizing Knowledge. In *[Tergan and Keller, 2005]*, 2005.

Alan Dix, Janet Finley, Gragory Abowd, and Russell Beale. *Human-Computer Interaction*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1997. ISBN 0-13-437211-5.

George W. Furnas. Generalized Fisheye Views. In *CHI '86: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 16–23, New York, NY, USA, 1986. ACM Press. ISBN 0-89791-180-6.

Julie Goldstein. Concept Mapping, Mind Mapping and Creativity: Documenting the Creative Process for Computer Animators. volume 35, pages 32–35, New York, NY, USA, 2001. ACM.

Lance Good and Ben Bederson. CounterPoint: Creating Jazzy Interactive Presentations. Technical report, HCIL, 2001.

Lance Good and Ben Bederson. Zoomable User Interface as Medium for Slide Show Presentations. volume 1, pages 35–49, 2002.

Marilyn Hanf. Mapping: A Technique for Translating Reading into Thinking. volume 14, pages 225–238, 1971.

David Holman, Predrag Stojadinović, Thorsten Karrer, and Jan Borchers. Fly: An Organic Presentation Tool. In *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*, pages 863–868, New York, NY, USA, 2006. ACM. ISBN 1-59593-298-4.

Kasper Hornbæk, Benjamin Bederson, and Catherine Plaisant. Navigation Patterns and Usability of Zoomable User Interfaces With and Without an Overview. volume 9, pages 362–389, New York, NY, USA, 2002. ACM.

Takeo Igarashi and Ken Hinckley. Speed-dependent Automatic Zooming for Browsing Large Documents. In *UIST '00: Proceedings of the 13th annual ACM symposium on User interface software and technology*, pages 139–148, New York, NY, USA, 2000. ACM. ISBN 1-58113-212-3.

Heinz Mandl and Frank Fischer. Mapping-Techniken und Begriffsnetze in Lern und Kooperationsprozessen. In *[Mandl and Fischer, 2001b]*, 2001a.

Heinz Mandl and Frank Fischer. *Wissen sichtbar machen. Wissensmanagement mit Mapping-Techniken*. Hogrefe, Göttingen, 2001b.

Jakob Nielsen. Iterative User Interface Design. volume 26, pages 32–41, Novembre 1993. `http://www.useit.com/papers/iterative_design/`.

Joseph Novak and Alberto Canas. The Theory Underlying Concept Maps and How to Construct Them. Technical report, IHMC, 2006.

Ken Perlin and David Fox. Pad: An Alternative Approach to the Computer Interface. 1993.

Jef Raskin. *The Humane Interface*. Addison Wesley, Boston, MA, USA, 2000.

Siemens AG. Site Visit Protocol Switzerland. internal document, 2006a.

Siemens AG. Site Visit Protocol USA. internal document, 2006b.

Siemens AG. Gedächtnisprotokoll UI-Prototyp Vorstellung bei Kunden. internal document, 2007.

Robert Spence and Mark Apperley. Data base navigation: an office environment for the professional. pages 333–340, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc. ISBN 1-55860-533-9.

Bongwon Suh and Benjamin Bederson. OZONE: A Zoomable Interface for Navigating Ontology. Technical report, UM Computer Science Department, UMIACS, 2001.

Yan Sun and Thomas J.Overbye. Visualizations for Power System Contingency Analysis Data. volume 19, pages 1859–1866, November 2004.

Sigmar-Olaf Tergan and Tanja Keller. *Knowledge and Information Visualization; Searching for Synergies*. Number LNCS 3426 in Lecture Notes in Computer Science. Springer, Berlin, Germany, 2005. ISBN 3-540-26921-5.

TheBrain Technologies LP. *Personal Brain*, 2005. `http://www.thebrain.com/#-47` Last visited: 07/03/2007.

Douglas K. van Duyne, James A. Landay, and Jason I. Hong. *The Design of Sites*. Addison Wesley, Boston, MA, USA, February 2003. ISBN 0-201-72149-X.

Alexandra von Meier. *Electric Power Systems*. Wiley Interscience, Hoboken, NJ, 2006.

Colin Ware. *Information Visualization*. The Morgan Kaufmann series in interactive technologies. Kaufmann, San Francisco, CA, 2000. ISBN 1-55860-511-8.

Allison Woodruff, James Landay, and Michael Stonebraker. Constant information density in zoomable interfaces. In *AVI '98: Proceedings of the working conference on Advanced visual interfaces*, pages 57–65, New York, NY, USA, 1998a. ACM.

Allison Woodruff, James Landay, and Michael Stonebraker. Goal-directed zoom. In *CHI '98: CHI 98 conference summary on Human factors in computing systems*, pages 305–306, New York, NY, USA, 1998b. ACM. ISBN 1-58113-028-7.

Index

Typeset March 25, 2008