Title	Creativity Assistants and Social Influences in KJ-Method Creativity Support Groupware
Author(s)	VIRIYAYUDHAKORN, Kobkrit
Citation	
Issue Date	2013-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/11339
Rights	
Description	Supervisor:Susumu Kunifuji, 知識科学研究科, 博士

Creativity Assistants and Social Influences in KJ-Method Creativity Support Groupware

Kobkrit Viriyayudhakorn

Japan Advanced Institute of Science and Technology

This page is intentionally left blank.

Creativity Assistants and Social Influences in KJ-Method Creativity Support Groupware

by

Kobkrit Viriyayudhakorn

Submitted to
Japan Advanced Institute of Science and Technology
in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy

Supervisor: Professor Susumu Kunifuji

School of Knowledge Science
Japan Advanced Institute of Science and Technology

March 2013

This page is intentionally left blank.

Dedication

To my respected parents, my beloved wife, and my son.

This page is intentionally left blank.

Abstract

Creativity support and groupware is a multidisciplinary endeavor and is the combination and intersection of the fields of creativity support systems and computer-supported cooperative work. The goal of creativity support groupware is to understand and construct an efficient groupware to maximize the creative power of a group of users through excellent groupware design and various creativity promoting techniques.

With the popularity and excellence of the KJ-Method creativity technique, we developed the Gugeek KJ-Method creativity support groupware to enable people, regardless of their location or time constraints, to perform group KJ-Method activities on the largest computer network nowadays, the Internet. Gugeek is a Web application developed by JavaScript and modern Web development technology, which allows easy access from any personal computer and mobile phone. Gugeek can store and organize the discovered ideas and guide a group of users to think systematically by following the framework workflow of the KJ-Method creativity technique. For pursuing the next level of support, the Gugeek groupware enhances team creativity performance by using various creativity assistant modules in both divergent and convergent thinking.

Creativity assistants proposed in this dissertation support the user at the generative level, which can automatically suggest relevant information to promote the creative thinking performance of users. Creativity assistants read all information about the problem topic from the input information (the previous discovered ideas) and external knowledge-base, generate helpful suggestions, and show these to the user on the groupware screen.

For enhancing the divergent thinking capability of a user, we proposed a divergent thinking assistant module based on the associative search engine GETA queried through the Wikipedia knowledge base. The GETA can be dynamically computed to discover relevant information associated with the problem topic. We found that the proposed divergent thinking assistant modules can promote idea association performance in divergent thinking to around 34% in the number of useful ideas, and can improve the number of perspectives and original ideas to around 19%.

We also proposed a novel technique for enhancing the convergent thinking capability of users, as well as automatic category naming based on the knowledge-based automatic topic identification on the general ontology such as the YAGO, and WordNet. Given an untitled category in a KJ-Chart, the knowledge-based automatic topic identification computationally selects a concept in these ontologies that covers the majority of the concepts in that category, with the highest specificity being the title of that category. We found our proposed method and the existing topic selection techniques can automatically predict the category title of the untitled group of labels at around 44% and 49% for noun and verb topics respectively.

Another perspective that affects the efficiency of the proposed KJ-Method groupware is the social influences among team members during the groupware use. Since these have a strong impact on team creativity, they should be carefully considered. In this dissertation, we focused on the two major social factors of team communication and characteristics. The team communication method is a way to communicate with team members. It is intensively used during the team discussion. The team characteristics are traits of a team derived from characteristics of individual members. They are the basis of team creativity and collaboration.

For team communication, we found that the proposed chart interaction method, which is a unique way to communicate with team members via the Gugeek creativity support groupware, is the most suitable method for the KJ-Method creativity support groupware since it is best at promoting team creativity. We also performed experiments to understand the formation of team members who contribute to the team characteristics and yield the highest team creativity. We discovered specific rules and patterns in selecting team members for building a creative team to maximize the highest creativity performance during the groupware use.

According to this discovered knowledge, the efficiency of the KJ-Method creativity support groupware can be dramatically improved by the creativity assistant modules and excellent groupware design, which induces positive social influences on team creativity. The high performance KJ-Method creativity support groupware with generative support modules can significantly promote creative thinking capability, regardless of problem topics and complexities with the help of high-performance computers. It can promote the creative thinking processes of all human users, which can significantly accelerate the innovative and scientific advancements for all humankind in the near future.

Acknowledgments

I wish to express my sincere gratitude to my principal advisor, Professor Dr. Susumu Kunifuji of the Japan Advanced Institute of Science and Technology. I am tremendously grateful to be able to work under the excellent supervision of Professor Dr. Susumu Kunifuji, who always encourages, gives kind guidance and provides unlimited support throughout the duration of my Ph.D. research in JAIST. I wish to give my grateful thanks to my co-advisor, Associate Professor Dr. Tsutomu Fujinami, for his valuable efforts and cooperation.

I would like to thank my minor research supervisor, Professor Dr. Mizuhito Ogawa of the Japan Advanced Institute of Science and Technology, for his helpful guidelines, discussions and suggestions. I also wish to express my sincere thanks to Mr. Osamu Mimura of the Mimura Creative Technology Institute for educating me intensively in the original KJ Method. I would like to give my grateful thanks to Mr. Takeo Higuchi of the Idea-Marathon Institute for introducing me to the world of ideas and creativity.

I would like to express my gratitude to all who made suggestions or influenced me during my research work. Professor Dr. Jun Munemori of Wakayama University, Professor Dr. Takaya Yuizono of the Japan Advanced Institute of Science and Technology, Associate Professor Dr. Motoki Miura of the Kyushu Institute of Technology, Assistant Professor Dr. Takahiro Kawaji of the Ohtsuki City College, and Dr. Ujjwal Neupane, who inspired me through their constant activities in the creativity support system field, and gave me valuable suggestions and kind encouragement.

I would like to show my gratitude to Assistant Professor Dr. Tessai Hayama and Assistant Professor Dr. Shohei Hidaka for their kind guidance on research work and support. In particular, the Writing Group Meeting directed by Assistant Professor Dr. Shohei Hidaka gave me motivation in my research work, writing research papers, and this dissertation.

I devote my sincere thanks and appreciation to all of my colleagues, Dr. Md. Shiful Islam, Ms. Wulanqiqige, Ms. Cheng Hui, and Mr. Mohammed Irfan. I wish to show my gratitude to my friends for helping in my lengthy experiments and giving a lot of support. Mr. Konlakorn Wongkasetchai, Ms. Arunee Ratikarn, Mr. Apimuk Muengkasem, Mr. John QJ. Quest, Mr. Chaianun Damrongrat, Mr. Ponprud Doungthaphet, Mr. Kittikorn Ariyasuk, Ms. Lalitta Suwantaweekul, Mr. Teerat Namchaisawatwong, Ms. Porawee Jitpratak, Mr. Pakorn Techaveerapong, Mr. Kriangkrai Traichaiyaporn, Ms. Patanan Ariyakornwijit, Mr. Krittichai Chaiseeda, and all Thai, Chinese, and Japanese friends.

I wish to express my overwhelming sincere gratitude to my parents, Mr. Somchai Viriyayudhakorn and Ms. Vannee Viriyayudhakorn, who kindly support me in all I do. Most importantly, my heartfelt wishes and special thanks to my beloved wife, Ms. Lin Xiaohui, who gave me kind encouragement, nurturing our son, and gave precious support every day for the past three years. Love and noteworthy thanks go to my son, Mr. Chanthanit Viriyayudhakorn (nickname: JAIST), who is my heart and the ultimate meaning of my life.

Finally, I wish to express my sincere thanks to all JAIST staff for their cooperation in different ways to make this dissertation successful.

Contents

Abstract **i**

Acknowledgments **iii**

I Introduction **1**

1 Introduction **2**
1.1 Introduction and Background 2
1.2 Objectives of this Dissertation 4
1.3 Researches Methodologies and Originalities 5
1.4 Organization of this Dissertation 6

2 KJ-Method **8**
2.1 Introduction 8
2.2 Requirements 8
2.3 Solo KJ-Method Process 9
2.3.1 Four Basic Steps 10
2.3.2 Analogy to the Reconstruction of Ancient Shards of Earthenware . . . 12
2.4 Group KJ-Method Process 12
2.4.1 Label Making Step 13
2.4.2 Label Grouping Step 13
2.4.3 Chart Making Step 14
2.4.4 Explanation Step 14
2.5 Benefits and Applications 14
2.6 Conclusion 16

3 Literature Reviews: Previously Proposed KJ-Method Systems and Groupware **17**
3.1 Introduction 17
3.2 Previously Proposed KJ-Method Support Systems 19
3.2.1 KJ Editor 19
3.2.2 D-ABDUCTOR 19
3.2.3 Group KJ Method Support System Utilizing Digital Pens: GKJ 20
3.3 Previously Proposed KJ-Method Groupwares 22
3.3.1 Groupware for a New idea Generation Consistent Support System: GUN-GEN 23
3.3.2 Group Idea Processing System : GrIPS 27

3.3.3 GRAPE: Groupware for Acquiring, Processing, and Evaluating Knowledge . . . 30
3.4 Conclusion . . . 31

4 Gugeek: Web-based KJ-Method Creativity Support Groupware 33
4.1 Introduction . . . 33
4.2 Eureka!: Prototype Divergent Thinking Support System . . . 34
4.2.1 Divergent Thinking Support System . . . 34
4.2.2 Structure of System . . . 36
4.2.3 Workflow of System . . . 36
4.2.4 User Interface . . . 37
4.2.5 Comparison between the Eureka! and the Previously Proposed Divergent Thinking Support Systems . . . 39
4.3 Gugeek: Realtime Multi-User KJ-Method Groupware . . . 39
4.3.1 KJ-Method Groupware . . . 39
4.3.2 Groupware Architecture . . . 41
4.3.3 User Interface . . . 42
4.3.4 Workflow of Gugeek . . . 49
4.3.5 Comparison Between Gugeek and Previously Proposed KJ-Method Groupware . . . 51
4.4 Key Issues of Groupware and Interface Designs . . . 51
4.4.1 Group Process . . . 51
4.4.2 Static and Dynamic Parts . . . 51
4.4.3 Group Process Models . . . 55
4.4.4 Group Communication . . . 56
4.4.5 Concurrency control . . . 57
4.4.6 Role of Group Members . . . 61
4.4.7 Management of Group Interaction . . . 62
4.4.8 Management of Share Context . . . 62
4.5 Qualitative Evaluations of the Gugeek Design . . . 66
4.6 Conclusion . . . 68

II Creativity Assistants in KJ Method Groupware 69

5 Divergent Thinking Support Modules 70
5.1 Introduction . . . 70
5.2 Structured Documents . . . 71
5.2.1 Informative Entity Extraction . . . 72
5.3 Association Search . . . 73
5.3.1 GETA - Generic Engine for Transposable Association Computation . . . 74
5.4 Divergent Thinking Support Modules . . . 78
5.4.1 Knowledge-base . . . 78
5.4.2 Four Divergent Thinking Support Engines . . . 78
5.5 Experimental Setting . . . 79
5.5.1 Evaluation Methods . . . 79
5.5.2 Procedure of Experiment . . . 81

5.6 Experimental Result and Observation . . . 81
5.6.1 Quantitative Result . . . 81
5.6.2 Qualitative Result . . . 85
5.7 Conclusion . . . 87

6 Convergent Thinking Support Modules 90
6.1 Introduction . . . 90
6.2 Related Works . . . 91
6.3 Idea Chart and Knowledge Bases . . . 92
6.3.1 Structure of Idea Chart . . . 92
6.3.2 Knowledge Bases . . . 93
6.4 Automatic Topic Identification . . . 94
6.4.1 Extraction of Nouns and Verbs . . . 94
6.4.2 Mapping of Words to Concepts . . . 95
6.4.3 Hierarchical Hypernym Graph Construction . . . 96
6.4.4 Topic Selection Methods . . . 96
6.4.5 More Non-overlapping Topics . . . 99
6.5 Usage in Automatic Category Naming . . . 100
6.6 Empirical Evaluations . . . 100
6.6.1 KJ Method . . . 103
6.6.2 Mind Mapping . . . 105
6.7 Discussion . . . 106
6.7.1 Criticism . . . 106
6.8 Conclusion . . . 111

III Social Influences in KJ Method Groupware 112

7 Team Communications 113
7.1 Introduction . . . 113
7.2 Team Communication Methods . . . 114
7.3 Experimental Settings . . . 114
7.3.1 Experimental Procedure . . . 114
7.3.2 Quantitative Evaluation . . . 116
7.3.3 Qualitative Evaluation . . . 117
7.4 Experimental Result . . . 117
7.4.1 Quantitative Result . . . 118
7.4.2 Qualitative Result . . . 122
7.5 Discussion . . . 124
7.6 Conclusion . . . 124

8 Team Characteristics 127
8.1 Introduction . . . 127
8.2 Team Characteristics . . . 128
8.2.1 Background Knowledge . . . 128
8.2.2 Difference of Background Knowledge . . . 128
8.2.3 Friendship Closeness . . . 128
8.2.4 Difference of Friendship Closeness . . . 128

8.3 Control Variables 129
8.3.1 Team Communication Methods 129
8.3.2 Problem types 129
8.4 Experimental Settings 129
8.4.1 Experimental Procedure 129
8.4.2 Team Creativity Evaluation 130
8.4.3 Team Characteristic Evaluation 131
8.4.4 Control Variables Evaluation 131
8.5 Experimental Result 133
8.6 Conclusion 136

IV Conclusion **139**

9 Summary and Conclusion **140**
9.1 Summary 140
9.1.1 Creativity Assistants 140
9.1.2 Social Influences 142
9.2 Conclusion 142
9.3 Future Research 144

Publications **146**

Biblography **147**

List of Figures

1.1 Organization of this dissertation 6

2.1 A sample of the whole KJ Method process 9
2.2 A sample KJ Chart (contrary to the KJ Method learning material of Osamu Mimura[1], a professional KJ Method instructor). 10
2.3 Recommended relationship symbols in chart making 12
2.4 An analogy of the KJ-Method process with the reconstruction of ancient shards of earthenware (contrary to the KJ Method learning material of Osamu Mimura [1], a professional KJ Method instructor). 13
2.5 W-shaped of problem solving model 15

3.1 Two superimposed screen of the KJ-Editor, universal and local screens 20
3.2 The screenshot of D-ABDUCTOR 21
3.3 Final KJ-Chart captured by Anoto pens in the traditional group KJ-Method session constructed by the GKJ application 22
3.4 A GUNGEN SPIRAL II screenshot 23
3.5 GUNGEN's conceptual model for distributed and cooperative KJ-Method support system [2]. 24
3.6 GUNGEN's initial screen [3]. 25
3.7 GUNGEN's electronic tag on the label grouping step [3]. 25
3.8 Wadaman screen in GUNGEN system[3]. 26
3.9 GUNGEN's linking [2]. 27
3.10 KJWare user interface in GUNGEN 28
3.11 A GUNGEN DX II screenshot and functionality 28
3.12 The screenshot of GrIPS 29
3.13 The screenshot of GRAPE (1) 32
3.14 The screenshot of GRAPE (2) 32

4.1 A Eureka! screenshot and functions 34
4.2 The workflow of a divergent thinking support system 37
4.3 Architecture of Gugeek groupware 41
4.4 A Gugeek screenshot and description of functions 43
4.5 The User Interface of Labels and Categories in Gugeek 44
4.6 Idea labels and category operations (No. 1) 45
4.7 Idea labels and category operations (No. 2) 46
4.8 Idea labels and category operations (No. 3) 47
4.9 Minimap and tele-pointing system 49
4.10 Index Screen of Gugeek Groupware 50
4.11 The KJ-Method group process 52

4.12 The group process model of the KJ-Method groupware 55
4.13 Group Interaction Management: Comb Model 62
4.14 The questionnaire for evaluating the quality of Gugeek groupware. 67

5.1 An example structure of target knowledge-bases. 72
5.2 The associative information in a Wikipedia article and its source code 72
5.3 Four similarity measures and their formulas 76
5.4 An example of WAM and sample results. 77
5.5 The location of the suggestions in the Eureka! screen 82
5.6 Quantitative result: number of ideas (per user per KJ-Chart) 86
5.7 Quantitative result: fluency of ideas (per user per KJ-Chart) 87
5.8 Quantitative result: flexibility of ideas (per user per KJ-Chart) 88
5.9 Quantitative result: originality of ideas (per user per KJ-Chart) 89

6.1 A sample idea chart (in the KJ Method program). 92
6.2 Tree representation of the sample idea chart (in the mind map form). 93
6.3 In the upper half, the noun hierarchical hypernym graph $HHG_{\mathrm{in}(h)\cup\mathrm{out}(h)}$ where h is the concept "???? (G)" is shown. The solid line graph is the hypernym tree $HT_{\mathrm{in}(h)\cup\mathrm{out}(h)}$. The dashed line graph is the word to concept graph $CG_{\mathrm{in}(h)\cup\mathrm{out}(h)}$. In the lower half, the tree representation of the idea graph is displayed on the left hand side. The diagram on the right hand side shows the result of the input extraction $\mathrm{in}(h)$ and $\mathrm{out(h)}$. The numbers in parentheses are the values of variables and six topic selection measures corresponding to each concept where they are located. The order of variables is shown at the legend box located at the top-right corner of this figure. 97
6.4 The pruned hierarchical hypernym graph (original is Fig 2) used to extract the second topic. 101
6.5 A screen of Gugeek groupware when the creativity assistants are activated. . . . 102
6.6 Ranking screen during the automatic category naming experiment 104
6.7 Difficulties found in the proposed convergent thinking support. 107

7.1 Text chat box located at the right sidebar of the Gugeek screen. 114
7.2 Team members perform the KJ-Chart and discuss the problem by using Chart Interaction and Face-to-Face (FF) communication methods during an experiment 115
7.3 Quantitative result: fluency of ideas (per user per KJ-Chart) 118
7.4 Quantitative result: original of ideas (per user per KJ-Chart) 119
7.5 Quantitative result: flexibility of ideas (per user per KJ-Chart) 119
7.6 Quantitative result: elaboration of ideas (per user per KJ-Chart) 120
7.7 Quantitative result: category fluency (per user per KJ-Chart) 120
7.8 Quantitative result: within-category fluency (per user per KJ-Chart) 121
7.9 Quantitative result: average depth of top categories (per user per KJ-Chart) . . 121
7.10 Qualitative result: participant cross-evaluation in divergent thinking phase (per user per KJ-Chart) 122
7.11 Qualitative result: participant cross-evaluation in divergent thinking phase (per user per KJ-Chart) 123
7.12 Questionnaire about the effectiveness of team communication method 125

8.1 Questionnaire for deriving team characteristics 132

9.1 A KJ-Chart considering future research for KJ-Method groupware. 145

List of Tables

4.1 Comparison between three divergent thinking support systems 39
4.2 Comparison among four KJ-Method Groupware 53

5.1 A log of ideas during the experiment. 80
5.2 A list of viewpoints during the experiment. 80
5.3 A list of idea viewpoints of Topic 1-5. 83
5.4 Topic and engine assignment 84
5.5 Means and Standard Deviations of Quantitative Evaluation Results (per user per time period) 84

6.1 Rank results for all topic selection methods except the RB in the KJ Method experiment (the lower is the better) 104
6.2 Suggestion Accuracy in the KJ Method experiment 104
6.3 Average word similarity score [0,1] between predicted topics and actual topics in the KJ Method experiment $(\times 10^{-2})$ 108
6.4 Suggestion Accuracy in the Mind Mapping Experiment 109
6.5 Average word similarity score [0,1] between predicted topics and actual topics in the Mind Mapping Experiment $(\times 10^{-2})$ 110

7.1 Group communication methods and problems types assignment. 116
7.2 Qualitative results on the effectiveness of three communication methods. 124

8.1 Communication methods and problem type assignment 130
8.2 Descriptive statistics of all variables in this study. 133
8.3 Bivariate correlations and significant test of four divergent thinking measures, three convergent thinking measures, four team characteristics, and two control variables. 134
8.4 Regression analysis on the divergent and convergent thinking performances: base model 137
8.5 Regression analysis on the divergent and convergent thinking performances: including four team characteristic factors. 138

Part I

Introduction

Chapter 1

Introduction

1.1 Introduction and Background

"To raise new questions, new possibilities, to regard old problems from a new angle, requires creative imagination and marks real advance in science."

– Albert Einstein

A great innovation is rooted in great creativity. Creativity is the invention or origination of any new valuable thing. It is a process of producing something that is both original and worthwhile [4], and is one of the hardest human intelligence abilities.

With the emergence of information processing technology, the dream of a computing machine which can think creatively as a human is becoming closer. Artificial Intelligence (AI), a computer research field that creates, designs, and implements computer systems to act and think as a human, seems to be the closest way for pursuing this dream. Even though the success in the artificial intelligence field is clearly and recently visible in many applications and research fields such as the development of understanding and responding to natural language processing, machine learning, deductive reasoning, automated planning, problem solving, and decision making, no claim was made that a computer has been able to "think" autonomously as a human. The difficulty in making such a claim is that it is hard to determine whether or not a machine needs to be conscious or self-aware in order to be humanly thinking.

This controversial issue is a core difficulty in pursuing autonomous creativity. A crucial question arises: "Can a computer be creative or is creativity inseparable from the human mind?" [5]. To answer this question, better understanding about the creative process in a human brain and limitation of computability is needed [5]. This knowledge has been researched and has not been understood enough to answer such a question. In 1950, Turing proposed a popular AI approach that does not mimic the human thinking process to make computers intelligent, but merely exhibits intelligent behavior, indistinguishable from human output in comparable conditions (called Turing test [6]).

Turings AI approach for developing a machine to exhibit creative behavior is called "computational creativity" research (also known as artificial creativity). The goal of computational creativity is to model, simulate, or replicate creativity by machines for constructing a program capable of human-level creativity [7]. Successful computational creativity software, exhibiting

an acceptable degree of the creativity humans possess, can replace humans, who are the original agents responsible for creativity to innovate and develop new knowledge. It can dominate human creativity by its accuracy, high computation power, and ultra-speed performance.

Beside replacing human creativity by machines, the creativity support system, which is a research field concerned with the supportability of human creativity thinking by a computer system, is another interesting dimension for enhancing human creativity by machines without necessarily machines being creative themselves. The creativity support system is not extraordinary AI techniques that can fully automate the production of a complete creative work [5], but merely support users by encouraging exploration and enhancing collaboration to maximize their creativity performance [8]. The creativity support system still needs humans to operate it, unlike computational creativity. The creativity support system involves the storage, analysis, and representation of ideas in a form that facilitates human interaction to aid creativity.

With the limitation of present-day technologies and current research progress of creativity in the human brain, the creativity support system provides more benefits, practical support, and useful applications than computational creativity, which is waiting for future technologies. The creativity support system is gaining valuable attention from several interested parties, especially in competitive creative economies.

With the emergence of groupware technology, creativity support groupware, a kind of creativity support system aimed at supporting a group of users to collaborate based on group creativity techniques, has been recently invented. It simultaneously supports many users who are remotely located from each other through the computer network and enables users to perform group creativity techniques synchronously or asynchronously over geographic distances by providing tools that aid collaboration, communication, and the process of group creativity techniques.

The KJ-Method [9] is a popular data organizing method invented by JIRO Kawakita in 1951 and is still popular, especially in Japan, to the present day. It is an excellent, ready-made formula for organizing vague information into understandable knowledge, which helps to unbiasedly discover the truth from the observed information. Many top-level companies and organizations in Japan have proved the KJ Method as an efficient tool for organizing and prioritizing tasks, problems, and ideas, etc. Even though performing the KJ Method needs few resources, people can still be discouraged from performing it. To provide a more convenient way for performing the group KJ-Method and eliminate some of their requirements, i.e. gathering participants into the same time and place, wasting a lot of stationery, and hand copying a KJ-Chart to distribute it among participants after a meeting, KJ-Method creativity support groupware was invented.

Constructing an efficient KJ-Method creativity support groupware requires much consideration. Creativity support groupware contains two main perspectives.

The first perspective is the design of the groupware itself (creativity support groupware). The following are components of typical groupware [10], which requires careful consideration to encourage team creativity, which is the objective of creativity support groupware.

- **Communication Support.** Interactions among team members to facilitate team creativity, but investing too much time in communication can degrades individual creativity.

- **Shared Workspace.** How to make a shared workspace suitable for a collaborative task (a group creativity technique)?

- **Shared Information.** Security for preventing leakage of team secrets to outsiders must be carefully implemented.

- **Workflow Management** Workflow controls helps in gaining consensus before a deadline. How can workflow control be enforced without damaging creativity?

The tasks do not end with only considering groupware designs. In any groupware, social influences, which are influences from the group working environment on team creativity occurring during the groupware use, must be carefully considered. A social influence can be either positive or negative. It either helps team creativity or damages team creativity. The way to avoid negative social impact and obtain positive social impact should be integrated in the design of the groupware.

In addition to researching KJ-Method groupware designs for maximizing team creativity, creativity assistants or creativity support (creativity support groupware) is the second perspective that needs to be considered. In recent research in the creativity support system, a number of intelligent information processing systems are proposed and developed for enhancing human creativity. Such intelligent information processing systems process a mass amount of data related to a topic or a problem under consideration, wisely extracting the beneficial parts that can support creativity performance, and present it to users through groupware.

Creativity support groupware is a multidisciplinary endeavor that is a combination mainly of both Creativity Support System (CSS) and Computer-Supported Cooperative Work (CSCW). The objective of creativity support groupware is to understand and construct an efficient groupware for maximizing the creative power of a group of users through the excellent groupware designs and the various creativity promoting techniques.

1.2 Objectives of this Dissertation

The main objective of the dissertation is to build and invent a high-performance KJ-Method creativity support groupware. The term "high performance" means that creativity support groupware must have functions to support a group of users to maximize their team creativity. It leads us to three sub-objectives as follows:

1. The proposed groupware must be well designed for allowing the group of users to perform the group KJ-Method creativity technique regardless of location and time constraints. It should visualize a KJ-Method procedure workflow and track a current stage of group activity for reminding users.

2. The proposed groupware should provide artificial intelligence modules that automatically suggest some information or automatically support users to perform KJ-Method processes for promoting both divergent and convergent thinking.

3. The design of the proposed groupware should induce positive social influences among team members to increase team creativity performance.

To achieve the research objective and three sub-objectives, the following research questions categorized into three dimensions must be answered.

1. **KJ-Method Groupware Designs**

   (a) What is the KJ-Method?

   (b) What are the core philosophies, procedures, and benefits of the KJ-Method?

(c) Are there any previously proposed KJ-Method creativity support systems or groupware?

(d) What is the design for a great groupware?

(e) How to design an efficient collaborative multiuser KJ-Method creativity support groupware?

(f) How to construct an efficient groupware?

2. **Creativity Assistants**

   (a) Which artificial intelligence techniques are suitable for generating the suggestions intended to promote individual and team creativity?

   (b) How to generate associative information that potentially promotes the users idea association capabilities in the divergent thinking phase?

   (c) How to generate suitable category titles of groups of associated idea labels for supporting users in the convergent thinking phase?

   (d) How to measure the efficiency of the proposed creativity assistants? Should we measure it by quantitative evaluation or qualitative evaluation or both?

   (e) How can we evaluate the degree of creativity appearing in a KJ-Chart?

3. **Social Influences**

   (a) What social influences that significantly impact on team creativity performance?

   (b) Which type of communication method accelerates team creativity performance and is suitable for the KJ-Method groupware?

   (c) Do the social activities have a strong relationship with the performance of team creativity?

   (d) Do the traits of a team, combined with the traits of individual members, have a strong relationship with team creativity?

   (e) How can we evaluate or acquire the traits of a team?

   (f) How can we measure team creativity?

1.3 Researches Methodologies and Originalities

The three dimensions of the dissertation's objectives listed in the previous sections require different research methodology as follows.

1. **KJ-Method Groupware Designs:** The research in the KJ-Method groupware design is constructive qualitative research. A KJ-Method groupware was practically developed, tested by a set of users, and later evaluated qualitatively by questionnaires. Research for practical application (such as KJ-Method groupware) is applied research. It deals with solving the practical KJ-Method groupware design for maximizing team creativity productivity.

   The originality of this research is at the incremental level. It is multi-disciplined, fielded intensively, extended from prior knowledge, which are the knowledge of computer supported cooperative work, information technology, and the creativity support system.

2. **Creativity Assistants:** The research carried out with the creativity assistants is constructive quantitative research. Several novel algorithms in both divergent and convergent thinking are developed, then plugged into Gugeek groupware, and evaluated quantitatively by Gugeek users. The quantitative evaluation can be done by evaluating the creativeness appearing on KJ-Charts constructed by an individual or a team using Gugeek groupware. The evaluation is based on widely accepted creativity measures and statistical testing. The research of the creativity assistants is applied research aimed at yielding the practical creativity assistant module for the KJ-Method groupware.

   The originality of this research is novel. To the best of our knowledge, there is only one research work that attempts to propose an algorithm for a creativity assistant in 1994 [11].

3. **Social Influences:** The research in social influences is empirical quantitative and qualitative. A number of social factors that possibly affect team creativity performance are studies from recent research works and are set up as hypotheses. The quantitative and qualitative evaluation can be done by measuring the creativeness appearing on KJ-Charts, and the result of questionnaires respectively. This research gives better understanding in the fundamental principle of the KJ-Method groupware, thus it is fundamental (pure) research.

   There are several research works on social influence in team creativity through groupware use, including team communication, but there has not previously been research work focusing on team characteristics.

1.4 Organization of this Dissertation

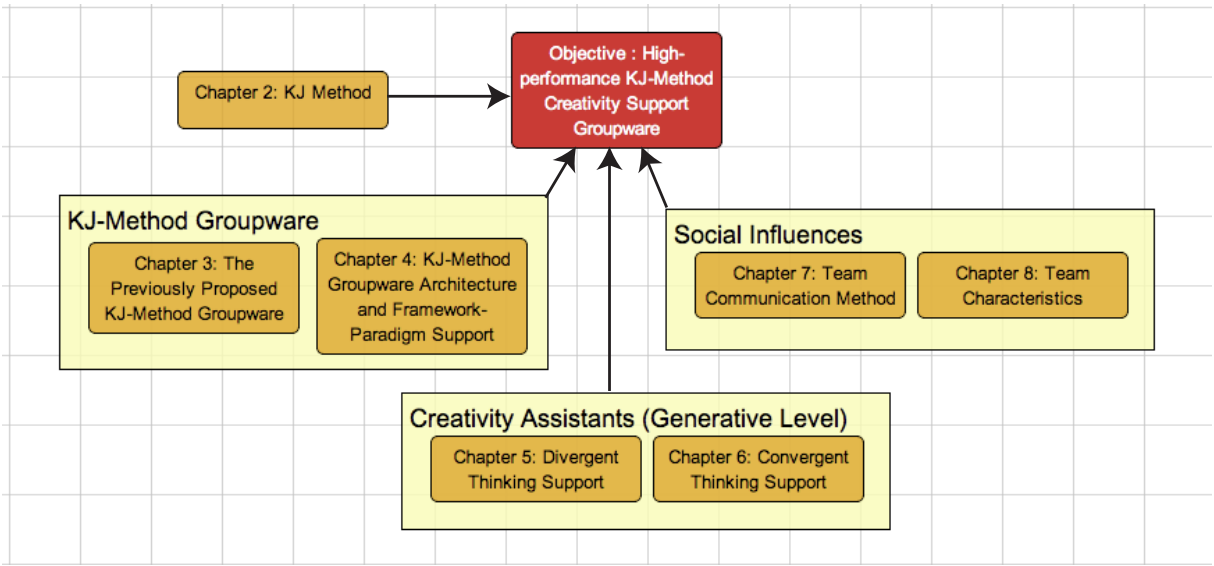


Figure 1.1: Organization of this dissertation

This dissertation is divided into four parts as shown in Figure 1.1. The first part contains two chapters, namely the introduction of the KJ-Method and the proposed KJ-Method creativity support groupware. Chapter 2 introduces the KJ-Method including its definition, basic procedures, benefits and typical application in the present day. Chapter 3 introduces the KJ-Method

creativity support groupware and previously proposed KJ-Method support systems and groupware. Chapter 4 proposes a novel KJ-Method creativity support groupware named "Gugeek", which is based on the prototype divergent thinking support system named "Eureka!" The detailed design and architecture of Gugeek groupware are thoroughly explained.

The second part contains two chapters in which the creativity assistant modules for promoting the creativity performance of users are explained. Chapter 5 describes four divergent thinking support engines proposed by authors to promote the idea association capability of users by displaying the associative information. Experimental settings and results for discovering the most efficient divergent thinking support engine are given. Chapter 6 describes the proposed automatic category naming tool, which can automatically suggest an accurate category title by reading the content of all labels appearing as an untitled category in the label grouping step of the group KJ-Method. Empirical experiments among the proposed algorithm and the existing algorithms based on the Gugeek KJ-Method groupware and mind mapping software are provided.

The third part contains two chapters, where the effect of social influences among team members on team creativity is described. Chapter 7 explains three communication methods for use among team members during the group session. Experiments for understanding the relationships between three communication methods and team creativity are presented. Chapter 8 presents four team characteristics which affect team collaboration and cooperation during the group session in the Gugeek KJ-Method groupware. The influence of four team characteristics on team creativity is examined and described.

The last part contains one chapter. Chapter 9 summarizes this dissertation, provides the conclusion and gives future interesting research in this field.

Chapter 2

KJ-Method

2.1 Introduction

The KJ-Method is a paper card system that organizes vague information into understandable knowledge, helping practitioners to unbiasedly discover the truth from data determined by observation. Since the KJ-Method is based on the truth only, free and creative thinking can be unlocked from the bias thinking barrier, which usually relies on assumptions and preconceived ideas. According to its benefits and output, the KJ-Method can be considered as a creativity technique and an intellectual production tool.

The KJ-Method was firstly conceived by Jiro KAWAKITA in 1951 [12] for establishing a method to orient cooperative works toward innovation, elaborated with the Tadao UMESAOs paper card system for intellectual productive works [3, 13]. The title of the KJ-Method was taken from the initial of Kawakitas name. The KJ-Method was developed and systematized by Kawakita during his fifteen years of field work in the post of Assistant Professor of Geography at Osaka City University [12], was proposed as the present KJ-Method in his book published in 1967 [14] and has gained popularity, especially in Japan, to this day.

This chapter will explain the descriptions, requirements, procedures, goals, and benefits of both solo and group KJ-Method creativity techniques. It will be organized as follows: Section 2.2 describes the requirements of the KJ-Method; Section 2.3 explains the descriptions, procedures, and goals of the solo KJ-Method; Section 2.4 explains the descriptions, procedures, and goals of the group KJ-Method; Section 2.5 presents the benefits of the KJ-Method; and Section 2.6 concludes the chapter.

2.2 Requirements

The KJ-Method is a solo/group activity. For a group activity, it requires a group of practitioners who have the same time availability and stay in the same location. Moreover, the stationery needed to perform the KJ-Method is as follows:

1. A large sheet of rectangular paper or a writing surface..
2. A large amount of rectangular self-adhesive labels. If these are unavailable, small rectangular labels with glue will be fine.
3. A large amount of paper clips.

4. Writing implements.

The above requirements are not needed if a KJ-Method support groupware or a similar system is used. Instead, a personal computer with a usable network connection is needed.

The undeniable requirement, regardless of the use of the support system, is a clear problem topic. Without a problem topic, the KJ-Method has no input to solve.

2.3 Solo KJ-Method Process

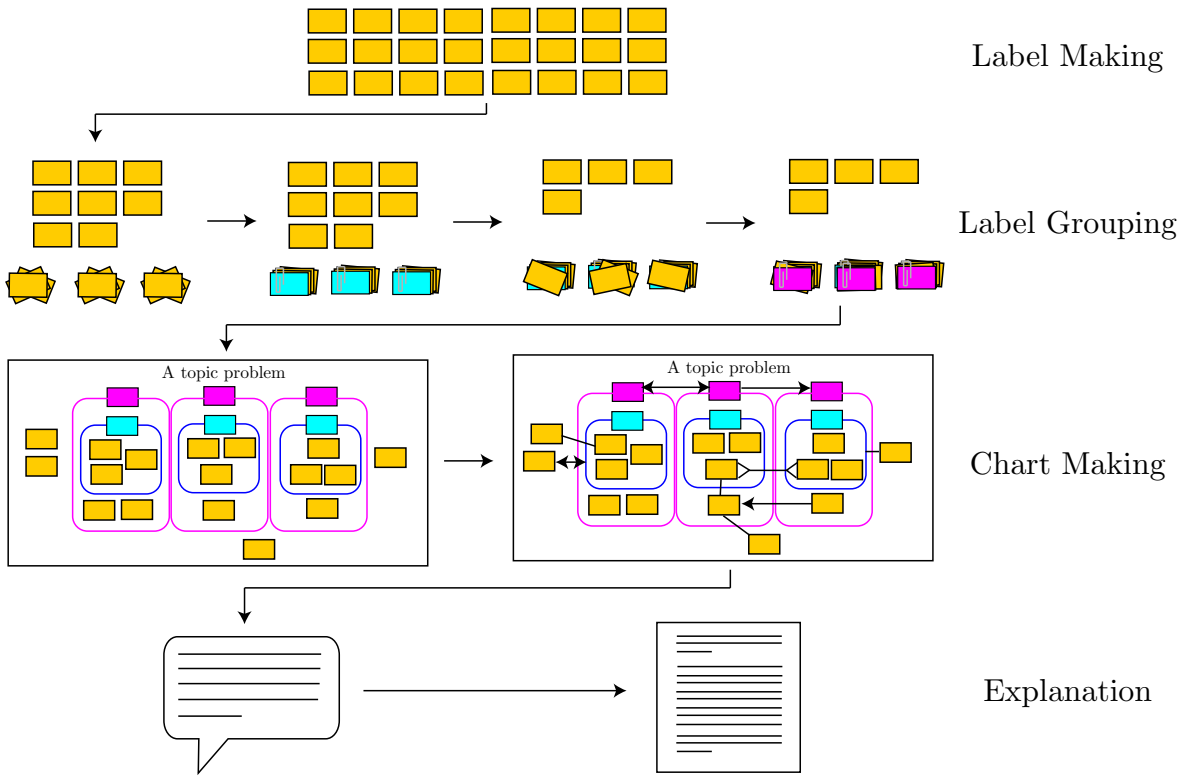


Figure 2.1: A sample of the whole KJ Method process

The KJ Method consists of four basic steps as illustrated in Figure 2.1. When the KJ Method is completed, a KJ Chart to visualize the entire structure of the problem or the topic under consideration as shown in Figure 2.2 [1], and a written explanation in a detailed report of interrelationships among elements in a KJ Chart, is obtained. The KJ-Method process aimed at deriving a KJ-Chart, which firstly needs three basic steps, is called the "A-type KJ-Method". Another KJ-Method process aimed at deriving a written explanation, and which needs all four basic steps, is called the "B-type KJ-Method". Naturally, the B-type KJ-Method is performed after the A-type has been completed [2]. The four basic steps are listed as follows [12]:

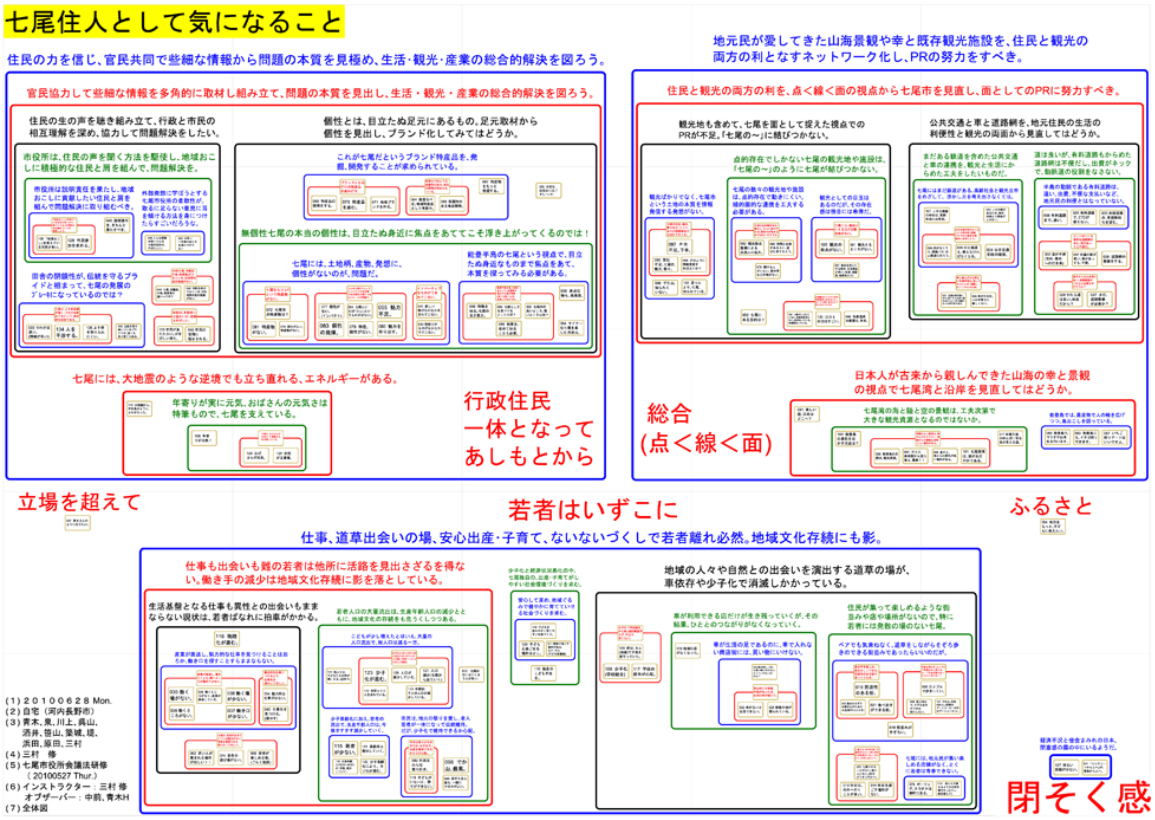


Figure 2.2: A sample KJ Chart (contrary to the KJ Method learning material of Osamu Mimura[1], a professional KJ Method instructor).

2.3.1 Four Basic Steps

Label Making Step

The practitioner collects information, observes situations related to the problem, and writes down all the discovered data on the labels. Only one fact, thought or concept related to the problem should be written on each label. Each written label must have only one “Kokorozashi” (Japanese translation: a real intention). There is no limit on the number of written labels. The more labels the better the chart. Label making is performed until the practitioners exhaust all the data necessary for solving the problem.

Label Grouping Step

After writing all the facts, the labels should be stacked, shuffled, and spread well on a wall, a table, or a floor. The practitioner must be free from any prejudices or preconceived ideas and should carefully listen to what the labels are trying to say. Some of the labels that appear to belong together should be arranged close to each other and quite far away from any other labels to make a category (island) of labels. Note that the labels should not be grouped based on similarity, but on their connection or mental association.

If there is a label that does not seem to belong with any other labels, it is called a lone wolf. Even if there are many lone wolves; practitioners should not include a lone wolf in a group unnaturally. They will be fitted into a group at a higher level of the label grouping process.

After about two-thirds of all labels are grouped, the practitioner starts to make a title for each category. The practitioner should re-read all labels in the category, and then think of a suitable title to describe the essence of all labels in that category. It should not be too abstract or too specific. A category should be carefully entitled, since it is the key component for organizing the category and discovering the relationships among elements in the chart making step. Once a title is decided, the practitioner writes it on a new label, called a title label. The practitioner should perform this process until all ungrouped labels are lone wolves.

When the label grouping procedure is finished, all labels in a category and their title label are stacked. The title label is on the top of the stack. All of them are clipped with a paper clip. This procedure is performed until all categories are clipped.

From the beginning of the label grouping step to this point, it is one iteration of the label grouping process, and can be performed repeatedly until the number of categories are less than ten [12]. At the beginning of the next iteration, categories and lone wolves will become labels. They are spread on a table, and the practitioner starts label grouping and category title naming again.

Chart Making Step

After the label grouping step, the bundle of final groups are obtained. In the chart making step, all final groups are spatially arranged on the large sheet of paper. The practitioner should consider carefully a stable spatial arrangement, which is a consistent understanding of all labels and categories. The practitioner should make an arrangement based on their "Kansei" (Japanese translation: understanding). The arrangement should allow all labels and categories to potentially express themselves. After the arrangement of the final categories is satisfied, all of their sub-groups or labels should be arranged using the same justification recursively until all labels are arranged.

The relationships between objects on the KJ-Chart such as a category-to-category, a category-to-label, a label-to-category, and a label-to-label relationship, etc. should then be expressed by four symbols [15] as shown in Figure 2.3.

- **Cause and effect, and order of occurrence:** An object is a predecessor or a cause of another object.
- **Contradiction:** Both objects are in conflict with each other.
- **Interdependence:** Both objects are required by each other to exist.
- **Correlation:** Both objects are in harmony with each other.

The relationships among the objects in the chart should be easily and clearly understood.

When this step is completed, a completed KJ-Chart, which illustrates relations between labels visually, is obtained. A KJ-Chart helps audiences to grasp the entire structure of the problem at a glance. At this point, the A-type KJ-Method is completed.

Explanation Step

The last step in the KJ-Method is to explain the chart clearly. The explanation should begin with a general scenario of the problem, and then be more specific.

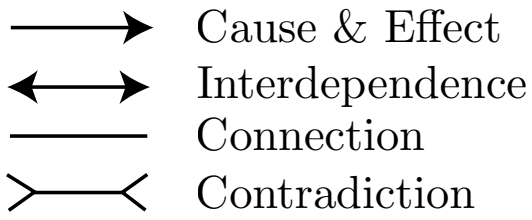


Figure 2.3: Recommended relationship symbols in chart making

Usually, the verbal explanation should be done first. The explanation could be started from any position of the chart, and then proceed to an adjacent part until all parts are explained. The verbal explanation should be revised several times until it is concrete, smooth, and concise. After that, the written explanation should be performed. The practitioner writes down the verbal explanation and makes several revisions until it is clear, smooth, and concise. The explanation step helps the audience to understand the interrelationships among components of the problem thoroughly. At this point, the B-type KJ-Method is completed.

2.3.2 Analogy to the Reconstruction of Ancient Shards of Earthenware

The KJ Method is very much similar to the reconstruction of ancient shards of earthenware. Information in the label (a shard) is a part of undiscovered knowledge (earthenware). According to the top-left subfigure of Figure 2.4, shards can be collected from the ancient site, in the the same way as pieces of information can be collected from field observations or experiments. It is identical to the label making step.

In the two bottom sub-figures, shards that seem to belong or are connected to each other (not similar) are arranged in the same group on a tray. They are connected by big metal clippers, which are visualized inside the red dotted circle in the bottom right sub-figure. They become a group of shards (a bigger piece of connected shards), which connect and are marked by a big metal clipper. This process is repeatedly performed until the completed earthenware is yielded. Similarly, the labels that seem to be connected to each other are grouped and clipped with a group headed by a paper clip. It is identical to the label grouping and chart making step, which is repeatedly performed until the undiscovered knowledge is recognized or discovered.

In the top right sub-figure, the earthenware is completely re-constructed. This is similar to the knowledge being completely discovered and ready to be explained in the explanation step.

2.4 Group KJ-Method Process

The group KJ-Method provides an additional point to consider. It is possible to have social conflict arising from the differing opinions of practitioners on borderline concepts. For the group KJ-Method, the practitioner is not allowed any discussion about the problem topics content until at the chart making step. Premature discussions often focus on borderline labels which might be not important to the problem topic. Spending any time discussing them is a waste.

The four basic steps of the group KJ-Method process have slight additions from the solo KJ-Method process as follows [16].

Analogy: the way of KJ method process and the reconstruction of ancient shards of earthenware.

(suggested and photos by Mr O. Mimura, 2007)

Figure 2.4: An analogy of the KJ-Method process with the reconstruction of ancient shards of earthenware (contrary to the KJ Method learning material of Osamu Mimura [1], a professional KJ Method instructor).

2.4.1 Label Making Step

All practitioners perform the label making step in the solo KJ-Method process simultaneously. When a label is created, each practitioner puts the label up to a wall, a table, or a floor and then read other practitioners contributions. If they think of something else that should go on the wall, they write it down on a label and put it into the collection. Label making is performed until all practitioners exhaust the data necessary for solving the problem.

2.4.2 Label Grouping Step

All practitioners perform the label grouping step in the solo KJ-Method process simultaneously. Practitioners can move labels into groups created by other people, including reviewing someone elses group. If a group created by another practitioner does not quite make sense, the practitioner can freely rearrange the labels until it does. If a group title provided by other practitioners can be improved, practitioners can freely add a group name into the group header label. Label grouping is performed until all practitioners are satisfied with the grouping result.

2.4.3 Chart Making Step

All practitioners perform the chart making step in the solo KJ-Method process simultaneously. They help each other to add the interrelationship between objects in the KJ-Chart. If any conflict occurs, they are open for discussion and democratically voted on for a final solution.

2.4.4 Explanation Step

A representative of a group of practitioners publicly performs verbal and written explanations. All practitioners can vote for the significant and proper sequence of categories and labels in the explanation. All practitioners help each other to review and improve the explanation.

2.5 Benefits and Applications

According to the psychological nature of humans, we usually make judgments based on our past experiences. In other words, people tend to see what they want to see.

The human brain is rewarded with "happiness" emotions or satisfaction if the right decision has been made. When we encounter a new problem, the brain makes a decision by looking at similar problems encountered in the past, and then follows the solution applied previously to successfully solve such problems. This decisive strategy requires less mental power and is very fast, but it is not always the right decision.

This learned experience is later induced and concretely constructed to be a belief and continuously developed to be a dogma for each person. When a problem needing to be solved is presented, these dogmas make people making the decision biased towards their assumption and preconceived ideas. These dogmas conceal the truth residing inside a problem and also prevent decision making based on the truth.

The KJ Method is the well-defined procedure for discovery the truth by organizing amorphous information into understandable knowledge. The KJ Method is a thinking framework for getting rid of any biases. The discovered truth, which is a basis for clear judgments in the problem solving process, is synthesized through the process of chart making in the KJ-Method. Especially in the label grouping step, which is the heart of the KJ-Method, Jiro KAWAKITA stated that "*Practitioners must free themselves from any prejudice or preconceived ideas and carefully listen to what the labels are trying to say*". Without any biases, the problem is clearly understood and can be correctly solved.

According to the original KJ Method written by Jiro KAWAKITA [12], it supports important parts in the human problem solving process. The KJ-Method is invented to provide a ready-made formula for the hypothesis formulation step in the W-shaped problem solving model [15], which is a problem solving model progressed back and forth between the level of thought and the level of experience illustrated in Figure 2.5.

The level of thought can only be advanced by mental consideration; in contrast, the level of experience can only be progressed by observation activities. The basic processes of the W-shaped problem solving model are listed as follows [15, 12]:

- (A) Presents the problem.
- (A-B) Understands the current status of the problem and situations related to it.
- (B-C) Collects all relevant data through field observation.

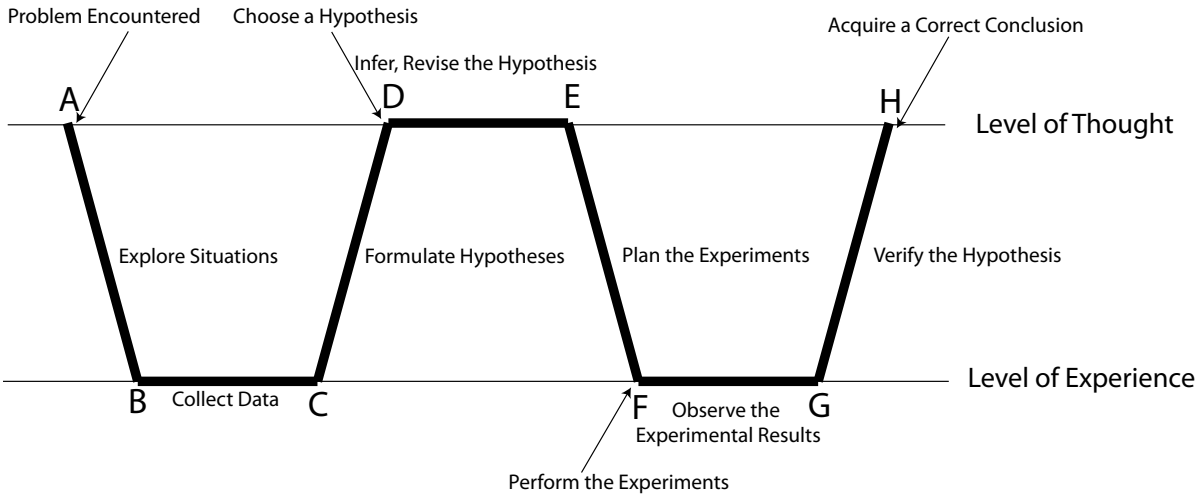


Figure 2.5: W-shaped of problem solving model

- (C-D) Formulates a number of hypotheses.
- (D) Evaluates hypotheses and chooses which one to adopt.
- (D-E) Infers and revises the adopted hypothesis through deductive reasoning.
- (E-F) Plans the experiment for testing the adopted hypothesis.
- (F-G) Observes the experimental results.
- (G-H) Verifies the adopted hypothesis based on the experimental results.
- (H) Acquires the correct conclusion.

Western scientific research was developed to yield the conclusion from hypothesis through an experiment, and these are the basic steps from points D to H. Due to the lack of systematic methodology from points A to D, especially from point C to D, and the KJ Method was invented to fill this gap. The four basic steps of the KJ-Method involve the basic steps of the W-shaped problem solving model as follows:

1. **Label making** from points A to C.
2. **Label grouping** from points C to D.
3. **Chart making** from points C to D.
4. **Explanation** from points C to D.

The KJ-Method, the practical thinking methodology, completes the whole W-shaped problem solving model. This is the combination of eastern and western discoveries.

The KJ Method, which frees us from dogma, allows creative thinking based on the reality. It is a practical tool for helping people to perceive and understand the complexities of any problems or situations accurately. It also opens up a completely new range of possibilities for

creative thinking. It has contributed to a number of major discoveries and inventions [12]. The KJ Method is proven to improve human creativity.

The KJ Method has not just only promoted the accurate thinking process and creativity, with the clear presentation of concept and detailed consideration recorded in both the KJ Chart and written explanation, the KJ Method improved mutual understanding among participants during the KJ Method process. Without the biased information, the KJ Method deepens mutual understanding and promotes harmony in a group of people, even though they have different cultures, different schools of thought, or different interests [12].

The KJ Method is recognized as an effective tool for gaining consensus among participants in the decision making process. Since the KJ Method allows people to equally express their opinions and take part in the designing of the final solution, group consensus can be easily gained. It can significantly reduce any wastage in resources for gaining consensus and allows the whole team to enjoy the benefits retrieved from the co-operation more quickly.

2.6 Conclusion

The KJ Method is a well-defined procedure for discovering the truth from badly organized information by arranging amorphous information to understandable knowledge. It was proposed by Jiro KAWAKITA in 1967 and gained in popularity to this day. The KJ Method consists of four basic steps: (1) Label Making Step; (2) Label Grouping Step; (3) Chart Making Step; and (4) Explanation Step.

With these four basic steps, the KJ Method helps us to unlock free thinking, support creativity, increase mutual understanding, and gain consensus. The KJ Method is an oriental missing piece elaborated with western scientific research methodology to complete the whole problem solving model. It is the combination of eastern and western discoveries.

In the next chapter, the KJ Method support systems and groupware is described. This is a computer-based system invented for supporting the KJ Method process. This is invented to reduce the requirements and difficulties in performing the KJ Method and to promote the thinking capability of the participants.

Chapter 3

Literature Reviews: Previously Proposed KJ-Method Systems and Groupware

3.1 Introduction

Working activities in our modern present day societies, high-level intellectual activities such as researching, designing, and strategic planning, requiring creative problem solving, are significantly increasing in both quality and quantity. In response to this rising demand, creativity techniques are proposed to support these intellectual activities.

Creativity techniques are methods for encouraging human creative action and are usually applicable to a range of domains and contexts. Various contexts of creativity are focused on, such as idea generation or divergent thinking (a method for generating creative ideas by searching for many possible solutions), idea selection or convergent thinking (a method for integrating many possible solutions into a single best solution), artistic expression, inspiration, and problem solving.

For the problem solving context, the human-thinking process for creative problem solving consists of four sub-processes according to Kunifuji and Kato [17] as follows:

1. **Divergent thinking** generates creative ideas by exploring many possible solutions[18].
2. **Convergent thinking** narrows down all discovered possible solutions to a single best solution[18].
3. **Idea crystallization** thoroughly revises the chosen solution.
4. **Idea verification** experiments with the selected solution and verifies its quality.

The KJ-Method, a popular problem-solving creativity technique in Japan, has been proposed to elevate these four sub-processes by its four basic procedures: (1) Label Making; (2) Group Making; (3) Chart Making; and (4) Explanation. Label making encourages investigators to write all possible ideas for solving a problem onto a set of labels (divergent thinking). In group making, investigators cluster the set of ideas that seem to belong together as a group and make a group title which describes the essence of all members (convergent thinking). In chart making, all groups are spatially arranged based on their association on a large sheet of paper called a "KJ chart" (idea crystallization). In explanation, the chart is thoroughly described by investigators (idea verification).

With the current modern information technologies, creativity support systems, an interactive computer system aimed at increasing creative thinking abilities of users[5], have been recently proposed by many authors. A creativity support system consists of idea visualization and intelligent suggestion modules. These modules suggest reading users input, processing, and suggesting knowledge, formations, data, or ideas aimed at accelerating creative thinking.

Many systems support creativity through their new invented methodologies [17, 19, 20, 21]. Many of them support creativity by strictly following the instructions of a creativity technique, such as the KJ Method [22, 23], brain storming [24], and brain writing [25]. This type aims to reduce the barrier of these creativity techniques, to increase their efficiency, and to expand their benefits. The increased efficiency in creative techniques by these support systems means efficiency in the human thinking process.

With the emergence of groupware technology, creativity support groupware, which are computer systems invented with the aim of supporting the group creativity techniques and simultaneously serving multiple users who are remotely located from each other. Creativity support groupware usually provide several tools for supporting the process of group creativity techniques in several aspects such as communication, construction, and collaboration via computer networks. Communications in creativity support groupware are mainly designed for accurately sharing ideas, data, and information for brainstorming purposes. In the construction aspect, creativity support groupware provides creativity assistants (an application that provides information or hints to stimulate the creativity, such as random words, etc.) for promoting team and individual creativity powers.

In the collaboration aspect, creativity support groupware provides tools for systematically managing and promoting participants contributions to the final solution, such as version control systems, shared workspace, data visualization, etc. Version control systems track the changes and developments of the groups ideas. Shared workspaces are used for identically distributing and displaying the status of the groups ideas or documents, currently being worked on by users in a team, to every participant. Data visualization is used for supporting idea presentation for group understanding and coordinating their conversational communication.

The KJ-Method support groupware is a creativity support groupware aimed at supporting the group KJ-Method. KJ-Method support groupware is supposed to be the direct translation of a process that a group of people performs in the group KJ-Method with paper and pen. The communication channel in a KJ-Method groupware will provide functionality for easily specifying which idea label or category is currently in discussion. Creativity assistants will give support in divergent thinking, convergent thinking, idea crystallization, and idea verification, which are the thinking processes found in the four basic steps of the KJ-Method. Since the final output for the group KJ-Method is a KJ-Chart, KJ-Method support groupware will be distributed, tracked, and visualized with KJ-Charts in the shared workspaces.

There are several KJ-Method support systems and groupware recently proposed by many researchers especially in Japan. In this chapter, we will review all previously proposed KJ-Method support systems and groupware. It will be organized as follows: Section 3.2 describes the previously proposed KJ-Method support systems; Section 3.3 explains the previously proposed KJ-Method support groupware; and Section 3.4 concludes the chapter.

3.2 Previously Proposed KJ-Method Support Systems

Back in 1990, desktop computers became more easily accessible to consumers in Japan. Computers provide very high computational power and accurate data storage. Due to the benefit of using computers, the demand for enabling a computer to conveniently store and manage the huge number of ideas discovered during the brainstorming sessions gradually increased. To respond to this rising demand, the first KJ-Method support system was invented.

3.2.1 KJ Editor

The first KJ-Method support system "KJ Editor" was proposed by Ohiwa et.al. in 1990 [26, 27, 28, 22]. The KJ-Editor is a single-user KJ-Method support system that allows a user to perform the KJ-Method creativity technique on a computer on an ordinary table. The KJ-Editor consists of a graphical user interface, which displays a KJ-Chart with a number of virtual KJ cards. These virtual KJ cards can be generated and placed in any position on the desktop. The content on a card can be written using a keyboard. Using a mouse can change the position and size of a card.

Since the size of computer display limits the size of the virtual KJ-Chart, the KJ-Editor superimposes two screens on a computer display. One is a universal screen that shows the overall arrangement of the cards, but the content of individual cards is invisible. Another is a local screen that gives a close-up view of the card arrangement on the universal screen. The written characters on the individual cards are clearly visible and readable on the local screen. Two superimposed screens can be easily distinguished by using different colors. The two superimposed screens are shown in Figure 3.1. According to Youngs classification [5], this system supports at the framework-paradigm level.

3.2.2 D-ABDUCTOR

"D-ABDUCTOR", a compound graph drawing application that is able to draw a KJ-Chart on a SunOS workstation [29], was proposed by Misue et.al. in 1993, 1996, and 1998 [20, 30, 31]. The screenshot of D-ABDUCTOR is shown in Figure 3.2. D-ABDUCTOR is a universal drawing application which is not intended to be built for supporting the KJ-Method, but it is very popular among KJ-Method practitioners for saving a KJ-Chart. It provides a complete GUI desktop environment for drawing and editing a compound graph (a KJ-Chart), including tools that facilitate drawing and handling compound graphs automatically.

The key features of D-ABDUCTOR are listed as follows:

1. **Compound Graph:** D-ABDUCTOR provides a complete GUI desktop environment for creating, editing, and customizing a KJ-Chart. It allows the user to put text or image data into an idea label. It enables KJ-Chart drawing such as adding, editing, moving, arranging, and grouping idea charts and linking the relationships among objects in a KJ-Chart. The appearance of a KJ-Chart can be customized.

2. **Display with Animation:** D-ABDUCTOR displays the changes happening in the chart with the smooth animation. It reduces the instantaneous visual changes, which can preserve the users mental map. The speed of animation can be customized.

3. **Collapse and Expand Operation:** A group of KJ Labels can be collapsed or expanded for saving on the screen area. The outline of a KJ-Chart can be shown by collapsing all

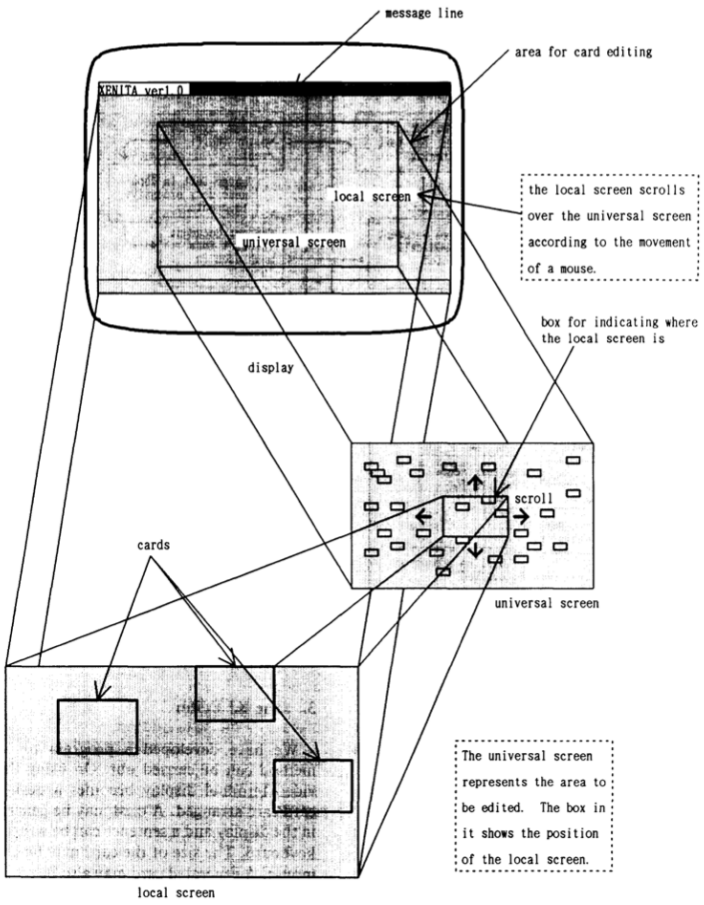


Figure 3.1: Two superimposed screen of the KJ-Editor, universal and local screens

group labels. The detailed information of a KJ-Chart can be viewed by expanding all group labels.

4. **Fisheye View:** The size of group title or idea label is varied according to their importance. The larger size is more important, and vice versa, the smaller size is less important.

5. **"Simple" KJ-Chart Text Description Language:** A graphical KJ-Chart can be represented as text by Simple description language. Text form allows D-ABDUCT to save a chart, easily edit a chart in command line windows, and transmit it over the computer network in another instance.

3.2.3 Group KJ Method Support System Utilizing Digital Pens: GKJ

With the limitation on capturing and sharing the KJ Chart among practitioners during the traditional group KJ Method session, the "Group KJ Method Support System Utilizing Digital Pens" or in short "GKJ", is proposed by Miura et.al. [32] in 2011 for capturing the location of written idea labels and their written content by using a set of external devices. The performance of GKJ is later evaluated [33] and continually developed to allow the user to customize or add their own gesture commands [34]. According to Youngs classification [5], this system supports at the framework-paradigm level.

The set of external devices are list as follows:

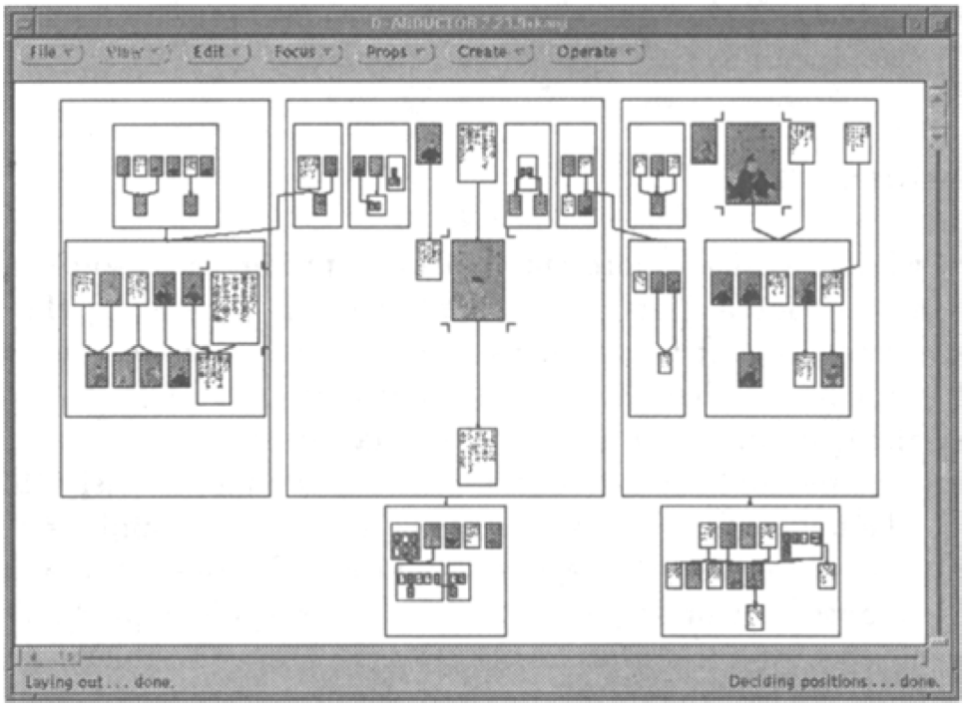


Figure 3.2: The screenshot of D-ABDUCTOR

1. **Anoto Pen:** An electronic pen used for capturing the written text and the location of the card.

2. **Digital Pen Gateway System:** The gateway box wired with a personal computer for forwarding data from several Anoto pens simultaneously via Bluetooth to the personal computer.

3. **Personal Computer that runs the GKJ Editor:** The main personal computer for collecting information from several Anoto pens and constructing the KJ-Chart based on such information by using the GKJ Editor.

4. **Small Rectangle Labels and a Big Based Sheet:** They are electronic tags and devices for communication with the personal computer. Each small rectangle label has a unique pattern in the back (similar to the QR code). They are used in detecting the location of the idea labels and their arrangement on the base sheet.

Users perform the traditional group KJ-Method by using Anoto pens instead of legacy pens or pencils. For the label making step, users draw or write the text on small special labels. In the label grouping step, users group the labels as a stack, as in the traditional group KJ-Method, and then draw a round stroke covering on all grouping cards at once. In the label ungrouping step, users draw a single stroke covering all cards in the group at once. For the chart making step, users can attach the rectangular label into the base sheet by drawing over the border of border on both the base paper and label. Users can also draw a circle or write text on the base sheet directly to create the KJ-Chart.

With the above gestures, the electronic version of the KJ-Chart is copied from the physical KJ- Chart in the base sheet. The electronic version of the KJ-Chart can be stored right after the end of a group session and can be distributed to other participants or third parties immediately.

The final stage of the KJ-Chart, captured by the Anoto pens, is constructed by the GKJ system on the main computer, as shown in Figure 3.3.

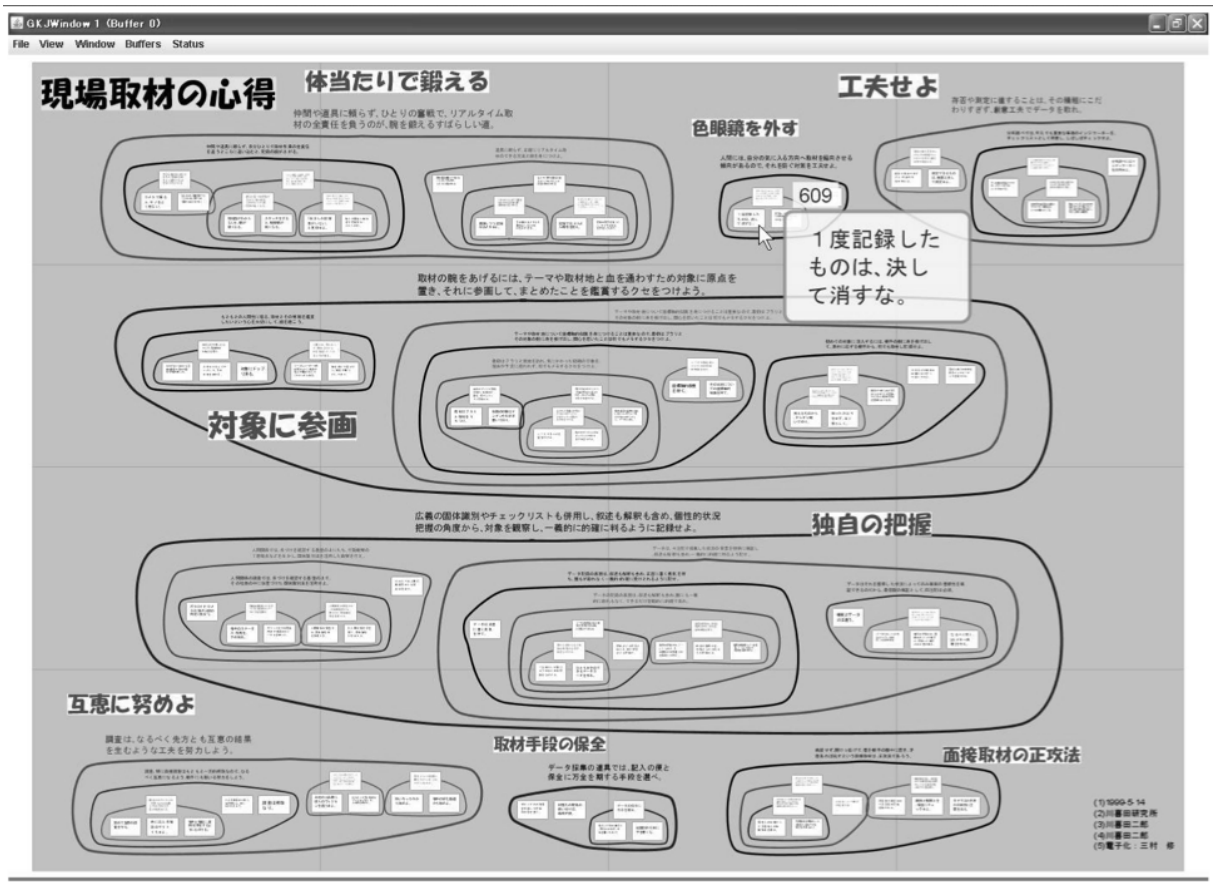


Figure 3.3: Final KJ-Chart captured by Anoto pens in the traditional group KJ-Method session constructed by the GKJ application

Even though GKJ supports multiple users in performing the group KJ method, it is classified as merely a KJ-Method support system, since this system does not provide any group communication modules for collaboration among team members while remotely located from each other. In other words, the users of the GKJ support system must meet in person while using the system.

3.3 Previously Proposed KJ-Method Groupwares

In the last two decades, the KJ-Method groupware was proposed by many researchers, especially in Japan, due to the popularity of the KJ-Method and the emergence of the single user KJ-Method support system. Most KJ-Method groupware is developed on the basis of their previously proposed single-user KJ-Method support systems. Such single-user KJ-Method support systems became the KJ-Method groupware by adding the shared screen, the group interaction, and the group communication functionalities.

3.3.1 Groupware for a New idea Generation Consistent Support System: GUNGEN

The first KJ-Method groupware “GUNGEN” was proposed by Munemori and Nagasawa firstly in 1991[2], and then continuously improved in 1992[3], and 1996[35]. The latest version of “GUNGEN” was proposed in 1998[36] by Yuizono et al.

Later “GUNGEN-GO” was proposed by Yuizono et al. in 2003[37]. It extends three functions of hypermedia systems over HyperCard, which is intensively used in GUNGEN, for supporting awareness functions that improve shared consciousness between remote sites.

After that, “GUNGEN DX II” was proposed by Shigenobu et al. in 2003[23] and in 2005[38]. Its performance is empirically evaluated [39]. For better accessibility of the system, the external devices are used as part of the groupware. Using PDA as the input device for collecting ideas for the GUNGEN DX II was proposed by Yuizono et. al. in 1999 and by Yoshino et. al. in 2001[40, 41].

For allowing a groupware display on the composition of multiple screens, Yuizono et. al. proposed the “KUSANAGI” KJ-Method groupware, which was developed by the JAVA programming language [42, 43]. KUSANAGI is platform independent software, which has the capability of the JAVA programming language. It can be used in most operating systems for personal computers.

For completely removing the necessity of installing software, eliminating plat form dependence, and improving accessibility in mobile devices, the Webbased groupware “GUNGEN-SPIRAL II” was developed and proposed by Jun Munemori et. al. in 2012 [44]. The screenshot of GUNGEN SPIRAL II is shown in Figure 3.4.

Figure 3.4: A GUNGEN SPIRAL II screenshot

GUNGEN

GUNGEN consists of AppleTalk, Timbuktu, HyperCard, and Wadaman. AppleTalk is a networking protocol that allows computers to be connected to a local area network [45]. Timbuktu is remote control software that allows the user to control another computer across the local network, viewing its screen and using its keyboard and mouse as if he or she were sitting in front of it [46]. Hypercard is an application program that creates, stores and manages a stack of virtual cards that contain information such as text and images, and also contains a number of interactive objects, including text fields, check boxes, buttons, etc [47].

Wandaman is an extension of HyperCard system that is an intellectual productive work support system acted as a database that can stores and retrieves the previous ideas (cards). It used as a multimedia database and retrieval system in GUNGEN.

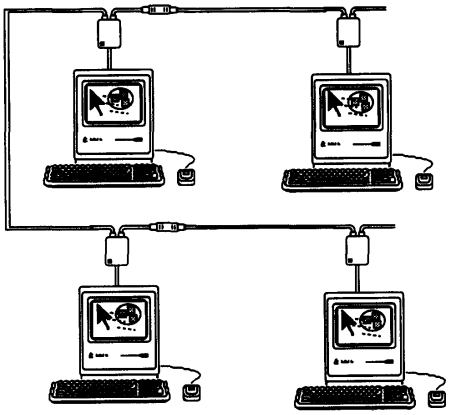


Figure 3.5: GUNGEN's conceptual model for distributed and cooperative KJ-Method support system [2].

The conceptual model of the GUNGEN system is shown in Figure 3.5. Computers are connected to a local area network by using Appletalk. Shared space can be achieved via Timbuktu software.

GUNGEN is written by the HyperTalk programming language, which is based on the Hypercard application. GUNGENs user interface was inspired from Colab [48], which is a famous electronic conference system developed by Xerox Corporation, and Cognoter [49], which is a program to help a group to organize their thoughts for a presentation built on top of the Colab.

GUNGEN has design policies as follows:

1. Support a distributed and cooperative KJ-Method, especially the A-type KJ-Method.

2. Combination of the KJ-Method with a multimedia database.

3. Easy to operate.

GUNGEN's user interface consists of four major components as follows:

1. **Brain Storming Windows:** This is shown in Figure 3.6. A problem topic, names of participants, and paint tool icons are presented.

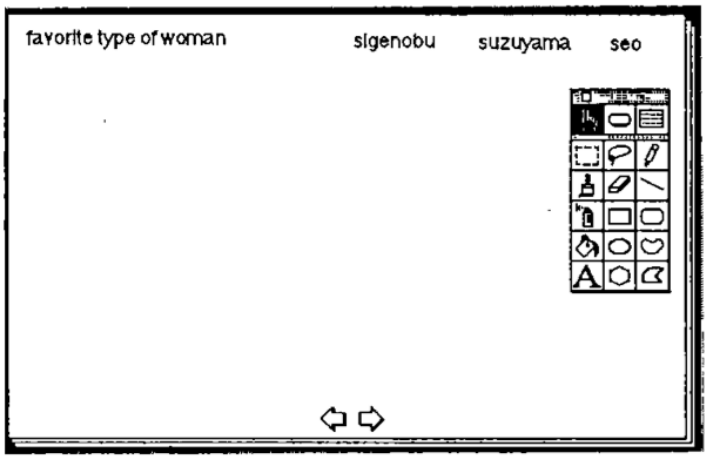


Figure 3.6: GUNGEN's initial screen [3].

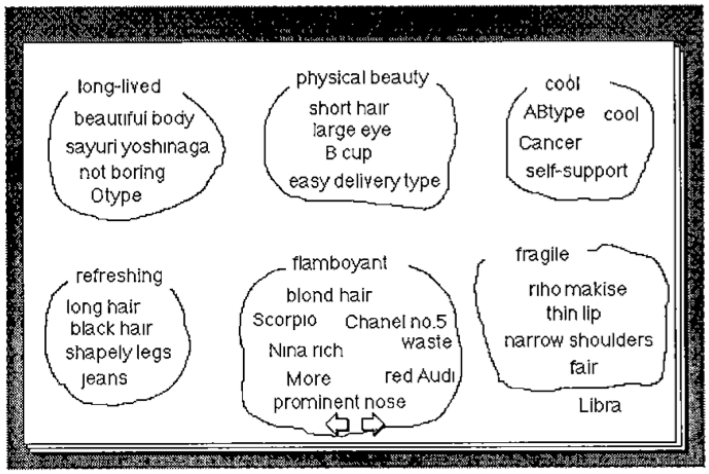


Figure 3.7: GUNGEN's electronic tag on the label grouping step [3].

2. **Paint Toolbox:** Tools in the paint toolbox on the right border of Figure 3.6 used for writing, drawing and putting a interactive objects onto a card.

3. **Electronic Tags:** Text and image data on a screen are taken as electronic tags. A select tool or lasso tool in the paint toolbox is used to move tags.

4. **Multimedia Database:** Finished KJ-Charts are stored on the Wadaman database. Previous KJ-Charts and their labels can be called up with Wadaman while the cooperative KJ-Method is carried out. The screen of Wadaman is shown in Figure 3.8

Since the HyperCard application shows one card at a time on a screen, the hierarchical structure of cards in a KJ-Chart can be done by using a linkage button, which is a one of interactive object that navigates towards another card when it is clicked. A linkage button is attached to the title part of each group. Then, the screen is navigated to show the detail of that group. Figure 3.9 shows the linkage function in GUNGEN.

GUNGEN supports label making, label grouping and written explanation steps in the KJ-Method. The details of the KJ-Method supports are listed as follows.

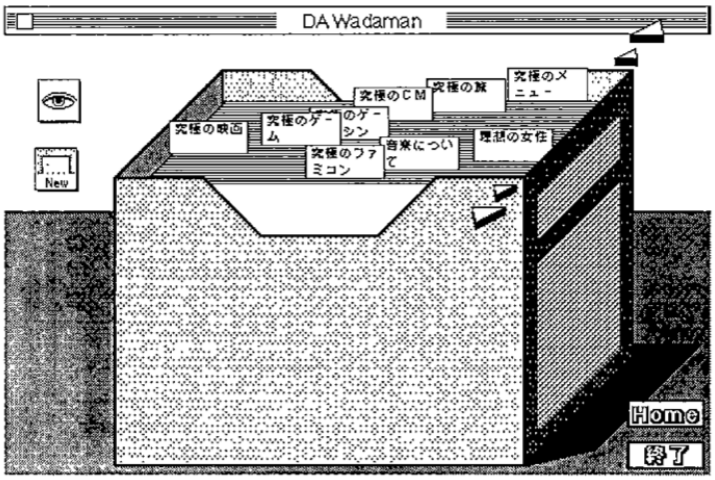


Figure 3.8: Wadaman screen in GUNGEN system[3].

1. **Label Making:** Each user separately creates cards (labels) by typing text or choosing the existing card from the Wadaman database.

2. **Label Grouping:** Users jointly arrange and group the cards on the shared screen using tools in the paint tool box, but the chart making step is not included in the groupware.

3. **Written Explanation:** Users write the report on the card based on the final KJ chart.

In 1996, a text-based communication system was added into the GUNGEN groupware In 1998, many improvements were made as follows:

1. **KJWare:** KJWare was added. It provides a better user interface than earlier versions. A number of cards (labels) can be shown on a screen at once. The inconvenient linking function found in the earlier version for showing the hierarchical structure of a KJ-Chart on a screen that can show only one card at a time is no longer required. KJWare provides the share screen function and text-chat communication for the group KJ-Method. KJWare allows users to input, remove, arrange, and group idea labels. It has a fully functional text editor in a separated window to write a conclusion report. The screen of KJWare is shown in Figure 3.10.

2. **NetGear:** NetGear was added. NetGear provides multimedia communication (videos and voices) and file transfer services among team members.

GUNGEN DX II

The GUNGEN DX II, which was continuously developed from GUNGEN for better supports of the large amount of ideas, and for quicker gaining of the group consensus in the convergent thinking phase, was later proposed by Shigenobu et. al. in 2003 and 2005 [23, 38]. The GUNGEN DX II has a unique idea grouping user interface as shown in Figure 3.11. The idea grouping step in the GUNGEN DX II is separately performed by each user, not jointly performed as in the earlier version of the GUNGEN groupware. A user solely groups the idea based on his opinion only, thus the idea grouping pattern of each user can be different.

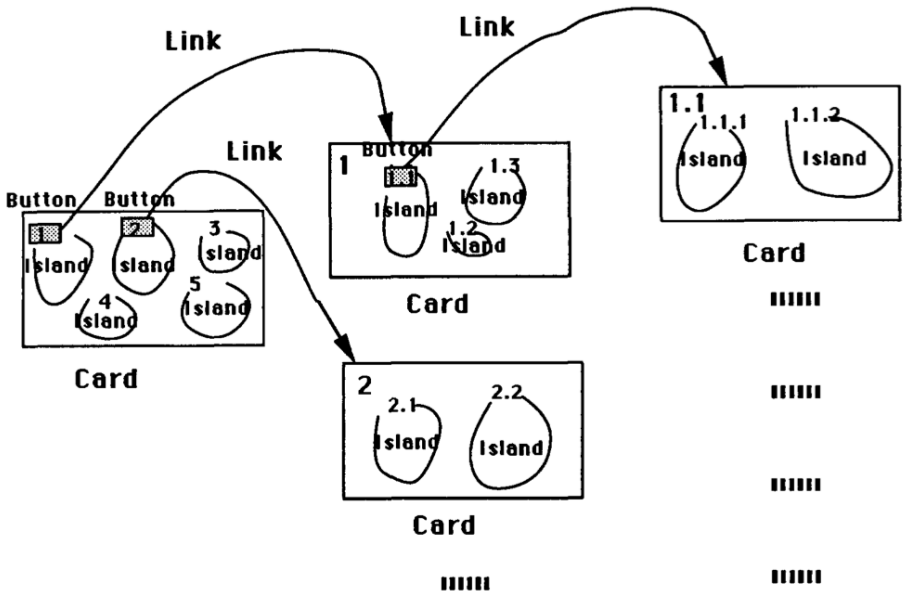


Figure 3.9: GUNGEN's linking [2].

Once all users finish the idea grouping step, the GUNGEX DX II generates the island of idea groups automatically based on the grouping results of all users. This method supports consensus building in grouping ideas, which in turn also supports the convergent thinking of the group.

The performance in consensus building of GUNGEN DX II is later empirically evaluated [39]. The study claimed that the number of times that idea labels moved decreased over 80%, compared with the GUNGEN. The time usage for gaining group consensus decreased by around six times, compared with the original groupware (GUNGEN).

3.3.2 Group Idea Processing System : GrIPS

The "Group Idea Processing System: GrIPS" was proposed by Kohda et. al. and developed by the Fujitsu Laboratory in 1993 [50]. The GrIPS is developed based on their previously proposed KJ-Method support systems, which are "IdeaEditor" invented by Isamu Watanabe and "D-ABDUCTOR" invented by Misue et.al. According to Young's classification [5], this groupware supports at the generative level. It was developed by the Fujitsu Laboratory in 1990 [51]. The screenshot of the GrIPS system is shown in Figure 3.12.

The GrIPS supports a group of users to create the KJ-Chart through two thinking phases [52]:

1. **Label Making: Divergent Thinking Phase:** During the divergent thinking phase, each user inputs data, ideas, or information that are relevant to the problem topic into the system. The system visualizes such information as cards on an electronic shared board.

2. **Label Grouping, Naming and Chart Making: Convergent Thinking and Idea Crystallization Phase:** During the convergent thinking and idea crystallization phases, users jointly arrange and group the cards into groups and organize them into a diagram.

The GrIPS provided three useful support modules for each thinking phase as follows:

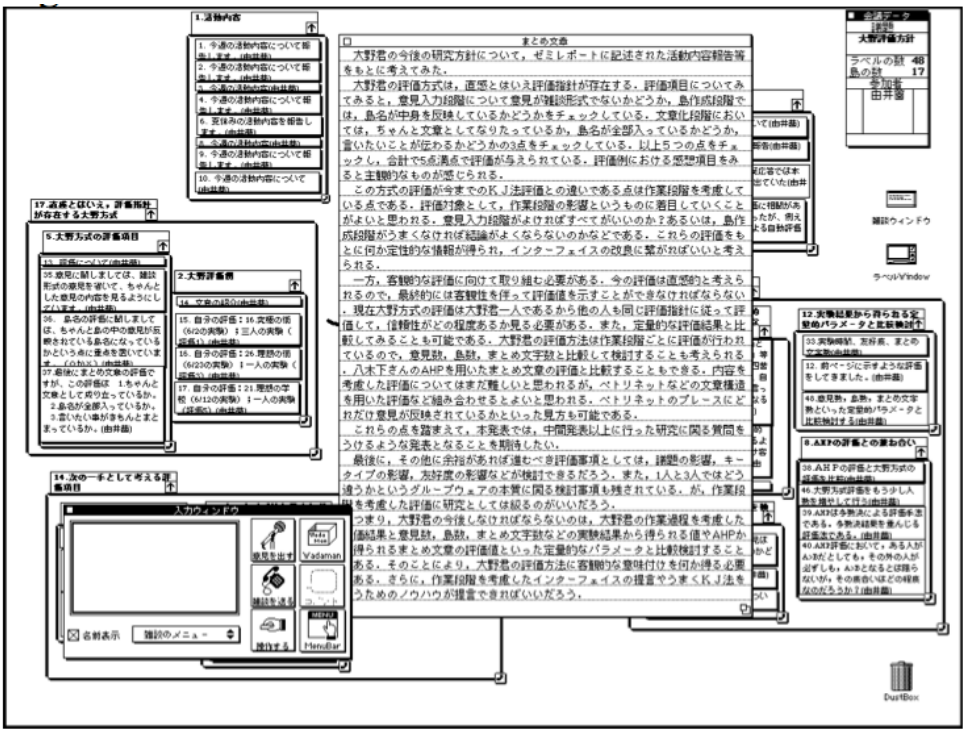


Figure 3.10: KJWare user interface in GUNGEN

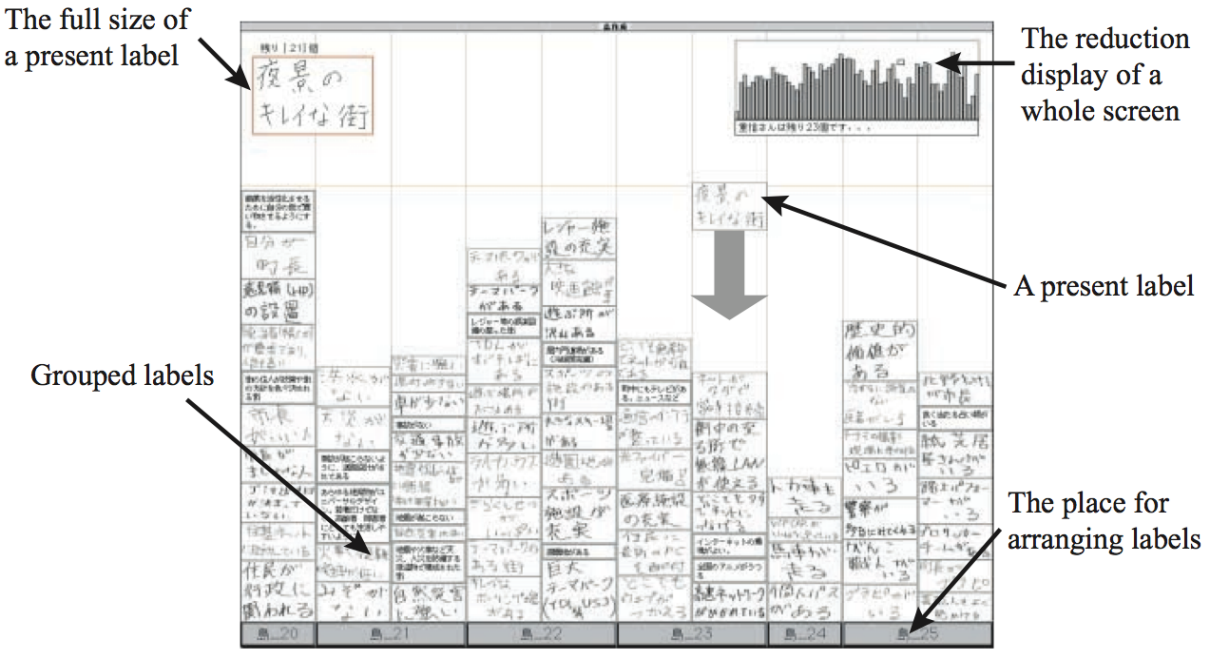


Figure 3.11: A GUNGEN DX II screenshot and functionality

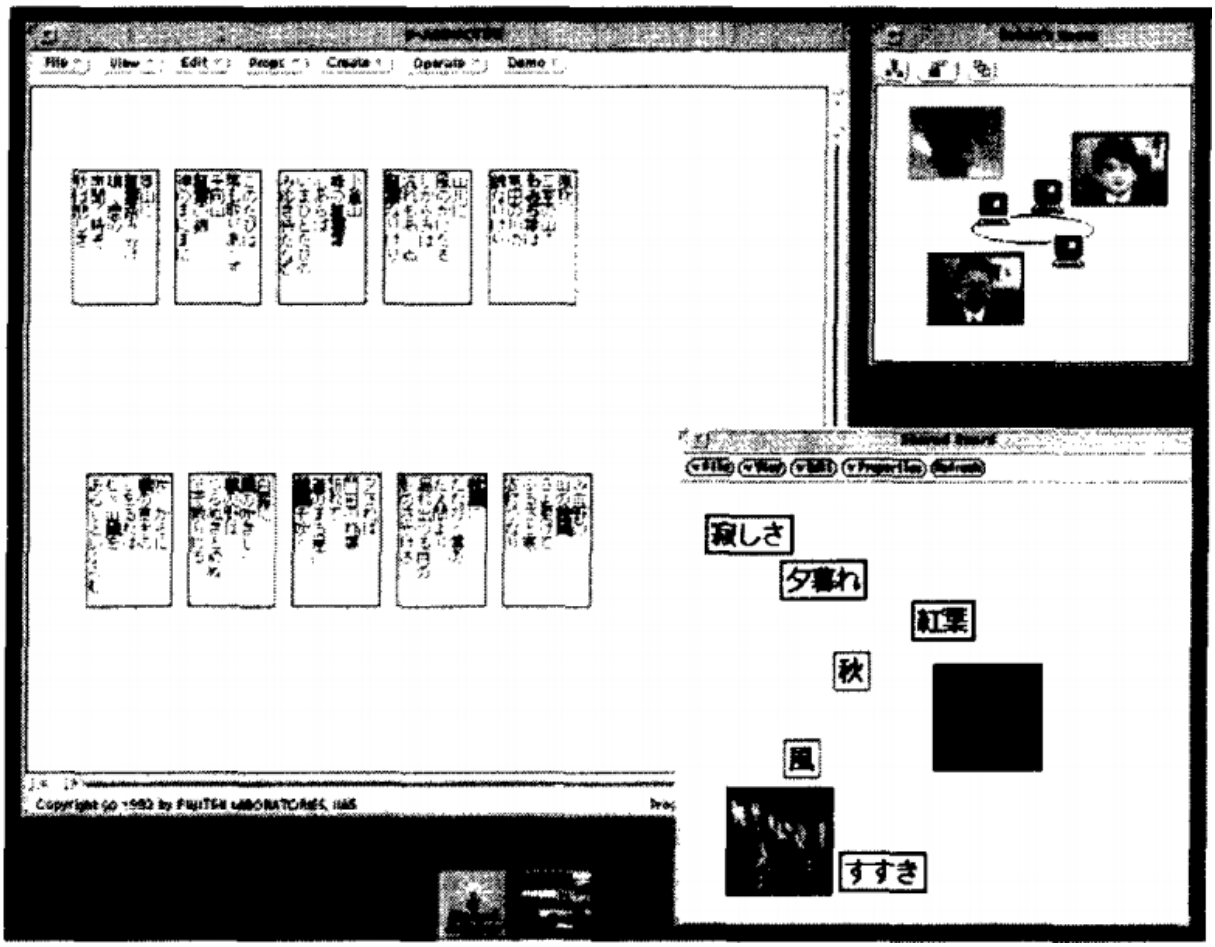


Figure 3.12: The screenshot of GrIPS

1. **Associative Keyword Retrieval:** The list of associative keywords is shown during the divergent thinking phase for promoting idea association aimed at invoking the generation of new ideas. These keywords are retrieved from the "Keyword Associator" [53] and "Picture Library" databases.

2. **Automatic Diagram Drawing:** The automatic diagram drawing converts a complex diagram into an easy-to-understand (tidy) diagram that has the same structure as the original during the convergent thinking phase. It reduces time consumption and workloads for formatting the KJ-Chart.

3. **Individual-Basis Sound Mixing:** This module supports team communication during both phases of thinking. The individual basis of sound mixing simulates the acoustic effects in a meeting room, by mixing the voices of each team members in suitable proportion. It is aimed at compensating for the loss of communication reality due to the virtual working environmental.

3.3.3 GRAPE: Groupware for Acquiring, Processing, and Evaluating Knowledge

GRAPE, a knowledge acquisition support groupware, was proposed by Haruyasu Ueda and Susumu Kunifuji in 1993 [54] and mentioned again in 2007[17, 15]. It is very useful for constructing initial knowledge, which is necessary for rapid prototyping a knowledge base system. It acquires shared knowledge of a topic from a group of users, which tends to be less biased and more complete than from an individual.

It is designed for solving classification-choice problems. The main processes of GRAPE are candidate generation and candidate selection. In candidate generation, it follows the philosophy and procedure of the KJ-Method for acquiring (label making) and merging (label grouping) shared knowledge from many others. In candidate selection, it integrates and uses several techniques, such as:

- **PCP (Personal Construct Psychology):** PCP is used for acquiring the names of the attributes to evaluate the candidates.

- **Extended ISM (Interpretive Structural Modeling):** Extended ISM is used for describing and arranging the attributes.

- **AHP (Analytic Hierarchy Process):** AHP is used for eliciting the importance among the attributes and candidates.

GRAPE has WYSIWIS (What You See Is What I See) interface. Its procedures are described as follows.

1. **Candidate Generation:** for acquisition shared knowledge from participants, following the procedures and philosophy of the KJ-Method.

   (a) **Candidate Acquisition:** participants input their knowledge, information, or solution of the problem to generate the candidates via the WYSIWIS interface (matches with the label making step in the KJ-Method).

(b) **Candidate Structuring:** participants make a tree structure of the inputted knowledge (this matches with the label grouping step in the KJ-Method) by inputting the degree of similarity between candidates in the range $[0, 1]$. The fuzzy clustering (a convergent thinking support module) automatically generates the tree structures of knowledge from these similarity degrees.

2. **Candidate Selection:** for selecting the best solution for a problem topic, which intensively follows the procedure of AHP.

   (a) **Attribute Acquisition:** PCP is now used for acquiring the attributes for evaluating knowledge (labels) and trees (groups).

   (b) **Attribute Structuring:** Extended ISM is used for clustering the attributes. Dependent attributes (set of attributes where their value is statistically dependent on other variables) become leaves of trees. Independent attributes become roots of trees. Only independent attributes are used for AHP calculation in the next step.

   (c) **Class Evaluation:** AHP is used for evaluating the importance between attributes and is later used for evaluation the importance among candidates (labels and groups) for the best solution to the problem topic.

GRAPE is an excellent decision support groupware. Since it is based on the KJ-Method, it is also considered as a KJ-Method groupware that also provides excellent framework-paradigm convergent thinking support tools such as Fuzzy clustering, PCP, Extended ISM, and AHP. The screens of GRAPE are shown in Figure 3.13, and 3.14.

3.4 Conclusion

This chapter is a detailed review of two previously proposed KJ-Method support systems and three previously proposed KJ-Method groupware families started from 1990 to the present day. Since the KJ-Method is popular, especially in Japan, all systems and groupware were invented by Japanese university professors or Japanese companies. Software for supporting the KJ-Method was started with a single-user interface such as KJ Editor, and then later, a groupware like GUNGEN was proposed, which is the multi-user interface connected via the computer network. The platform of the KJ-Method groupware was started as a plug-in to existing card-based applications, then moved to desktop applications for better user experience on personal computers, and finally, was changed to a Web application for increased mobility.

In the next chapter, a novel KJ-Method groupware will be proposed. Its descriptions, user interfaces, infrastructures, and designs will be described in detail.

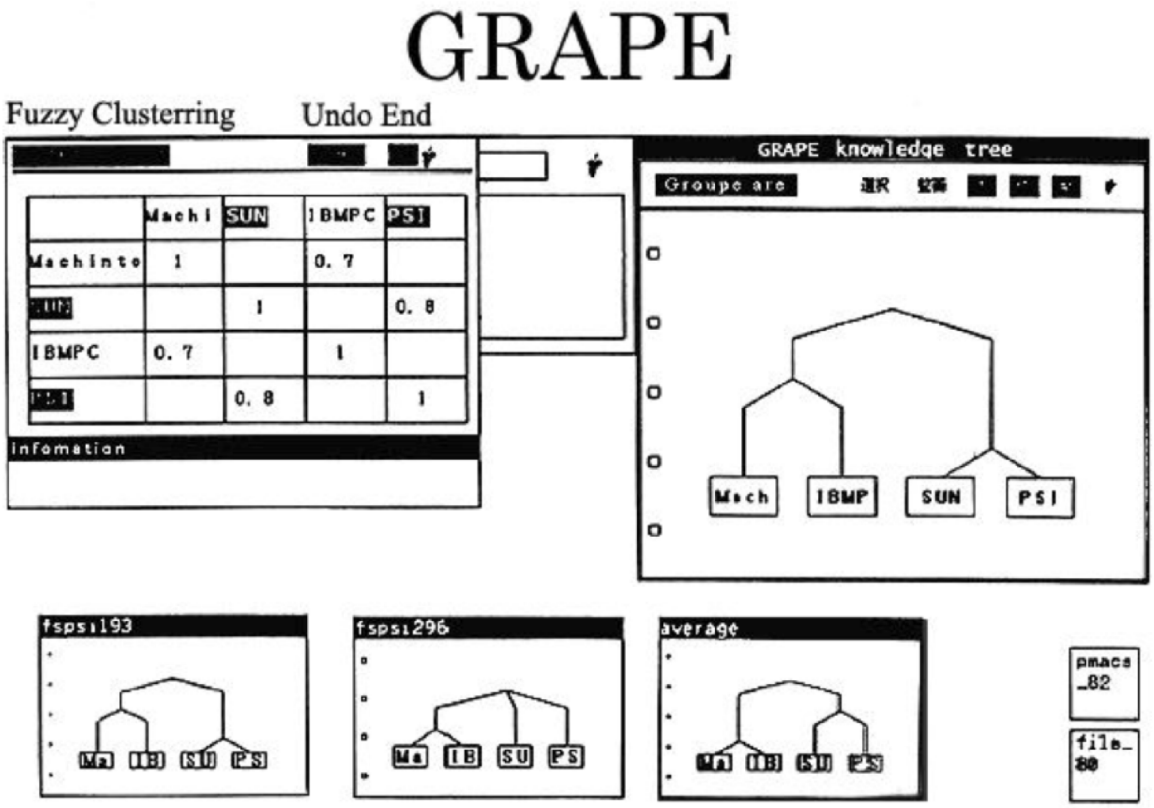


Figure 3.13: The screenshot of GRAPE (1)

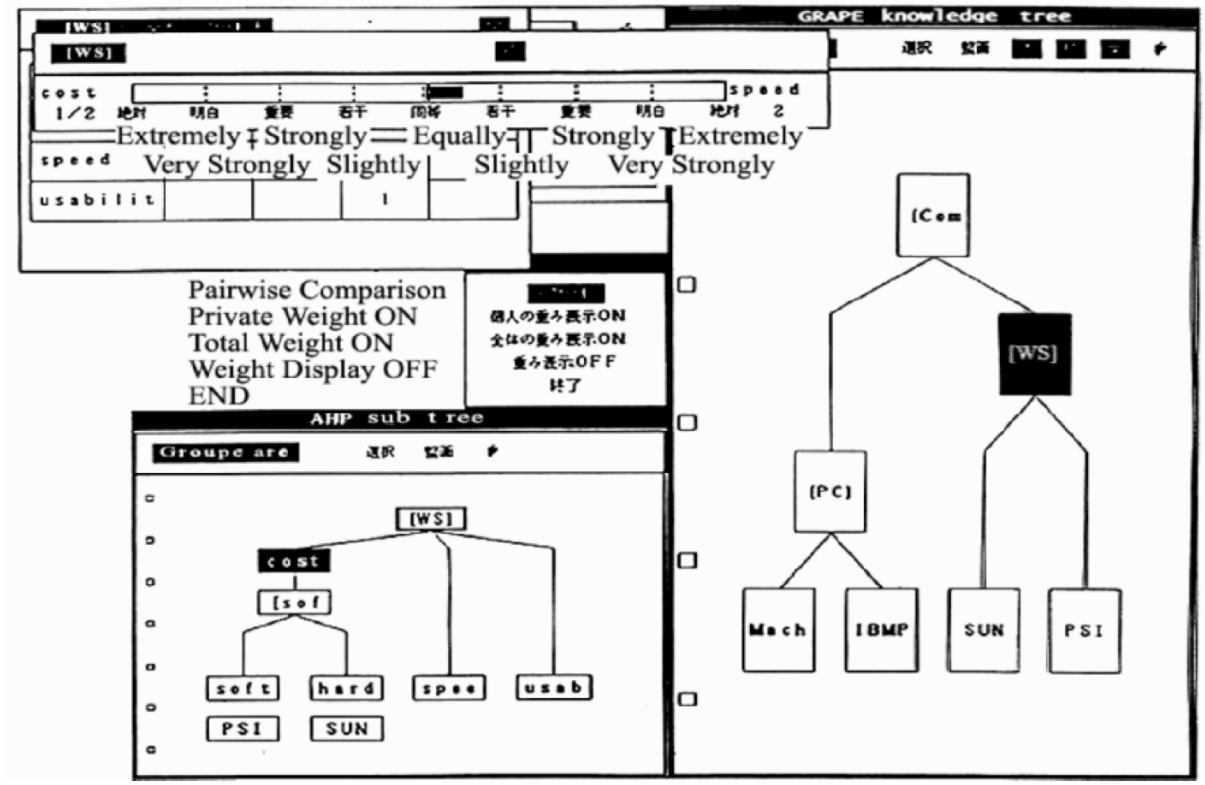


Figure 3.14: The screenshot of GRAPE (2)

Chapter 4

Gugeek: Web-based KJ-Method Creativity Support Groupware

4.1 Introduction

"Gugeek" is an ambitious attempt to provide open and accessible KJ-Method groupware with intelligence idea processing support. It allows a group of people to perform the group KJ-Method together via the Web even though they have different time availabilities and stay in different locations.

The Gugeek is exciting and important because it promises to permit a significantly more effective way of the group KJ-Method compared to the traditional face-to-face meeting. It significantly reduces the costs of operation including transportation cost, time wasted for transportation, time wasted for waiting for other participants, and so on. It significantly boosts the creativity of the output idea via the variety of creativity assistants.

The Gugeek is a Web application developed from JavaScript and HTML5 programming languages, which are the latest Internet technology in the present day. It can be easily accessed by most devices including mobile and personal computer without any pre-installed software. The only requirement for using this software is an Internet-connected device that has an Internet browser installed.

In this chapter, the real-time KJ-Method creativity support groupware, Gugeek [1], developed on the basis of our previous single-user divergent thinking support system, "Eureka!" [19], is proposed. Since the Gugeek groupware is based on the Eureka! single-user interface, this chapter starts with the explanation of the Eureka! support system. The structures, functions, workflows, and user interfaces of Eureka! are described. The comparison between Eureka! and other divergent thinking support systems are explained.

The Gugeek is then focused on in Section 4.3. The architecture and three main components are explained. The Web-based user interface developed on the basis of JavaScript is fully visualized. The sample steps and operations of the user interface are given. The workflow of the system is provided. The comparison between Gugeek and other KJ-Method groupware is shown.

Section 4.4 describes the key issues of the Gugeek architecture and user interface designs in detail. It compares the current selection of Gugeek design with other possible choices and provides decisive reasons. The advantage and disadvantage of the current selection is given.

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

The Gugeek was implemented and opened for testing by a group of students in the Japan Advanced Institute of Science and Technology. Section 4.5 shows the qualitative evaluation of the overall satisfaction of the Gugeek. It includes the list of difficulties found during the use of the system and the list of new features that should be added. Section 4.6 concludes the chapter.

4.2 Eureka!: Prototype Divergent Thinking Support System

Before the Gugeek groupware is developed, Eureka!, a single-user divergent thinking support system (a kind of creativity support system) is developed for the experiments. A screenshot of Eureka! is shown in Figure 4.1. Eureka! is used as an experiment platform for evaluating the performance of the divergent thinking support module, which is a part of a creativity assistant tool. The experiment and its results will be described in Chapter 5.

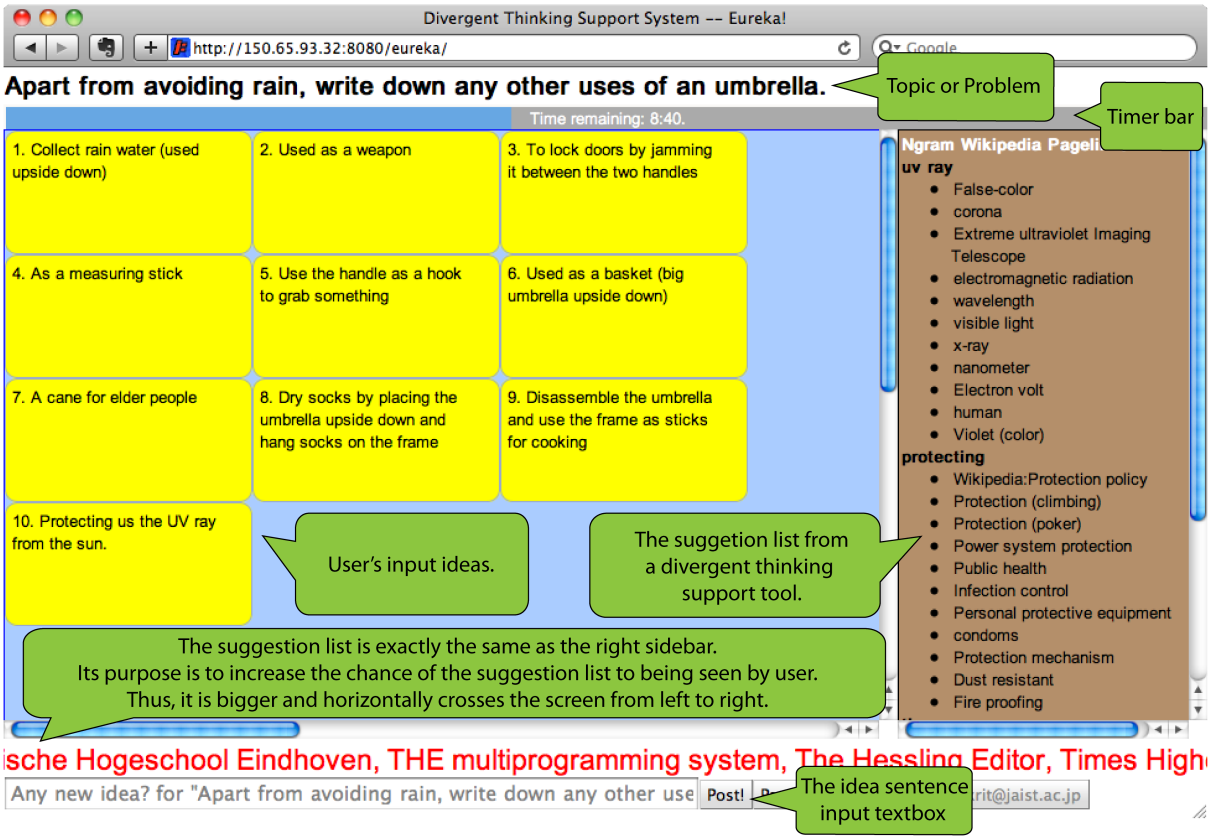


Figure 4.1: A Eureka! screenshot and functions

4.2.1 Divergent Thinking Support System

Before understanding the architecture and design of the Eureka! system, the goal and target of the divergent thinking support system should be firstly described. Divergent thinking is a thought process used for generating creative ideas by exploring as many possible alternatives as much as possible. It is an important step in creative thinking. According to Osborns rule in the brainstorming process, the most popular methodology for enhancing creative thinking proposed

by Osborn [18, 55], is: (1) Focus on quantity; (2) Withhold criticism; (3) Welcome unusual ideas; and (4) Combine and improve ideas, and support and stimulate divergent thinking.

Classification of the Idea Association Support Tools

A divergent thinking support system is a system for supporting the divergent thinking process. It supports the divergent thinking of users by stimulating the building of ideas by a process of mental association. Idea association [56] is a thought process that leads to a new idea from an original idea that is associated by principles; for example, a football and an orange can be associated because of their round shape. A divergent thinking support system can be classified into three categories by the degree of promotion of the idea association processes. Kawaji ranked these support systems from the naïve to the sophisticated [57] as follows:

1. Free association
2. Forced association
3. Analogy conception

The first category is random association, which provides random concepts or knowledge to users. It is believed to be useless.

There are several examples for the second category. Watanabe [51] proposed the "Idea Editor" that promoted an idea association process by displaying the list of associative keywords. The related keywords that are unfamiliar to users, and believed to be the best candidates for promoting idea association, are shown to users. Associative keywords are also used in the "Group Idea Processing System" proposed by Kohda, et al. [50]. In both systems, associative keywords are statically pre-computed by using the structure of documents saved in a knowledge base.

Kawaji and Kunifuji [57] proposed "Hasso-Tobi 2" divergent thinking support groupware, which extracts free links from the Japanese Wikipedia after analyzing and querying sentence input from users using a Japanese morphological analyzer. These free links are shown. Wang, et.al. [58] proposed the "Idea Expander", which extracts related images by querying an input sentence on the Microsoft Bing image search engine. In both systems, association relies on existing data and structures on the Internet.

The third category is the most difficult and only a few exist. Young [59] proposed the "metaphor" machine, which can construct the metaphors of a noun entity by searching other noun entities that use the same predicated in a large text corpus. The generated metaphors are presented to promote idea association.

Chapter 5, describes the instructions for constructing an efficient forced association module for the Eureka! divergent thinking support system. Information about the artificial intelligence module will go there. From now on, we will focus only on the design and architecture of the Eureka! support system.

Classification of Support Levels

Another criterion for classifying the divergent thinking support system is the level of overall support, in addition to way of promoting the idea association processes. There are three support levels of an idea support system, defined by Young [5], as follows:

1. **The Secretariat Level:** a system which only stores and displays the log of users thoughts, such as a word processor program.

2. **The Framework-Paradigm Level:** a system has secretariat level ability and also provides users with appropriate creativity techniques, framework, or paradigm to promote users thoughts.

3. **The Generative Level:** a system that has the framework-paradigm level of ability, and automatically constructs and displays the related information for promoting user creativity performance.

The secretariat level application is typically an office suite application such as Microsoft Word or just a naïve text editor such as Notepad, which lacks any support capability except the memorization of input ideas.

The framework-paradigm level application is a typical creativity support system currently released into the market. This application type supports the creative thinking capability of users by providing a framework for guiding users in each step of a creativity technique. For example, a KJ- Method support system with the framework-paradigm level of support should clearly provide tools to support four basic steps of the KJ-Method. The user interface should be adapted appropriately for each step in the KJ-Method. For example, in the label making step, the tool for creating virtual labels must be provided. In the label grouping step, the interface to group ideas must be provided, etc.

The generative level application is an intelligent software application that can generate related or associative information to promote users in the construction of creative ideas. For example, in a divergent thinking support system with a generative support level, the associative information related to the problem topic should be automatically provided.

The Eureka! divergent thinking support system is a generative level of support according to Youngs classification and is a forced association according to Kawajis classification. The Eureka! system is designed for presenting associative keywords and information to promote users idea association capability.

4.2.2 Structure of System

The Eureka! is a single-user divergent thinking support system. It still lacks the multi-user functionality. The goals of the Eureka! divergent thinking support system are:

1. To suggest the relevant information to a user for promoting creativity.
2. To achieve the goal of divergent thinking.

To achieve the goals of the Eureka! support system, it consists of two main components as follows:

1. Association engines.
2. User interfaces.

Association engines construct associative information for presenting to a user. A user interface is a medium which manages the interaction between a user and an engine. The structure of the Eureka! system is visualized in Figure 4.2.

4.2.3 Workflow of System

The workflow is listed as follows:

1. A user inputs a topic sentence to start the system as shown in Figure 4.2.
2. An association engine reads an inputted sentence and produces a list of associative information by using an association engine which consults with a knowledge-base, and forwards such information as a suggestion list to the interface.
3. An interface presents such suggestion list and displays it to the user.
4. The user reads the suggestion list, thinking divergently based on that suggestion list, and inputting the next discovered idea into the system.
5. The system reads such input and forwards it to the associative engine again, and repeats the process until the user is satisfied with the discovered ideas.

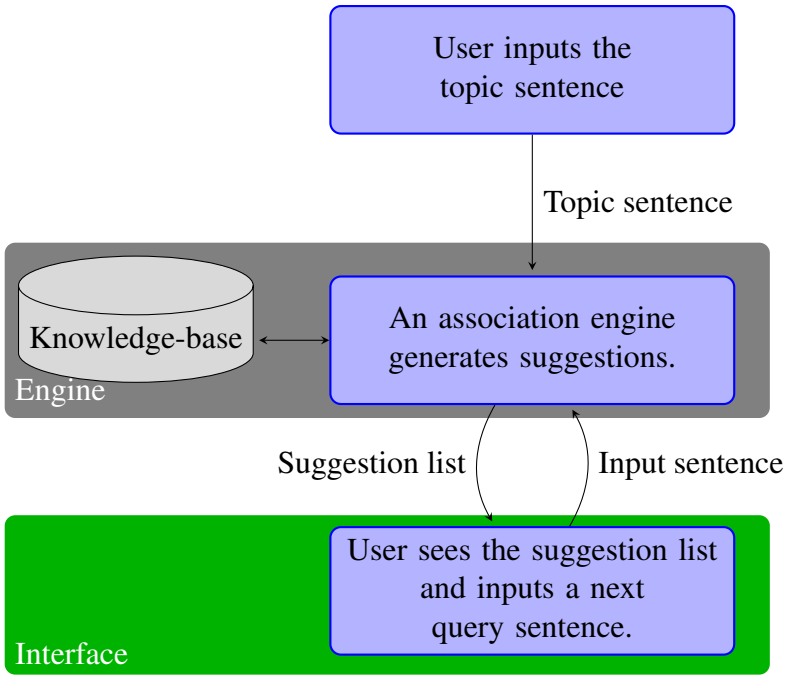


Figure 4.2: The workflow of a divergent thinking support system

4.2.4 User Interface

The user interface of the "Eureka!" system is shown in Figure 4.1, which consists of several components as follows:

The Title Topic or the Problem Statement

The title topic or the problem statement is shown at the top of the screen with large sized text. It is clearly visible aimed at preventing off-topic thinking. To set the new title topic, a user can change it by simply clicking on the title topic. The dialog box for editing the title topic is shown and is waiting for the input of new title topic. Once the new title topic is input, the topic title is changed.

The Display of Associative Information

At the right sidebar and the bottom sidebar, the suggestions are displayed. The right sidebar displays all suggestions with normal sized text to allow for a quick glance by the user. The list of associative information shown in the right sidebar is ranked based on the potential score evaluated by an association engine. The higher potential associative information is placed at the top of the list. The lower potential is placed at the bottom of the list. The ranking is done with the aim of showing the most potential information to the user first.

At the bottom sidebar, the associative information clearly displays and moves horizontally in a right-to-left manner (like a news headline). The display of associative information at the bottom sidebar is intended to give a better opportunity for a lower rank of suggestion to be seen, and providing the dynamic stream of associative information for users.

Idea Labels

All previously input ideas (query sentences) are displayed as rectangle labels in the central scrollable area. All idea labels are movable to another position in the central scrollable area by the mouse drag-and-drop operation. If a user drags an idea label into the edge of the central scrollable area, the area will scroll toward to the edges position.

The design of movable idea labels is inspired by the label grouping procedure in the KJ-Method creativity technique. A user can create islands of ideas by moving them next to each other. Allowing users to move ideas and place them next to each other, might help users to discover the association or the common principle among ideas, which promotes divergent thinking.

Another important point is the way to trigger the association engine to work. The association engine is a creativity assistant for supporting divergent thinking. The association engine finds and shows the related information of a topic currently under consideration computed from the large collection of knowledge retrieved from the quality electronic knowledge base aimed at improving the idea association process, which finally increases the performance of users in creative thinking. The algorithm of association engines used in Eureka! is described and explained more in detail in Chapter 5.

The simplest way to tell the system which label is currently under consideration is by focusing, pointing, and clicking it with a mouse cursor during the operation. When an idea label is clicked, the system assumes that the user is currently interested in that label. The selected association engine reads the content of that idea label, performs the association algorithm, and finally produces the list of associative information, which is aimed at promoting the users idea association thinking. The information is listed on both the right and bottom sidebars.

Central Scrollable Area

The central scrollable area contains all input idea labels. The position of idea labels is constrained to an invisible square grid inside the scrollable area. Overlapping between labels is not allowed. Thus, if an idea label is dragged into an unavailable grid (that already contains an idea label), the move operation of this label will be rejected. The dragged idea label will move back to the original position.

Constraining idea labels by the invisible square grid and disallowing overlapping between ideas have the followings two advantages.

1. **Preventing Duplicate Ideas:** Displaying non-overlapping idea labels allows a user to see all labels at once. All previous labels are regularly recalled by a user while performing

divergent thinking via this system. This design helps the user to recognize all previous labels, which prevents the input of duplicate ideas.

2. **Idea Recycling:** In the same way as the previous advantage, all previously inputted idea labels are regularly recalled to users. The previous inputted information is loaded into the users mind, and therefore the idea recycling process, which is a thinking process derived from the new ideas by combining or altering the previous proposed ideas, is dramatically promoted.

Note that the maximum number of input idea labels is unlimited. If the available grid is going to run out, the workspace will expand vertically from the bottom.

Idea Sentences Input Textbox

The idea sentence input textbox is located at the bottom of the screen. To add an idea label, the user inputs an idea sentence into that textbox, and then presses the Enter key. Once an idea sentence is input, an idea label is added automatically on the screen. Since the arrangement of idea labels is constrained inside the invisible square grid with no overlapping labels, the new input idea is placed on the top-most and left-most vacant grid.

4.2.5 Comparison between the Eureka! and the Previously Proposed Divergent Thinking Support Systems

Eureka! is mainly inspired from the "Hasso-Tobi 2" [57], which is a multi-user collaborative divergent thinking support system. The comparisons between divergent thinking support systems especially the user interfaces of "Eureka!", "Hasso-Tobi 2" [57], and "Idea Expander" [58] are summarized in Table 4.1.

Table 4.1: Comparison between three divergent thinking support systems

Interface	Environment	Type of Suggested Information	Maximum Number of Users	Focusing on Topic	Idea Labels Holding Area (% of Screen)	Can Recall Previous Suggestions
Eureka!	Web	Text	1	Yes	75%	Yes
Hasso-Tobi 2	Web	Text	10	No	55%	No
Idea Expander	Client	Image	2	No	30%	No

4.3 Gugeek: Realtime Multi-User KJ-Method Groupware

4.3.1 KJ-Method Groupware

As stated in the Chapter 2, the KJ-Method, a popular creativity technique in Japan, has been proposed to elevate these four sub-processes by its four basic procedures: (1) Label making; (2) Group Making; (3) Chart Making; and (4) Explanation.

Label making encourages practitioners to write all possible ideas for solving a problem onto a set of labels (divergent thinking). In group making, practitioners cluster the set of ideas that seem to belong together as a group and make a group title to describe the essence of all members (convergent thinking). In chart making, all groups are spatially arranged, based on their association, on a large sheet of paper called a KJ chart (idea crystallization). In explanation, the chart is thoroughly described by investigators (idea verification).

With the emergence of modern computer and information technologies, KJ-Method support systems, and computer systems aimed at increasing a users creative thinking capability by integrating all or part of the traditional KJ-Method creativity technique into such system, have been proposed by many authors. A KJ-Method support system mostly consists of two main components:

1. Idea workspace (KJ-Charts); and
2. The set of hardware or software tools for supporting each step of the KJ-Method.

KJ-Method groupware; multi-user computer groupware that support the creativity thinking of a group of members by integrating all or part of the traditional KJ-Method creativity technique, have been recently proposed. An additional component required for a KJ-Method groupware from just a single-user KJ-Method support system is a share session that allows all members to work together, while located remotely from each other. The KJ-Method groupware has communication systems that propagate group interaction among team members for cooperating in the group KJ-Method process in the shared collaborative workspace. Thus, a KJ-Method groupware consist of three main components, as follows:

1. Shared idea workspace (shared KJ-Charts);
2. A set of hardware or software tools for supporting each step of the KJ-Method;
3. A set of hardware or software tools for group communication and group interaction among team members.

In the pursuit of an efficient multi-user creativity support groupware, “Gugeek” is proposed. Gugeek is developed from the single-user divergent thinking support system “Eureka!”, described in the previous section. According to the three levels of an idea support system classified by Young [5], Gugeek is designed to be supported at the generative level, which includes secretariat support level capability, providing the framework-paradigm support level by guiding users to follow the procedures of the KJ-Method. This automatically generates associative information as the suggestion list for promoting user creativity as the requirement in the generative support level. For the rest of this chapter, the secretariat level and the framework-paradigm level of support is explained in detail, including the ways to store, manage, display, and arrange users thoughts, manipulate the group session, distribute/organize the shared information to teammates, and organize two distinct stages of thinking in the KJ-Method. The design and architecture of the Gugeek support groupware are also discussed.

The supports of Gugeek are categorized as in the generative level, which automatically gives appropriate associative information for promoting divergent thinking or automatically proposes the category title for promoting convergent thinking in the group KJ method, and are described in detail in Chapter 5 and Chapter 6 respectively. However, before diving into the Gugeeks architecture and design, the previously proposed KJ-Method support systems and groupware are described as follows:

4.3.2 Groupware Architecture

The Gugeek consists of three main components, which are JavaScript Web clients, a Web server, and a BigTable data store. The architecture of the Gugeek groupware system is shown in Figure 4.3.

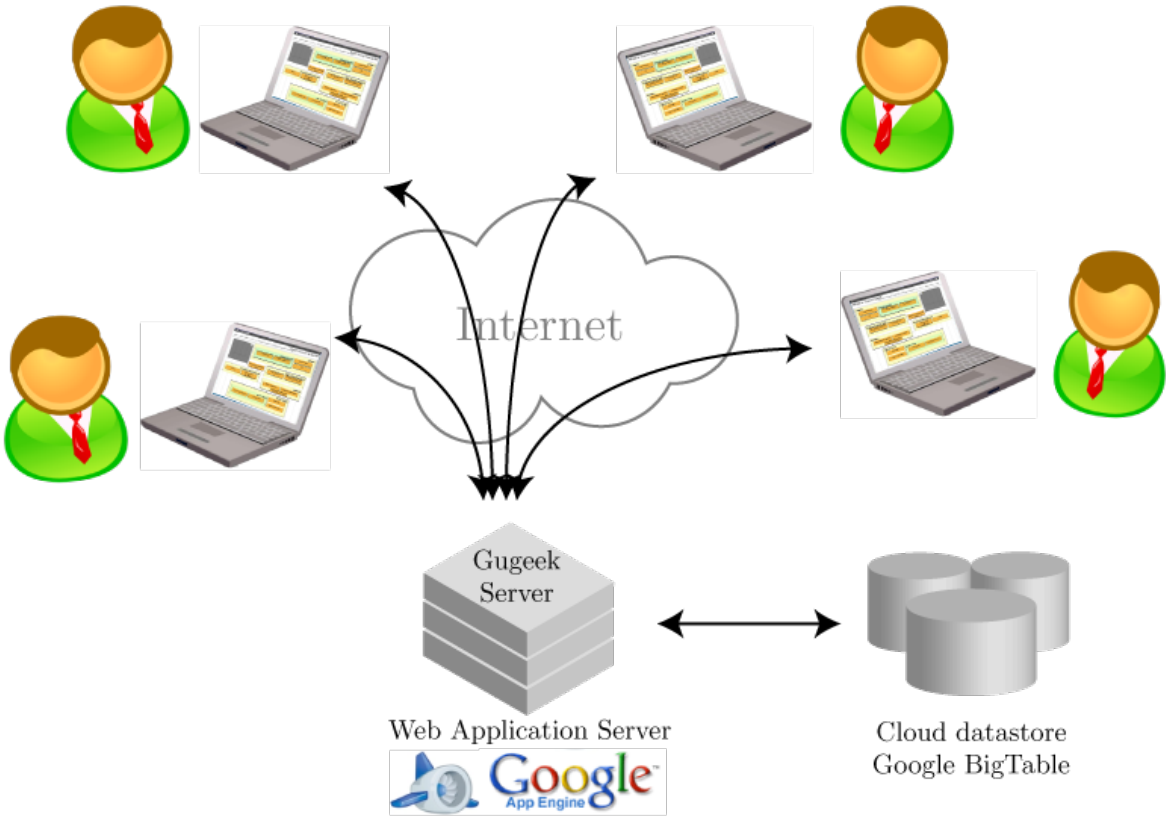


Figure 4.3: Architecture of Gugeek groupware

Three Main Components

JavaScript Web Client: An instance of Web client application is downloaded and executed inside a users Internet browser when visiting the systems Web site[2]. The Web Client application is a JavaScript Web application compiled and optimized from the Java codebase by using Google Web Toolkit library[3]. The supported Internet browsers are Google Chrome, Safari, and Mozilla Firefox. Since Microsoft Internet Explorer did not comply with HTML, Cascade Style Sheet (CSS) or JavaScript standards, the Web Client application could not run successfully inside the Microsoft Internet Explorer.

Web Server The Web server, which is written in the Java programming language, is deployed on the Google App Engine cloud server[4]. The Web server, in fact, is a Java Servlet (a kind of

[2] `http://www.gugeek.com`
[3] `https://developers.google.com/web-toolkit/`
[4] `https://appengine.google.com`

Java application) written for responding to any network requests from the enormous amount of Web clients and storing/retrieving data from the data store.

For any real-time groupware, the real-time persistent bidirectional connection between client and server (streaming) needed to be established. Due to the limitation of the original HTTP protocol, allowing only unidirectional communication (pull requests) from client to server, the generic Web technology could not fulfill this requirement. External libraries are needed. Thus, the push technology, such as the Channel API in the Google app engine, is used to create a bidirectional persistent connection between the Web clients and the Web server.

BigTable Datastore All states, information, written sentences, ideas, and data of all KJ-Charts are persistently stored in the BigTable data store. The BigTable data store is a robust NoSQL schemaless object data store, providing a query engine atomic transaction features. It is a scalable storage, which distributes at least three copies of data stored in different locations under the Google cloud infrastructure network. Due to its strong reliability and efficiency, the BigTable data store is chosen for this task. The Gugeek Web server queries information from BigTable by using the Java Persistence API 2.0 (JPA 2.0) library.

Flow of Group Interactions

Any actions generated by a user for a client, e.g., inputting a new idea, will be transformed into a request and sent to the server. Once the Web server receives a request, it processes that request, queries/updates information from/to the data store, creates a response, and propagates the response to all members of the team. Once a browser receives the response, the user interface is updated, e.g., the new label is shown on the screen, which allows members to interact and collaborate in the group creativity tasks.

4.3.3 User Interface

The user interface of the Gugeek groupware is shown in Figure 4.4, consisting of several components as follows:

Title Topic/Problem Statement

As in the Eureka! support system, a large title topic is shown at the top of the screen to prevent off-topic thinking. An administrator of the group session or an assigned user can change the title topic by clicking on that title topic, filling in the new title topic and clicking the Confirm button to change. The updated title topic is propagated to other users and the system provides notification of the change.

Idea Labels and Categories

The KJ-Method, unlike other creativity techniques, focuses on the hierarchical structure of a concept that allows practitioners to see the bigger picture of the solution from multiple pieces of data and information. Idea labels and categories, which represent the hierarchical structure of the concept, are the central content of every KJ-Chart and are at the heart of the KJ-Method. Thus, they are the most important components of the user interface of the KJ-Method groupware. The user interface design of this part is aimed at being as intuitive as possible to the user.

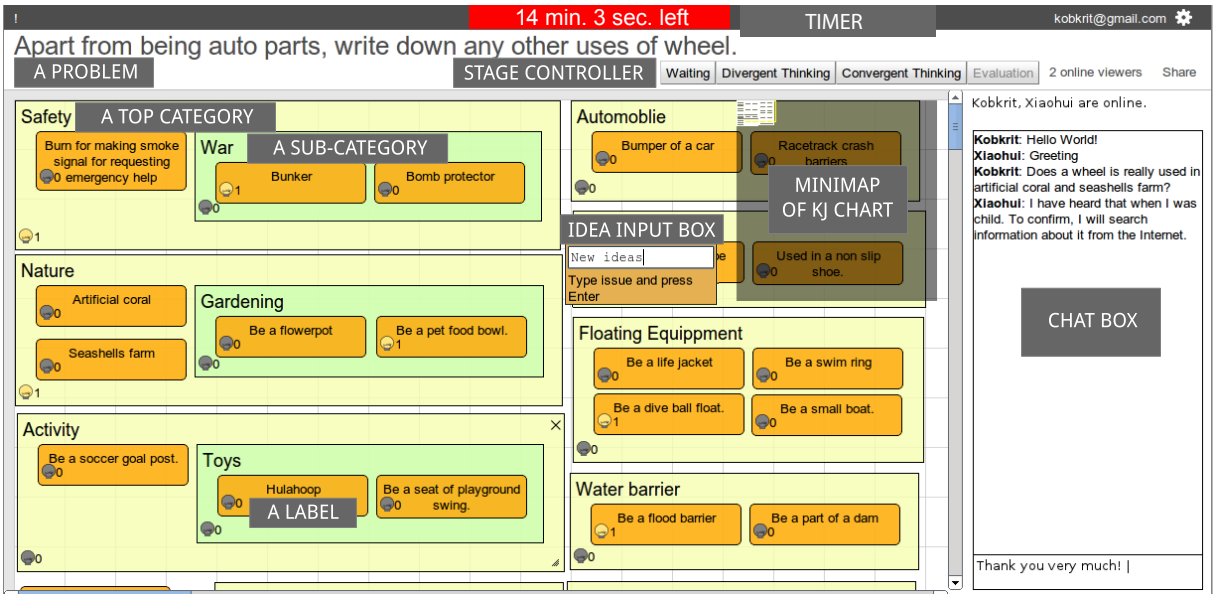


Figure 4.4: A Gugeek screenshot and description of functions

Presentation The graphical user interface components for idea labels and categories in the Gugeek groupware are shown in Figure 4.5. An idea label is represented by a round-corner orange rectangle panel. A category is represented by a sharp-corner rectangle panel. Note that the background color of a category panel represents the depth of that category. The background colors of each category level have a unique color, which is clearly different from adjacent levels. For example, the first, second, third, and forth level categories have yellow, light green, dark green, and light blue background colors respectively.

All objects (idea labels and categories) located inside the category panel are members of that category. A category can contain an unlimited number of labels or categories. The maximum depth level of a category is unlimited.

Operation To input an idea label, a user can click any free position (white grid paper) on the scrollable workspace by adding a label onto that position. A dialog box with an input text box is shown in Figure 4.6a. A user types sentence messages into such text box and presses the "Enter" key or clicks outside the dialog box to put a new idea label in that position.

To edit an idea label, a user clicks on that idea label. Then, a dialog box with a text box containing the content of the idea label, reappears. A user can modify the content of that idea label. The content of that idea label is modified when the user presses the Enter key or clicks outside that dialog box.

To move an idea label, a user drags it by mouse and drops it in an available space on the shared workspace as shown in Figure 4.6b. In the Gugeek groupware, there is no constraint in arranging idea labels, as in the Eureka! support system. All idea labels can be freely placed anywhere on the shared workspace.

To group two idea labels under a new category, drag an idea label and drop it over another label as shown in Figure 4.6c. A new category contained in these two idea labels with an empty category title is created. A user can modify the category title by clicking on it.

As with an idea label, a category can be renamed by clicking on that title and inputting the new category title as shown in Figure 4.7a. Similar to an idea label, a category can be moved by

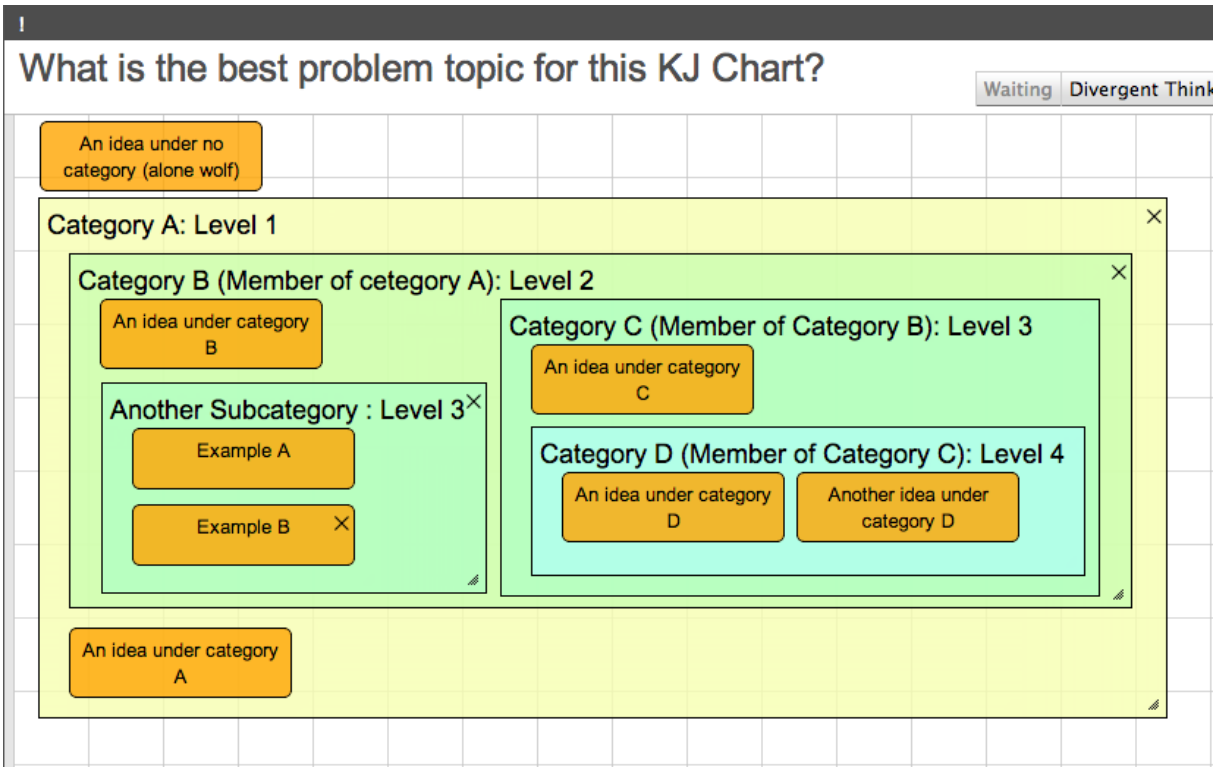


Figure 4.5: The User Interface of Labels and Categories in Gugeek

the drag-and-drop operation. A category can be resized by the drag-and-drop mouse operation using the panel at the bottom-right corner of that category as shown in Figure 4.7b. At the bottom-right corner of a category, the resize symbol is visible if that category is resizable. To remove an idea label or a category, a user simply clicks on the small close sign (the X shape) at the top-right of that object as shown in Figure 4.7c.

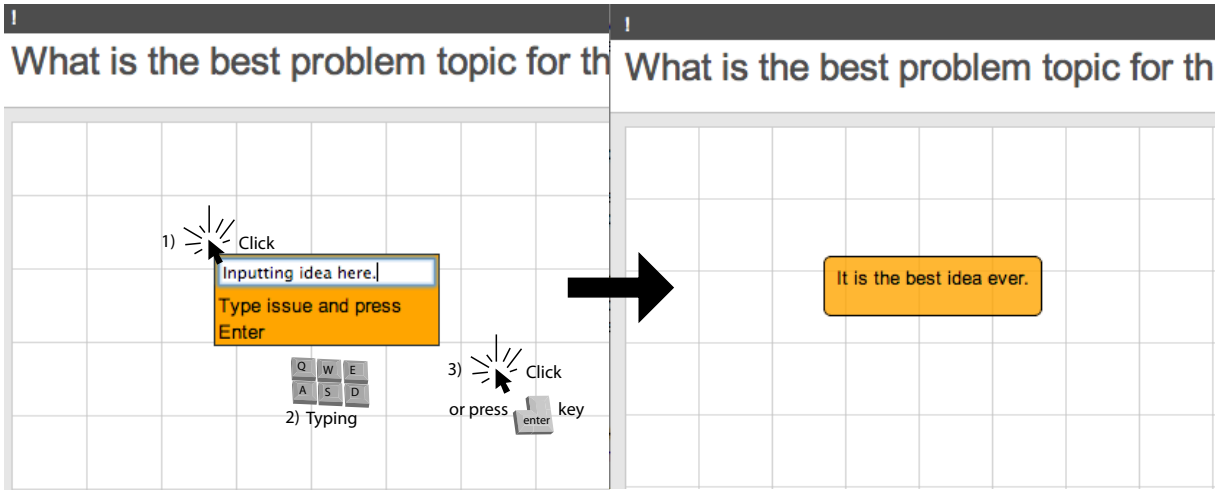
To add more members into an existing category, a user drags an existing idea label or another existing category and drops it into the free space on the panel of that category as shown in Figure 4.8a. In a similar manner, to remove a member from an existing category, a user drags a member object and drops it in an available space on the shared workspace or on another category panel as shown in Figure 4.8b. To ungroup a category, a user drags out all member objects from that category. When a category has no more members, it will be erased automatically.

Minimap

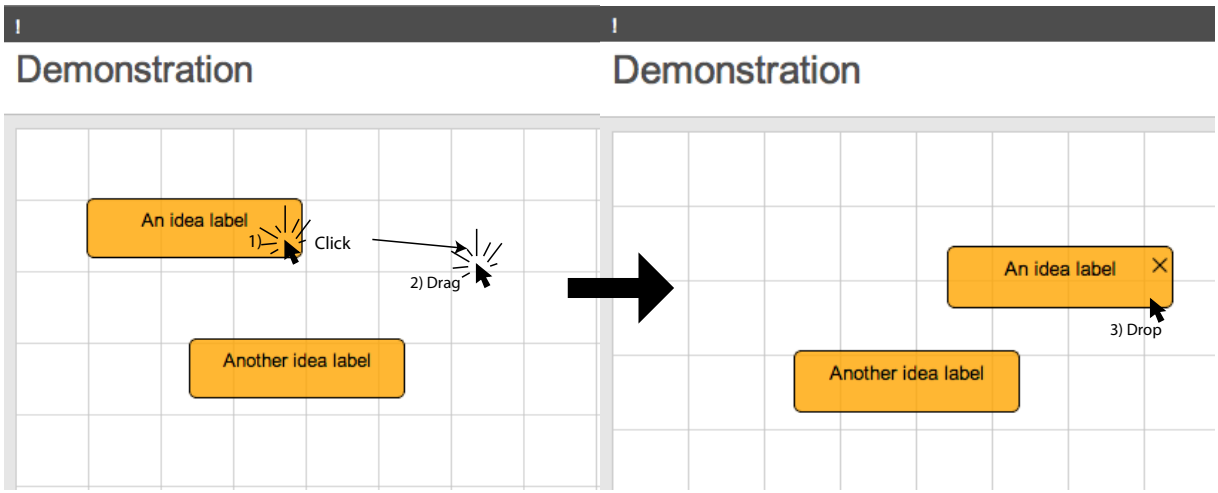
To provide a big picture of the KJ-Chart and make it easier to scroll to a users viewpoint on the shared workspace, minimap, which is the outline summary of the KJ-Chart as shown in Figure 4.9, located at the top-right of the screen is a component of the Gugeek groupware. minimap contains two wireframe shapes, which are:

1. **Viewport Box:** The yellow horizontal rectangle inside the minimap represents the size and position of the current viewpoint (the perspective size and the scroll position).

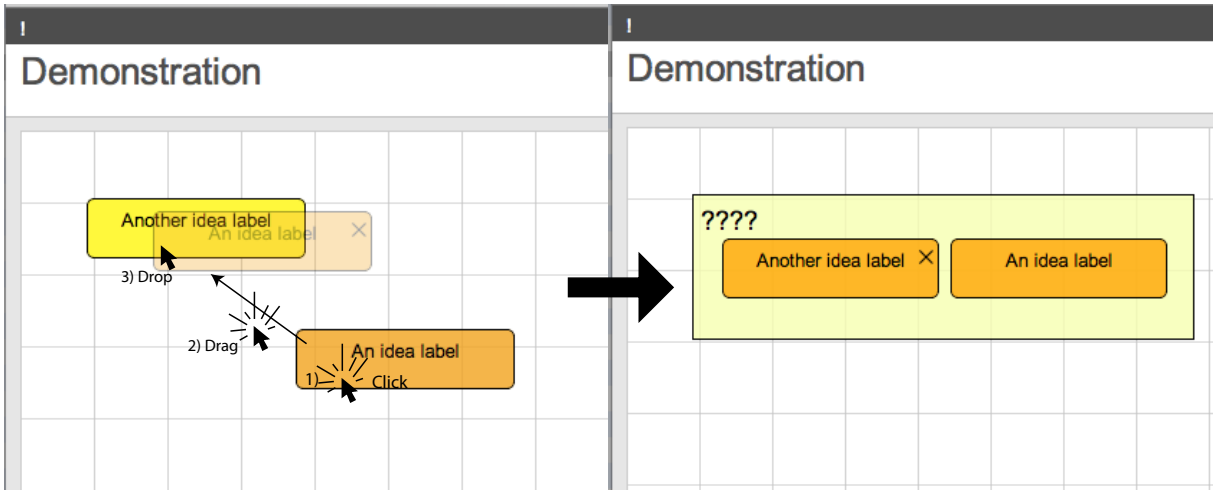
2. **Wireframe Labels and Categories:** The white horizontal rectangle inside the minimap represents the position and the size of both idea labels and categories.



(a) Adding Label

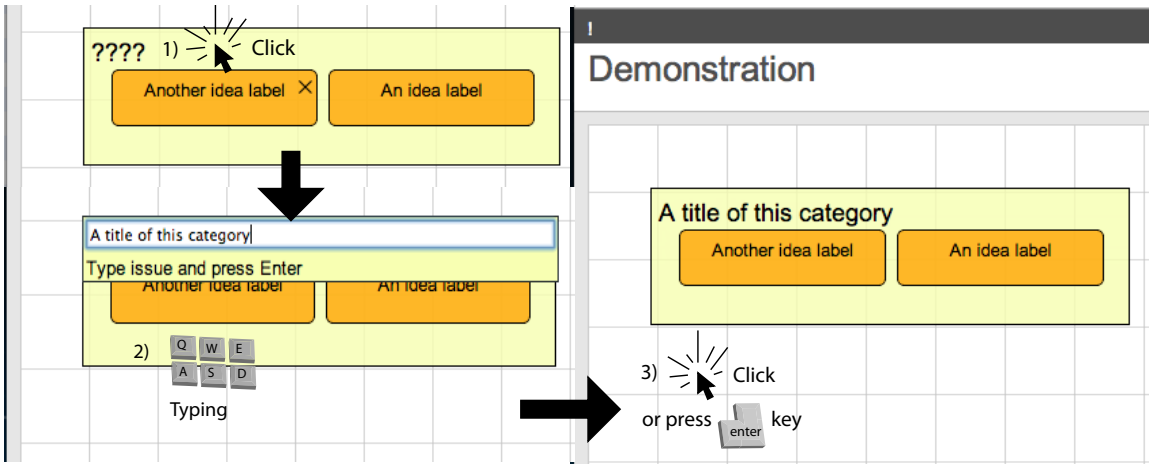


(b) Moving Label

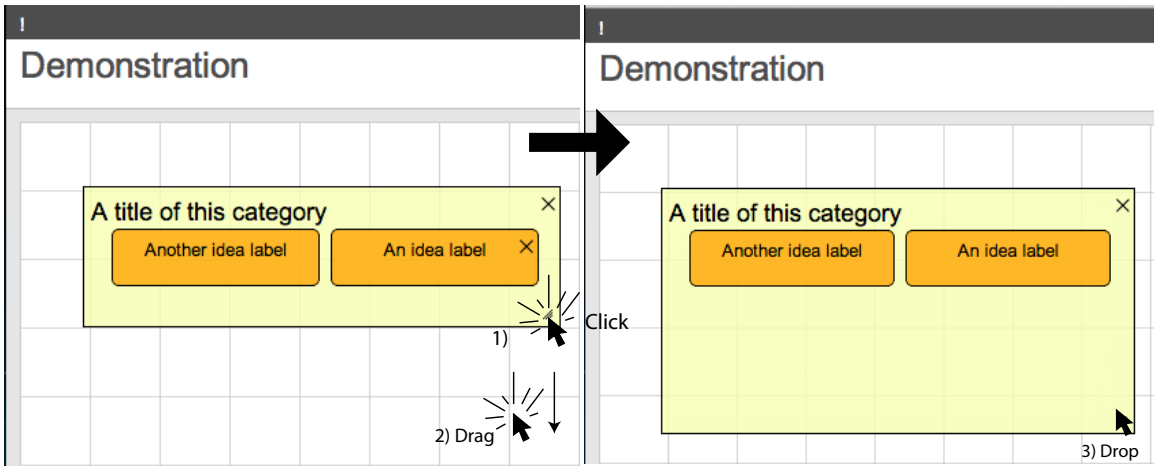


(c) Grouping Label

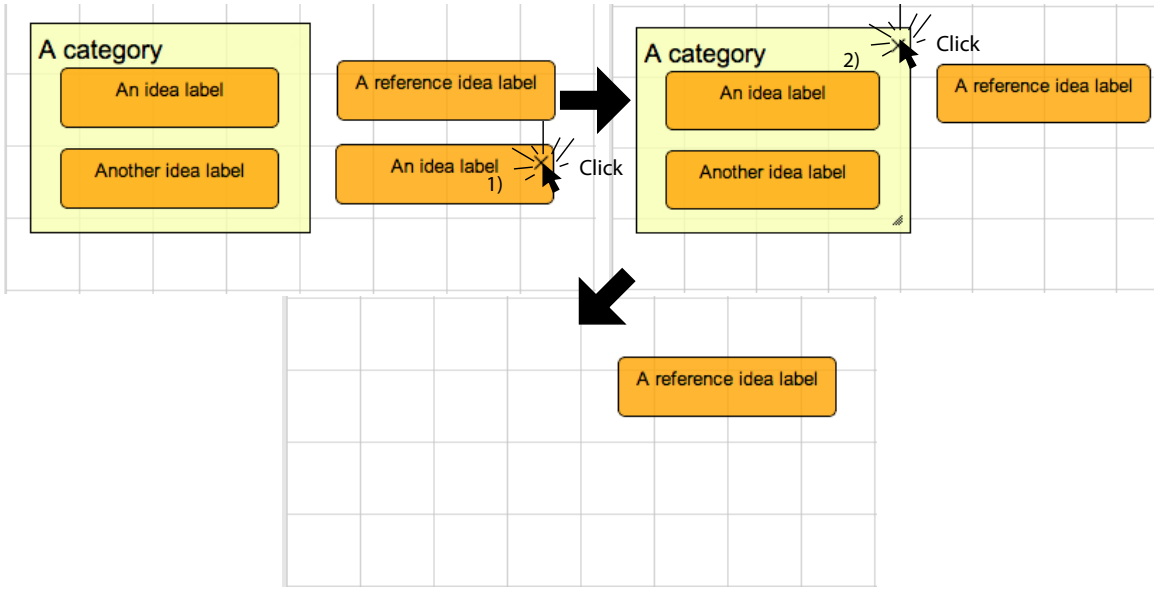
Figure 4.6: Idea labels and category operations (No. 1)



(a) Naming Category

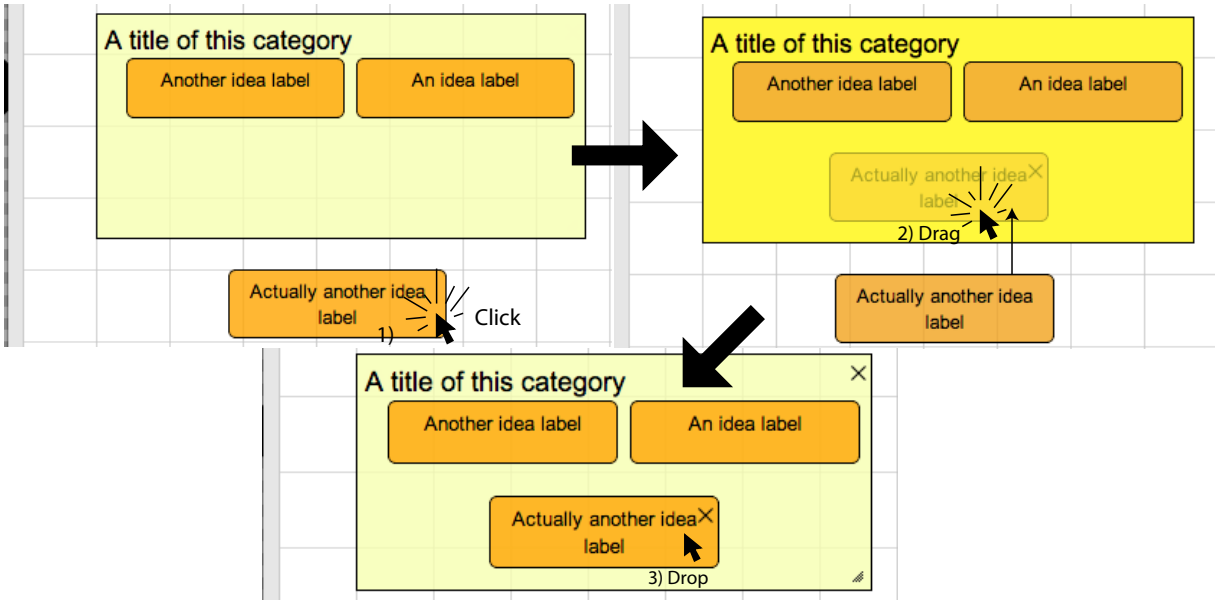


(b) Resizing Category

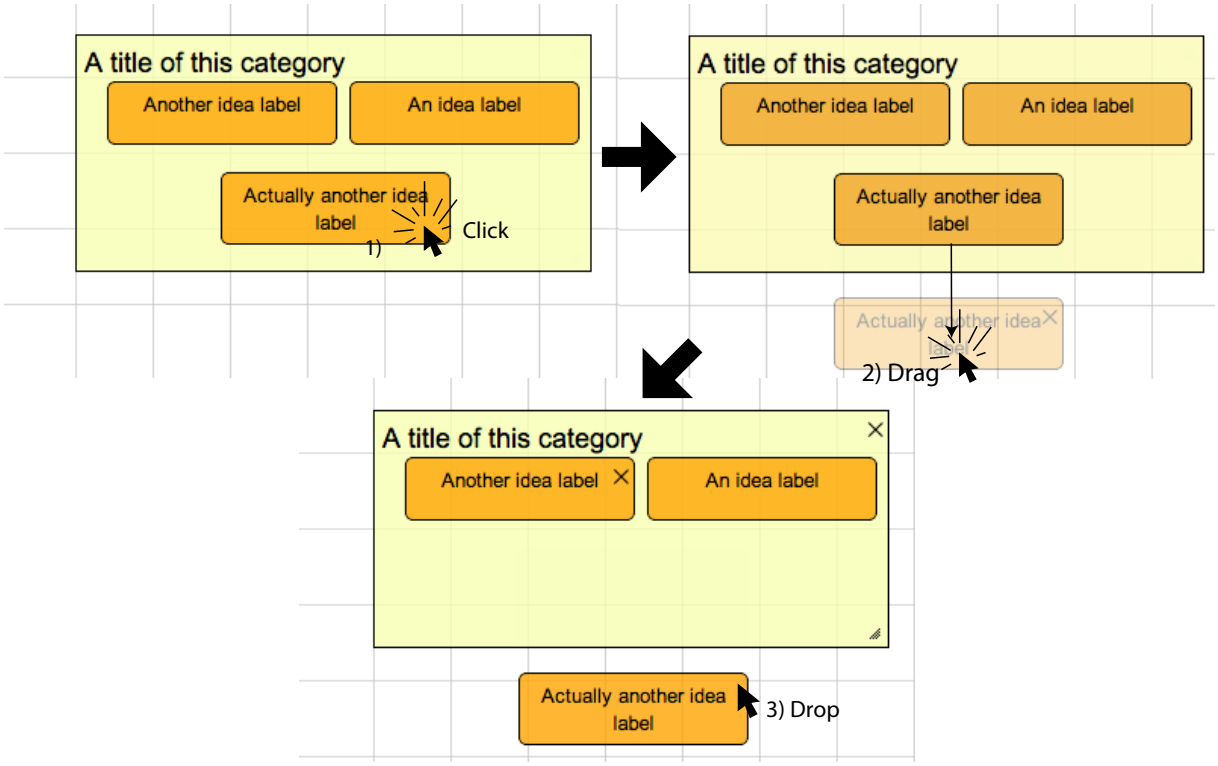


(c) Deleting Label And Category

Figure 4.7: Idea labels and category operations (No. 2)



(a) Adding Label To Category



(b) Removing Label To Category

Figure 4.8: Idea labels and category operations (No. 3)

Once a user clicks any position inside the minimap, the viewport is centered on that position. By using click-and-drag in the viewport box in the minimap, the viewport is moved according to its location.

Shared Workspace

A shared workspace is a rectangle area residing inside the center scrollable window of the Gugeek groupware. It is identical for all members screens. Note that the shared workspace is not expandable as in the Eureka! support system. Although enlarging of the workspace is not possible, the area of shared workspace is large enough for many thousands of ideas. The shared workspace can be scrolled by three methods.

1. Scrolling the workspace by clicking at the scroll bar.
2. Scrolling the workspace by dragging and dropping the viewport box inside the minimap.
3. Scrolling the workspace by clicking any position in the minimap for centering the workspace at that point.

Tele-pointing

Tele-pointing is a system for broadcasting a reference point or a position in the shared workspace to all members in the team. It can be used for calling for the attention of all members in a team to that specific location in the KJ-Chart during the group session.

In the Gugeek groupware, users can flag a reference point at any location on the shared KJ-Chart. The flag icon will be shown in both the group document and minimap. The sound notification will be played to get their attention.

The tele-pointing system is shown in Figure 4.9. The exclamation symbol in the minimap represents the specifying location. It blinks for 5 seconds and fades over time. The alert sound is played according to the blinking of the exclamation symbol.

List of Online Participants and Chatbox

According to Figure 4.4, the list of online participants and the chat box located at the right of the screen allows users to acknowledge online participants and for discussion with other online members. The list of online participants and the chat box can be hidden to increase the area of viewport.

Stage Controller

As stated in the Chapter 2, KJ-Method has four basic sub-procedures (stages): (1) Label making (Divergent thinking), (2) Group Making (Convergent thinking), (3) Chart Making (Idea crystallization), and (4) Explanation (Idea verification).

In each sub-procedure, the activity among group members is different. The stage controller, located at the right-top of the screen (as shown in Figure 4.4, is for displaying and changing the current group activity (the sub-procedure) during the group KJ-Method session. The buttons for changing stages are visible to only the administrator of a group session. Once he clicks on one of the four buttons (each representing a group stage), the group stage is changed according to the clicked button.

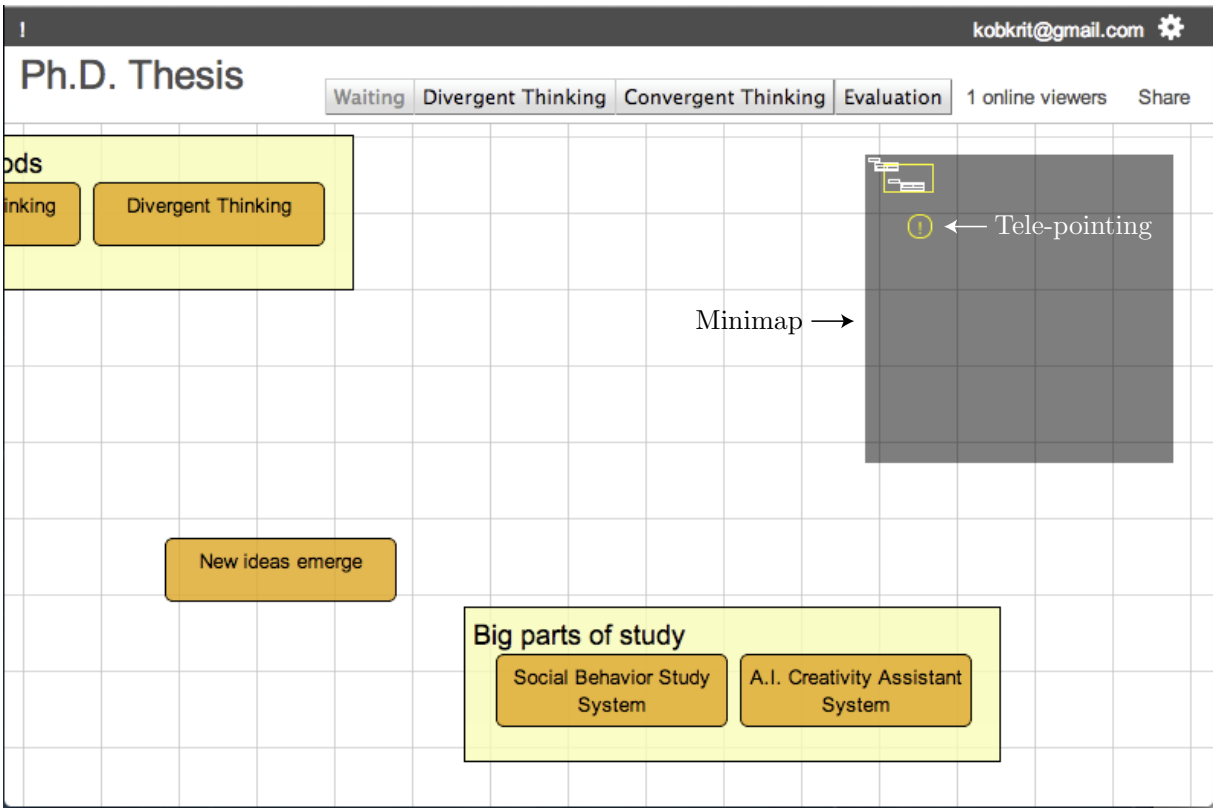


Figure 4.9: Minimap and tele-pointing system

If the administrator is not present, the stage is changed according to the pre-defined period. The time left before changing to the next stage is shown as a time countdown at the top of the screen. Detailed information about the working four stages will be described in Section 4.3.4.

Time Counter

The time counter, a red bar located at the top of the screen in Figure 4.4, is for displaying the time left before changing to the next thinking stage. When the time has almost run out, the time counter starts blinking and playing the warning sound.

4.3.4 Workflow of Gugeek

To start using the system, a user first enters the system by visiting the Gugeek Web site [5] on his Internet Web browser application. Then, the login page is shown to the user. A user enters his username and password for entering to the groupware system. Once he is successfully logged in, the index page of the system, showing the list of the opening group sessions, is displayed as shown in Figure 4.10.

The opened group KJ-Chart sessions are publicly listed at the index page of the system for showing and allowing users to join. The group KJ-Chart sessions are classified on the basis of topic category. At this point, users can choose between joining an existing group KJ-Method session and creating a new group session. If users want to join an existing group KJ-Method session, they click on one of the group KJ-Chart sessions . If users want to create a group

[5] `http://www.gugeek.com`

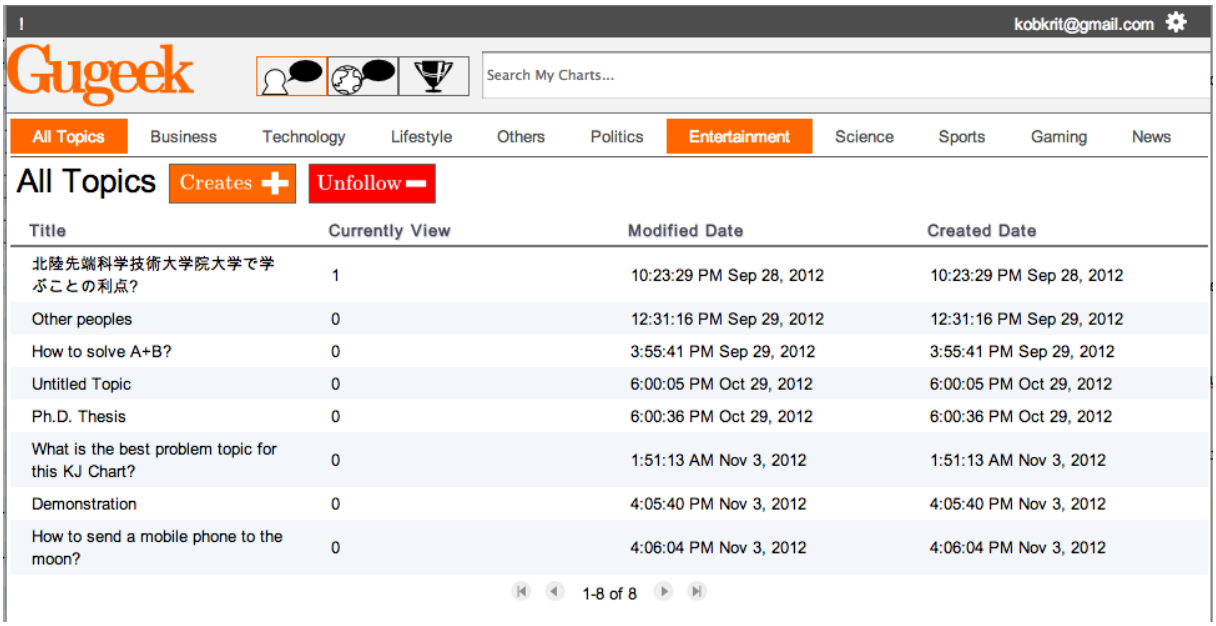


Figure 4.10: Index Screen of Gugeek Groupware

KJ-Method session, they click on the create button at the top-left of the index page as shown in Figure 4.10. The user will then go forward to the group session screen which is shown in Figure 4.4.

If the group session is newly created, the dialog box for inputting the problem topic is shown. After that, the group session will sequentially perform the following stages:

1. **Waiting Stage:** Waiting until all members of a team enter a shared KJ chart and are ready to start.

2. **Label Making Stage (Divergent Thinking):** All members perform divergent thinking and input their ideas into the system. An idea input box is shown when clicking on any available space in the KJ-chart. Users can edit a label by clicking on the surface of that label and can move a label using the drag-and-drop mouse operation. Only the operations of creating, editing, deleting, and moving labels are allowed.

3. **Category Making Stage (Convergent Thinking):** All members perform convergent thinking by grouping existing labels appearing on the shared KJ-chart. A grouping operation can be done by dragging a label into another label. After that, the new category containing both labels is displayed. A sub-category is created by dragging one category to another category or creating a new category inside an existing category. The only operations allowed are: grouping, un-grouping, putting in/out of a label/category to/from a category, resizing a category, and naming a category. Note that a category can contain an unlimited amount of sub-categories. Sub-categories can recursively contain other sub-categories with unlimited depths.

4. **Evaluation Stage:** Once the drawing is finished, all members vote for their favorite ideas by clicking at a bulb icon (to turn on the bulb) at the bottom-left of the favorite ideas/category. The number next to the bulb icon indicates the total number of users who like this idea/category. The group sessions during the evaluation stage are shown at Figure 4.4.

All modifications and group interaction on the shared KJ-Chart will be saved automatically by the server. If a user is disconnected during the group session, he/she can re-connect to the group session later.

All information of the group KJ-Method session is consistently saved in the data store. In the case that all users are disconnected from the group session, the group session can be resumed anytime.

4.3.5 Comparison Between Gugeek and Previously Proposed KJ-Method Groupware

In this section, the comparison between Gugeek, and the previously proposed KJ-Method groupware, namely GrIPS, D-ABDUCTOR, and GUNGEN are described. The summary comparisons are shown in Table 4.2.

4.4 Key Issues of Groupware and Interface Designs

The following section explains the key issues of the groupware and user interface designs and their decisive reasoning. The design alternatives are also included, which is very helpful for interested readers who wish to design a collaborative groupware in the future.

4.4.1 Group Process

The KJ Method's group processes are designed as follows:

1. Determine a problem topic and wait for other members to join the system.
2. Members separately propose their ideas (Label Making).
3. Members read all proposed ideas.
4. Members separately group similar items (Label Grouping) (iteration i^{th}).
5. Members observe every group (iteration i^{th}).
6. Members separately name each group (Label Naming) (iteration i^{th}).
7. Members observe the grouped chart.
8. Members separately make the chart (Chart Making).
9. Members observe the final chart, discuss it and achieve consensus (Explanation).

The KJ Method's group process is illustrated in Figure 4.11. The odd group states are synchronous states. These require every member to perform a group task at the same time. The even group states are asynchronous states. Members can perform a group task individually at any given time.

4.4.2 Static and Dynamic Parts

The KJ Method's group process contains both static and dynamic parts [60], listed as follows:

Static Parts

Group Goal Obtaining the best (and creative) solution to a problem topic and gaining consensus.

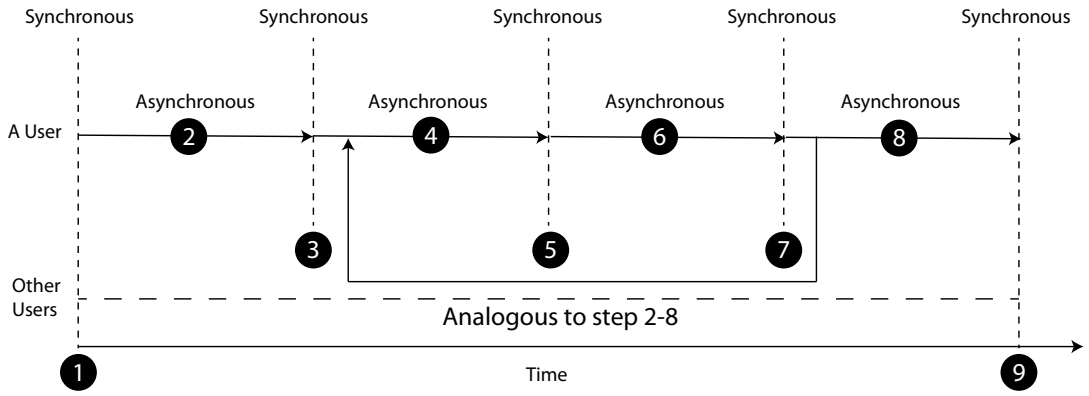


Figure 4.11: The KJ-Method group process

Table 4.2: Comparison among four KJ-Method Groupware

Features	D-ABDUCTOR	GrIPS	GUNGEN	Gugeek
Platform	Desktop	Desktop	Desktop, Web, Tabletop, Mobile	Web
Supported Network	LAN	LAN	LAN, Internet	Internet
Supported Sub-procedures	Label Making - Chart Making	Label Making - Chart Making	Label Making - Explanation	Label Making - Chart Making (Partial)
Team Communication	Voice	Voice	Text Chat, Video, Voice	Text Chat, Voice, Notification
Information Storage	File	File	Database	Server Database
Storage Format	Simple Language	Simple Language	Wadaman, MySQL(RDMS)	BigTable(NoSQL)
Creativity Assistants	Automatic Diagram Drawing	Associative Keywords Automatic Diagram Drawing	Group Consensus Support PDA	Associative Keywords Automatic Group Title Making
Supported Chart Representations	Text, Adjacency Graph Inclusion Graph	Text, Adjacency Graph Inclusion Graph	Inclusion Graph Compound Graph	Inclusion Graph

Group Organization Members should be people from different parts of the organization, or people who have different perspectives on the problem topic. Every member should be an active participant, respect democracy, respect other opinions, and have the confidence to express their own ideas.

Group Protocol Group protocol consists of two types:

1. **Technical Protocol:** is the technical procedure of interaction, cooperation, and communication among team members. A good example is that some groupware might limit the allowed maximum length of time a participant can speak during the group session to five minutes; a groupware application automatically turns off the microphone if the limit is exceeded to prevent the domination of interaction. In Gugeek, every member is free to talk and free to perform any KJ-Method operation without restriction.

2. **Social Protocol:** is a social convention of interaction, cooperation, and communication among team members in a team based on their social rule or status. For example, participants must require permission to speak from a team leader.

   Social rules and status can cause creativity blocking and obstruct the group goal. Some of those ideas can be very useful for achieving the teams goal but are disliked by a team leader. The expression of these good ideas is to be avoided during the group session because of the fear of termination of their employment. With the hope of reducing creativity blocking because of external negative influence, Gugeek does not show any personal identification of team members performing a groupware operation. All interactions (including expressing ideas) except verbal and text discussion are completely anonymous.

Group Environment Since our system is a Web application, participants can be distributed geographically in any location on any portable computer or mobile device that can access the Internet. Since all ideas and information is durably stored in the cloud database, the KJ Chart can be retrieved and edited any time. Once there is an update, an immediate notification is sent to every member of the team. If a user in the team is not online, the notifications are queued at the Web server and are released to the client application of that user when he or she comes back online.

Dynamic parts

Group Document Group document is a document that stores information yielded from the group process. In this case, it is a KJ-Chart.

Group Activity This is an activity taking place during the group process. In our KJ-Method groupware, all possible activities are list as follows:

1. Label Making
2. Label Grouping
3. Label Naming
4. Chart Making
5. Explanation

For more information about these KJ-Method group activities, see Chapter 2.

Group Session The group session is a part of a group activity. It can be a synchronous or asynchronous operation and may require the performance of one participant or many participants. For example, the Label Making group activity includes the following group session:

1. Each participant asynchronously thinks and writes down a new idea.
2. All participants synchronously observe all idea labels.

Group State This is the current state of the group activity. In this case, it can be one of five group activities.

4.4.3 Group Process Models

Group Process Models can be distinguished into three types:

1. The centralized group process model.
2. The distributed non-replicated group process model.
3. The distributed replicated group process model.

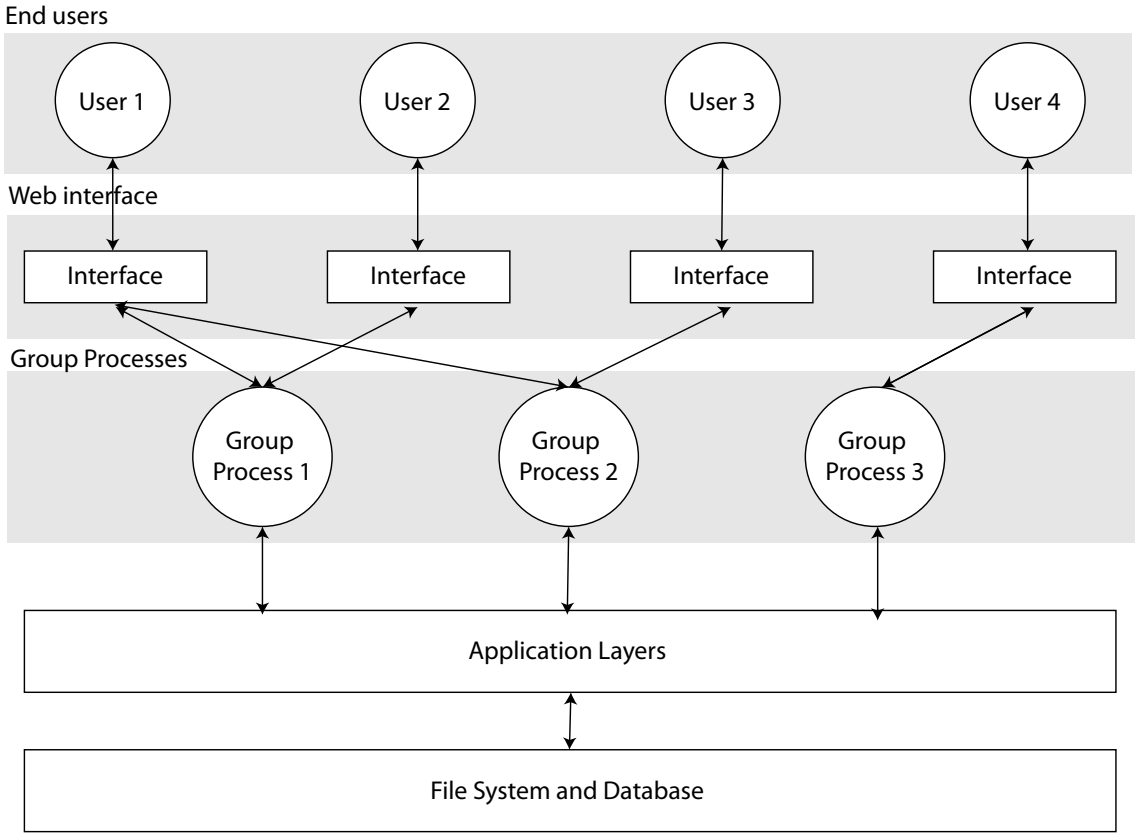


Figure 4.12: The group process model of the KJ-Method groupware

The centralized group process model, which is the most simple and efficient model, is chosen for the Gugeek. All data and information are stored and managed at the central database server as shown in Figure 4.12.

The Gugeek is able to manage multiple parallel group processes by separating each group process in the application layer, not the physical connection layer. Thus, a member can join many group processes (many browser windows) using the same computer simultaneously.

All members in the group process retrieve the same set of group documents and group interactions. In other words, the group KJ-Chart, group communication, and group update operations are identical for every member in the same group process. The private memory cache in each members Web application client is copied from the central Web server for buffering a group document to increase the responsiveness of the groupware system.

All group interactions and the latest version of group documents are recorded at the central database server. If new incoming members join a group process after the group activity has been started, they can acquaint themselves with the development of group documents and current group state by viewing the group interaction history acquired from the central database server.

4.4.4 Group Communication

All members in the same group process are able to communicate with each other by discussion, which is a bidirectional many-to-many communication. All group interactions (including chat messages and commands making changes in the group documents) are synchronously propagated from one member to all other members immediately.

If a member disconnects from the groupware (offline), the message for sending updates to him during the offline period is not lost. Since all messages are stored at the central database, he can asynchronously retrieve these messages later once he comes back online. According to this description, both synchronous and asynchronous communications are used in the Gugeek.

Transport Mechanisms

Every group interaction issued by a member is propagated from that members Web client application to the central server. Once the central server receives the update, it filters and propagates it to all team members. The central Web server is a router for choosing eligible users to be the receivers (typically all online users staying in the same team of the sender). All messages are delivered through the central server, which makes concurrency and access control simple and efficient.

Another possible method is direct propagation. The direct propagation method is propagating group interactions from a members computer directly to the computers of the rest of the members. The messages do not need to pass through the central Web server. The transmission time for delivering messages is a half that of the previous method, but the concurrency and access controls are extremely complex.

For our proposed groupware, the transmission via a central Web server is chosen over direct propagation for two important reasons.

Same Origin Policy Problem Since our groupware is an AJAX Web application and runs on a Web browser, the same origin policy is forcefully applied. The same origin policy is an important Web security concept defined by the World Wide Web Consortium (W3C), the concept is defined as follows [61]:

1. An origin point is part of a full URL. It is defined by the protocol, host, and port of a URL. For example, document A located at `http://www.example.com/docA.html` and

document B located at `http://www.example.com/somedir/docB.html` is considered to be retrieved from the same origin point because of the following three reasons:

(a) Both URLs use the same protocol: http;
(b) Both URLs use the same host name: example.com;
(c) Both URLs use the same port: 80;

If at least one part of this information is not the same, both documents are considered as originating from different origins.

2. If a script within a document tries to access a document that has a different origin (typically automatically accessed by a browser-side programming language, such as JavaScript), access will be restricted. Since this technique allows Cross-Site Scripting attacks (CSS) and Cross-Site Request Forgeries (CSRF), which are popular types of computer vulnerability security nowadays, the same origin policy is enacted to prevent vulnerability.

For our groupware, a user accesses it by visiting the Web application from the central Web server (`http://www.gugeek.com/index.jsp`). While the user is visiting ourWeb site, the JavaScript-based Web application is downloaded to the users Web browser. Our Web application is prevented from directly accessing another members computer to obtain their updates, and has access only to the central Web server (`http://www.gugeek.com`) . Due to technical and security issues, direct communication cannot be implemented.

Complex concurrency control Before a central Web server forwards a group interaction issued by a user to every member in the same group process, the concurrency control, which is the procedure for checking that the understanding of the group state remains consistent (identical) for all team members before making changes. This can be easily checked by comparing it with a golden copy held at the central Web server. Moreover, having the central Web server as the central hub of group interaction delivery means it easy to keep track of all group documents, making version control possible.

4.4.5 Concurrency control

Exchanging group interactions, information, and updates usually happens in the group process. To keep the group state consistent for every group member, a concurrency control is needed.

Our groupware operates by strict interpretation of concurrency control. All intermediate group states must be identical for all members during a group process. To avoid concurrency conflict, the golden copy (the master copy of the latest correct group state of a group process stored in the central Web server), is used for verifying an incoming group interaction from clients, whether it is generated from the latest same group state or not. This verification is called “Consistency verification”.

To perform consistency verification, the latest group state under the sender knowledge must be attached to the group interaction package sent from clients to the central Web server. To reduce the size of the group interaction package, instead of attaching full information of the group state, the 128-bit string representation (message digest) of the group state produced by a MD5 hash function is attached to the group interaction package. Once the central Web server receives an incoming group interaction package, the message digest representing the group state under the sender knowledge is compared to the message digest of the golden copy located in the central Web server.

If both copies do not match, inconsistency occurs. The central Web server will reject this group interaction package. The golden copy will not be updated. This group interaction, generated from the outdated group state will not be propagated to other members. For solving the inconsistency, the central server issues a full reloading command to the sender of the package. The golden copy is completely copied to that user.

However, if both group states are matched, inconsistency does not occur. The central Web server will accept this group interaction package. The golden copy will be updated according to this group interaction. The central Web server will propagate it to all other members.

Other Aspects

Responsiveness Fast response is a crucial property of successful groupware. Shorter response times means faster group interaction propagation, thus there will be less chance of concurrency conflict.

The groupware client exploits the Web socket feature of HTML5 for receiving updates from the central Web server without making periodically pull requests to the central Web server. The central Web server is programmed to propagate the group interaction to all other users right after the package passes consistency verification for concurrency control. Therefore, the time interval for transferring a group interaction from a user to all other users equals the sum of the time interval for transferring the network package from a senders computer to the server. This means less processing time at the central Web server, and a shorter time interval for transferring the network package from the server to all other users computers. It is approximately about 0.5 second or less. The proposed system is considered to have a fast response but it is slightly different among all members due to the condition of the network connection between the central Web server and each members computer at that time.

Group Interface All members who join the same group process received an identical group document and group state. Once a change occurs, the updates will be sent to all members immediately. The WYSIWIS (What you see is what I see) principle is also applied. In a case where that user is accidentally disconnected from the system, the warning message is clearly displays informing them that the current group document (KJ-Chart) is out-of-date.

Replication of Information Our proposed system yields high responsiveness by caching the group documents on each members computer. The main purpose of caching is to buffer the whole KJ-Chart into the computer memory, dramatically reducing the delay while displaying a whole KJ-Chart. To avoid concurrency conflict, replication of the group document in every members computer must be consistent. There are two typical methods for creating replication:

Reloading the Entire Group State To replicate a group document to a client, this method transfers a whole KJ-Chart from the central Web server to a members computer. The consistent ready-to-use group document is transferred, and does not require the previously stored group document to be updated thus avoiding consumption of high computation power on a members low performance personal computer. The disadvantage is that it consumes high bandwidth and waiting time to transfer such large data. The loading time will be high, which might disable a clients applications for a while. It significantly reduces the responsiveness of the system.

Propagation of the Updates Instead of transferring a big chunk of data, many small group interactions are transferred. These typically consist of detail of performed operations and sender information of those involved in the performance of such operations. The advantage is that as this consumes a low amount of bandwidth and it takes time to complete the transfer, the responsiveness of the system remains excellent. The disadvantage is that it requires computation power on the clients computer to update his out-of-date group document.

In our groupware, both methods are used in certain suitable situations. Reloading the entire group state is used in the following three situations, which demands that the replication process is guaranteed to be consistent.

1. If a member joins the group process for the first time the cache of the group document on his computer is still empty. The fastest way to copy the whole KJ-Chart is downloading the golden copy from the central Web server.

2. If the cache of the group document is too old (¿ 30 versions old than the golden copy in the central Web server),the time involved in reloading the whole KJ-Chart is faster than updating it thirty times bit by bit.

3. If the group document in the clients cache is not consistent or has a concurrency problem, the propagation of the operation is used to maintain high responsiveness.

Otherwise, the propagation of the operations is used for maintaining the high responsiveness.

Robustness Gugeek has an automatic recovery system if the clients computer crashes or there is a communication error during the groupware process. The Web application stored in the members Web browser periodically checks the health status of the communication link between itself and the central Web server (heart-beat signal typically every five seconds). Likewise, the central Web server also checks the condition of communication links between itself and all other clients.

If the Web application on a clients computer crashes or is accidentally disconnected from the network, both the central Web server and clients Web application acknowledge this and execute the fail-safe set of commands to recover the situation. The fail-safe commands consist of storing a queue of group interactions not issued to the disconnected user in the central Web server, sending the disconnect notification to other members in the same group process, and repeatedly tries to reconnect from the clients Web application to the central Web server.

In case the central Web server and database server are down, the database servers which stored all group documents and information are still persistent up until the last minute before the central Web server is down. This is because the database server periodically flushes all group information from the main memory into the hard disk arrays. The database servers have a replication service for safer data storage. All group documents and information backups are stored in at least three different storage locations in different parts of the world, and thus all group documents have an extremely low chance of being completely lost.

Notification Notification helps users to be aware of concurrent operations occurring in the groupware system. There are two different types:

1. **Immediate Notification:** Informing users of updates/conflicts immediately.

2. **Delayed Notification:** Informing users as soon as the asynchronous operations are completed.

In our groupware, once a change in the group state has occurred, notification is fired immediately. By the way, due to delays or problems in the communication link, the notification could be too slow and cause a concurrency conflict during the group session. Once the conflict occurs, two concurrency control schemes can be used: (1) Detection of Dependencies; and (2) Reversible Execution. For our groupware, the detection of dependencies scheme is used.

Detection of Dependencies Conflict can be detected by using a timestamp or version of a group document as additional information attached to the group interaction package sent from a clients Web application to the central Web server. The timestamp or version of the group document is information about the group document under that clients knowledge. In case a delay or error notification occurs, the timestamp at the clients side is not updated. If that client issues any operation based on the out-of-date group document to the central Web server, the timestamp attached with the group interaction package will not match with the timestamp of the golden copy in the central Web server. This means that a conflict has occurred and should be resolved.

For example, Let u_1, u_2 and u_3 be users and L a label in a group KJ-Chart. The following commands sequence occurred as follows:

1. The group process was established. The central Web server sent the initialized command to all users, the timestamp t_0 was attached. All users received the initialized command, and the timestamp t_0 was stored on all users' computers. The central Web server stored the timestamp t_0.

2. The u_1 created a label L in a group KJ-Chart, the timestamp t_0 was attached. The central Web server compared the timestamp attached to the package (t_0) with its locally stored timestamp (t_0). Since they are matched, the central Web server modified the golden group document according to the u_1's command. Once the u_1's command was done, the timestamp t_1 was generated. The central Web server propagated the u_1 command with the timestamp t_1 to all users. All users acknowledged the change, updated their display, and their timestamp to t_1.

3. The u_2 moved the label L in a group KJ-Chart, the timestamp t_1 was attached. The central Web server successfully matched with the timestamp t_1. It allowed the update to the golden group document according to the u_2's command, and then generated the timestamp t_2. It stored the timestamp t_2 and propagated the u_2 command with the timestamp t_2 to all users. Fortunately all users except u_3 successfully received the updated group interaction package. They updated their display and their timestamp to t_2. Since unfortunately the u_3 did not receive the command package due to communication error or delay, he did not acknowledge the change and his timestamp still was t_1.

4. The u_3 deleted the label L, and the timestamp t_1 was attached. Since the central Web server timestamp was t_2, it did not matched. Thus u_3's operation was rejected. The central Web server sent the "reloading" command to u_3. The central Web server notified that the concurrency conflict happened in u_3. The u_3 reloaded the whole group document including the timestamp t_2 and was ready for the group activity again.

Reversible Execution Instead of attaching the time stamp, the undo procedure of an operation is pre-defined. The operations can now be immediately executed without any concurrency checking. If any conflict arises on the golden copy in the central Web server, the undo procedure of conflict operations will be executed, reverting all changes done to the shared KJ-Chart into an older state. Next, the central Web server notifies of the undo and issues the refreshing command to all users. The time on all machines must be globally synchronized to define the order of the incoming operations.

4.4.6 Role of Group Members

The roles of group members influence the group process in the groupware. Roles of group members structure the interaction between team members, or define some administration right in controlling the groupware process for staying on the right track to the team leader. Administration power in the group process of the creativity support groupware can limit group creativity performance, but if this power is absent, the group process might not be able to achieve the group goal; in other words, chaos occurs.

The proposed groupware allows the creation of a group process under one of two visibility modes as follows:

1. **Public Mode:** A group process in the public visibility mode is indexed and listed in the public topic list of our system. Any participant can join the group process and have equal rights to other members in expressing their ideas. Anonymous participants are allowed. Banning disturbing participants and advancing the group state to the next group activity can be done by a majority vote of all joined participants, or by pre-defined conditions (such as advancing to the next state of the KJ-Method group activity if the number of labels exceeds 200 or the pre-defined time interval).

2. **Private Mode:** A group process in the private visibility mode is not indexed in the topic list page. The creator of a group process in the private mode is the first group administrator of the group process. Only a group of participants, who are invited by group administrators, can be joined into the room. The administrator of the group process can later invite more participants, exclude existing participants from the group process, promote someone to be an administrator, and remove some participants from the administrator list. Advancing the group state to the next level can be controlled by one of the administrators of that group process.

The public visibility mode is designed to minimize the usage of administration power to maximize creativity performance by using the power of democracy. For the public visibility mode, the anonymous participant completely removes personal identity and ignores the roles of group organization, and can degrade group creativity performance. For the public mode, personal identity does not need to be known by any participants; in contrast, it is impossible in the private mode because all participants must be identified by at least one administrator. Giving equal voting power (one vote per participant) as seen in the democratic system, removes the requirement for administrators. As in our recommendation the creative problem solving solution using the group KJ-method, via our groupware on a controversial topic, the visibility of that group process should be public.

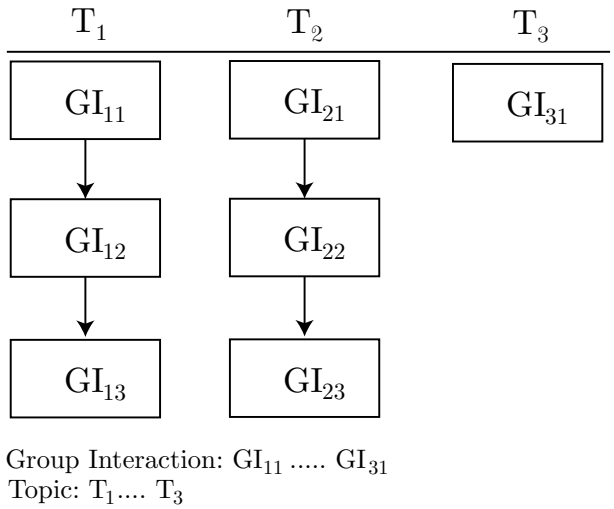


Figure 4.13: Group Interaction Management: Comb Model

4.4.7 Management of Group Interaction

In every groupware, team members repeatedly exchange information with each other (group interaction) for collaborative brainstorming. The proposed groupware, allowing parallel group sessions, requires efficient interaction management for delivery of group interactions to the right group sessions, and systemically stores this information.

Since a group session has a unique topic, group interactions under the same topic must be sequentially progressed in the same session. Group interaction in different topics should not involve each other. They must be entirely separate. Thus the comb model [60] is chosen.

According to the comb models description, a group interaction cannot be shared in two or more topics. If a group interaction should appear in multiple topics, that group interaction must be copied as a new group interaction and sent to these topics. A good example for this model is the channel on the Internet Relay Chat (IRC) [62], and the hierarchical topic structure in Usenet [63].

A topic, (a problem topic in our case), is appointed in each group process. A group process is built once a problem topic is created. Group interaction with a topic (a group process) is not shared with other topics (other group processes).

The topics will be listed on the topic list page. Any interested participants can select the set of the topics (group processes) to join. A participant who joins more group processes will receive group interactions from all group processes, but displayed separately (in different windows).

4.4.8 Management of Share Context

Designing a groupware requires the consideration of share context, which is the multiuser interface for displaying the share context (the progress of the group activity and group document), as well as for allowing multiple group members to efficiently manipulate it. In a traditional group activity (such as a face-to-face meeting), a whiteboard serves as a traditional tool for displaying and cooperating with a share context.

With the emergence of the groupware system, the tool for representing a share context is significantly improved as follows:

1. The space for representing a share context becomes unlimited on the groupware system. It allows us to putt more complex and larger information into the share context.

2. The share context is digitalized and thus it is consistent (no more illegible text), persistent (no more faded characters or lost text), and easy to be distributed (transferred via a computer network).

3. It can be later processed by machine or other intelligence for deeper understanding of the shared context.

4. Not only text or image can be presented in the share context. All multimedia including video, audio, or even Web pages can be included.

Since the groupware system allows us a more rapidly, more deeply and wider share context, the stream of changes in the share context is unlimited. The users awareness in understanding these changes becomes much more difficult (events/changes flooding). Thus, it is the duty of a groupware to be efficient in representing to users in both the share context itself and its dynamic over time.

Share context management is essential for synchronous cooperation. Synchronous cooperation helps users to cooperate on a joint task simultaneously with real-time and awareness of the activities. Group interactions and cooperated activities on the share context must be propagated immediately to every group member to achieving real-time cooperation.

WYSIWIS

The WYSIWIS (What you see is what I see), which is proposed by Stefik et.al [48], is a type of share context management approach focusing on consistent presentation of the share context to all participants. It is typically used in the most synchronous groupware. There are two types as follows:

Strict WYSIWIS It is WYSIWIS in the strictest form. The strict WYSIWIS will display the identical share context on all participants screens, which means that all participants have exactly the same context. Only the public share context is used, thus every participant receives the same view. For better visualization, the strict WYSIWIS is similar to a personal computer which has only one monitor but has many sets of input devices (such as keyboards and mice) and allows many users to operate on that computer simultaneously at the same time. If the share context appears in a scrollable window and a participant/ host scrolls in a window during the group session to view another portion of the share context, the context appearing on all participants screens would also move identically to his scrolling movement.

The advantage of this form is that it allows group members to specify objects by referring to their position via group communication. But problems in group cooperation might arise as follows:

1. **Scroll-War:** A participant scrolls the window to another position in the share context, while other participants read the share context in the original position.

2. **Window-War:** In this case the share context is in a desktop environment, containing movable sub-windows which divide a portion of the screen to display different information. Moving sub-windows while other participants are reading content in them might cause difficulty in the cooperation.

3. **Style-War:** The change in style representation of the share context will actively affect other users immediately. Changing the texts size to make it smaller, while other participants are reading it, disturbs and causing difficulty in cooperation.

The strict form of the WYSIWIS seems to have more disadvantages than advantages, and thus, a more relaxed form of WYSIWIS is proposed.

Relaxed WYSIWIS Typical modifications from the strict WYSIWIS to the relaxed WYSIWIS are listed as follows:

Private and Public Workspace A workspace can be represented as a panel or a window on the users screen. The public workspace, the only space for the share contexts presentation found in the strict WYSIWIS, exists in this relaxed form for the requirement of group cooperation. In addition to the public shared workspace, each participant can have a private workspace, allowing the operating user to perform private activities or tasks invisible to other users. Typical operations performed in the private workspace are: taking notes; preparing ideas; preparing document; drawing images etc, to be later proposed or contributed to the public workspace.

Transition of objects between private and public workspaces is needed. The seamless transition between both workspaces, such as the mouse drag-and-drop action, improves the rate of group cooperation.

A good example of groupware applying this modification is the Hassotobi. The proposed groupware system did not apply this modification since it does not match the KJ-Methods philosophy.

Cursor Display The cursor positions of all participants are transmitted and shown on every participants screen in the strict form of the WYSIWIS. Although a participants cursor can be personally customized by using different colors, patterns, or sizes, as the number of participants grow, the number of cursors will increase causing confusion to users during the group session.

For avoiding confusion, the relaxed WYSIWIS shows only the private use of respective users. By the way, choosing to retrieve the cursor positions of other users, or to broadcast a cursors position to other users, is allowed. Broadcasting a cursor is a great tool for specific locations in a workspace or in a group document during group communication.

The proposed groupware applied this modification. Only private cursors of respective users are shown. Even though broadcasting a cursor is not implemented in our system, telepointing is used. This is the system for broadcasting the reference point or position on the workspace to all members of the team. It can be used for specifying the location currently under consideration or of interest by some members in the group. Telepointing is used instead of cursor display, and is described in Section 4.3.3.

Personal Screen Layout The window-war can be avoided by giving each participant a private screen layout. Each participant can fully customizes his personal screen layout and screen property by style, position, and size.

The proposed groupware applies this modification. Since the proposed groupware is a Web application run in a Web browser, customizing the style, position, and size of the Web browsers window, it is usually allowed in most operating systems. Thus, the personal screen layout capability of the proposed groupware is automatically derived from the Web browser software.

Different Viewport of Displayed Information The scroll-war can be avoided by giving each participant a different viewport of displayed information (the share context, or group document). Each user has the right to fully control his viewport to show the desired portion of the share context. The operations for controlling his own viewport, such as zooming, moving, adding a viewport (showing two different position of the same share context), and deleting an added viewport can be implemented. The proposed groupware applies this modification.

Different Style of Displayed Information The style-war can be prevented by allowing each participant to customize his or her desired style of representation of the shared information. The representation of the group document can be customized to a users requirements. This typically includes appearance properties, including the size, color, text style, etc. of objects in a group document, and the way of representing the shared data, for example, choosing between a table or chart.

The following scenarios are useful in this modification. For the older users, allowing an increase in the size of text written on each label in the KJ-Chart is very useful. The background of the KJ-Chart can be filled by a dark color to give more contrast between the written text and the background. Customizing the private style of displayed information to satisfy the needs of the user dramatically improves the accessibility of the share information, and directly promotes the groups creativity performance. The proposed groupware has not yet applied this modification.

Time Divergence Time divergence allows a delay in the propagation of events from the action of a participant being transmitted to all other participants. The share context is synchronized after a certain time delay. Participants see a different version of the share context (or a group document) on their screen. The delay periods can be controlled in two ways:

Explicitly Controlled by User When a user modifies the shared object in this mode, the modification is privately visible. Once the modification is done, the user can command propagation of the change for public visibility, by invoking a command or pressing a button.

Implicitly Controlled by Application Applications can be commanded to propagate alterations from the privately visible to the publicly visible by a set of rules. A typical rule might be a delayed timer. If the modification occurred in the last five minutes (as an example), this change is propagated for public visibility. The more sophisticated rules can also be applied to more intelligent and more convenient groupware. Whether to use the expert system or artificial intelligence module can be decided when the modification is propagated for public visibility.

In our groupware, this relaxed modification is not applied. In our opinion, showing different versions of the KJ-Chart on the screens of team members causes confusion in the group KJ-Method. This modification is suitable for the longer private tasks, such as in collaborative drawing software, which allows participants to show their modification when it is ready.

Coupling of User Interfaces Coupling of user interface and shared information can be classified into three levels as follows:

1. **Strict Coupling Level:** This is a tight coupling of user interfaces and shared information. Each user sees the same shared object using an identical view.

2. **Relaxed Coupling Level:** This is a relaxed coupling. Each user can see the same object using different views. It can be a different representation or a different part of that object.

3. **Loosely Coupling Level:** Users have access to the shared information but it is manipulated on different objects. A portion of users might have full control over a shared object. Another portion of users can only view the shared object. Other users cannot access the shared object at all. But all users have a shared space or media for collaboration.

The coupling of user interface of the proposed groupware is the relaxed coupling level. All members see the same KJ-Chart, but each member can control which portion of the chart is shown to him.

4.5 Qualitative Evaluations of the Gugeek Design

To measure performance and quality of Gugeek groupware design, we opened Gugeek for testing by a group of invited testers. The testers consisted of Master and Doctoral students of the school of knowledge science and the school of information science, Japan Advanced Institute of Science and Technology (JAIST). All testers must have an English test score equivalent to TOEFL of 550 or more. All functions of the system were introduced to all the testers by oral presentation and demonstration.

They were told to use the system to create at least three KJ-Charts with two other testers. When the testers finished the evaluation, the questionnaire forms as shown in Figure 4.14 were distributed. These questionnaires ask students about difficulties found during the testing, desired features, and overall satisfaction of the system. The forms were filled in and the students left.

According to the questionnaires, the average of the overall satisfaction scores of the system, which are 1-5 Likert scales (1=*Very Poor* to 5=*Very Good*)), collected from 90 samples is 3.833 $\pm$ 0.838. The following is a list of difficulties followed by their frequency found in the survey responses ranked in descending order based on their frequency. The low frequency opinions are not shown.

- Lacking of event notification when changes occur. (9)
- Grouping operation is hard. (5)
- Erasing a group, the entirety of its children labels are removed and cannot be redone. (4)
- Bugs found in the system. (4)
- Overlapping between ideas so unable to see other ideas underneath. (3)
- Size of a group is not automatically adjusted when it is changed. (3)
- Text Chat is not convenient to operate. (3)
- GUI is not good enough, lacking in multiple label operations. (3)
- Topic is not interesting. (1)
- During the convergent thinking phase, labels should not be editable. (1)

KJ Groupware system

Did you encounter any difficulties of using this system?
Comment / Suggestion / Opinion

Describe any new features that you think this system should have?
Any improvement?

Overall satisfaction of this system *

	1	2	3	4	5	
Very poor	○	○	○	○	○	Very good

Submit

Powered by Google Docs

Report Abuse - Terms of Service - Additional Terms

Figure 4.14: The questionnaire for evaluating the quality of Gugeek groupware.

The following is the list of desired features followed by their frequency found in the survey responses ranked in descending order based on their frequency. The low frequency opinions are not shown.

- Automatic Idea Arrangement System. (5)
- Alert Notification (Operations and Events). (4)
- Adding Picture. (3)
- Dislike Button. (3)
- Bugs Fix. (2)
- Non-overlapped Idea Arrangement. (2)

- Face-to-Face Video Chat. (2)
- Group Voting System for Major Changes. (2)
- Multi-Labels Selection and Group Operation. (2)
- Automatic Label Resize. (1)

4.6 Conclusion

This chapter explains in detail descriptions, histories, components, architecture, Web user-interfaces, workflows, and designs of the proposed KJ-Method groupware, Gugeek. The prototype divergent thinking support system, Eureka! is also explained. The walkthrough of all user interface operations is narrated. All components of the groupware is listed and explained. The comparison between Gugeek and the previously proposed KJ-Method groupware developed by other researchers is described. Finally, the key issues of Gugeek designs, including the architecture and user interface designs and their decisive reasons are given.

This chapter explains all perspectives about the framework-paradigm support level of the Gugeek groupware, which is an important basis for the rest of this dissertation. The creativity assistant programs, which are high performance artificial intelligence techniques for supporting both the divergent and convergent thinking phases of Gugeek and mandatory requirements for the generative support level, are described in the next chapter of this dissertation.

Part II

Creativity Assistants in KJ Method Groupware

Chapter 5

Divergent Thinking Support Modules

5.1 Introduction

Problem solving by brainstorming proposed by Osborn [18] in 1948 is a popular framework and technique for enhancing creative thinking ability and to this day still has a high impact on research works in the psychology and cognitive sciences. The brainstorming technique consists of two major thinking phases:

1. Divergent thinking
2. Convergent thinking

In the first phase of solving any problems by the brainstorming technique, the divergent thinking phase is performed to focus on as many possible alternative solutions to such problems as possible. The divergent thinking phase is the process intended to discover all creative possibilities. It is well defined with the following rules [15].

1. Produce a large quantity of ideas without any criticism.
2. Unusual ideas are highly welcome.
3. Adapting or modifying previously suggested ideas is encouraged.

The convergent thinking phase is then performed by filtering, combining, and fusing the set of ideas derived from the divergent thinking phase and turning them into practical concrete ideas, which can actually be used for solving the problem. A divergent thinking support system helps the divergent thinking process by stimulating the construction of ideas by a process of idea association. Idea association [56] is a thought process that leads to a new idea from an original idea associated by certain principles. A divergent thinking support system can be classified into three categories by the degree of promotion of the idea association process [57, 29].

1. Free association.
2. Forced association.
3. Analogy conception.

The first category is random association, presenting random information or keyword to users to promote idea association. The difficulty in developing random association support is quite simple.

The second category is forced association, presenting related information to the topic problem to users to promote idea association. Even though the difficulty in developing this category is harder than in the first category, the forced association methodologies have been empirical evaluated several times in the past by much research confirming that the performance of forced association is efficient.

The last category is analogy conception, generating a metaphor of the topic problem to users to promote idea association. The difficulty in developing such support is extremely hard.

This chapter compares four methodologies of forced association in divergent thinking by experiments on Wikipedia. The first two adopt the *association search*, which dynamically finds associative relationships among documents by statistical computation. The latter two adopt the *informative entity extraction*, which finds maximal matching between an input sentence and entities of Wikipedia.

These four forced association modules require the following three external components:

1. The Eureka! divergent thinking support system was used as the experimental platform. It is a mediator for transferring users inputted information to four forced association modules, and for displaying the associative information yielded from these modules to users to promote their idea association. The detailed information of the Eureka! divergent thinking support system was previously described in Section 4.2.

2. Association search is used in the first two forced association engines, and implemented by the association search engine GETA [64].

3. The English Wikipedia articles database is a source containing an enormous amount of present-day knowledge. This knowledge-base is a raw resource for extracting related information about the problem topic, and is later displayed to users to promote idea association.

The quality of four methodologies is compared to experiments on Wikipedia in both quantitative and qualitative evaluations. Section 5.2 explains the association search and the informative entity extraction. Section 5.3 briefly reviews an association search engine named "GETA" [64]. Section 5.4 describes the divergent thinking support module and shows the four divergent thinking support engines. Section 5.5 shows the experimental settings. Section 5.6 gives experimental results and our observation, and Section 5.7 concludes this chapter.

5.2 Structured Documents

For divergent thinking support systems, we focus on multi-layered structured documents (e.g., Figure 5.1) as a knowledge-base. Each node of a structure is a non-empty sequence of tokens. Typically, a token is a word, and a node is a category, a title, or a document. A child node may contain hyperlinks that point to a node in a higher layer. For instance, Figure 5.1 describes the structure of Wikipedia such that "Category" is a parent of "Title", "Title" is a parent of "Content", in which a "Freelink" [1] points to a "Title".

[1] `http://en.wikipedia.org/wiki/Wikipedia:Free_links`

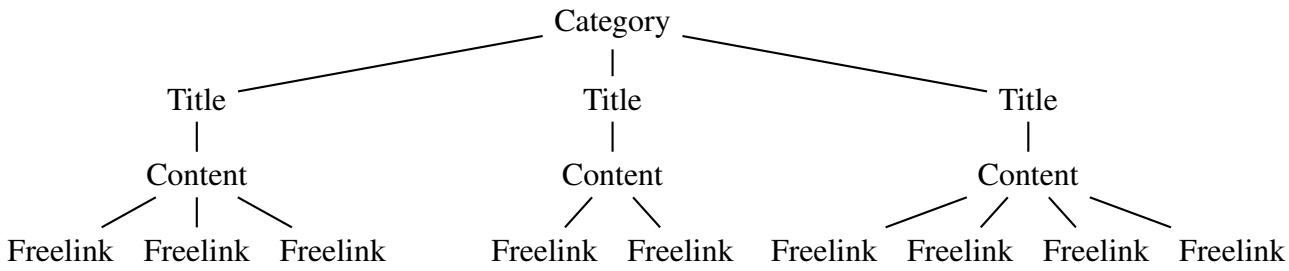
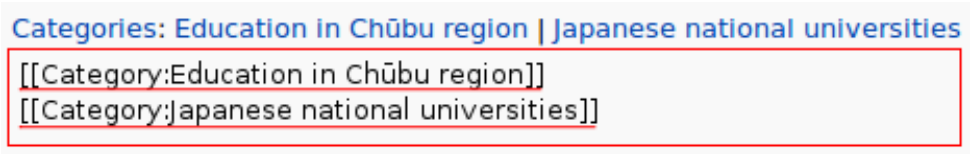


Figure 5.1: An example structure of target knowledge-bases.

(a) Freelinks in the JAIST Wikipedia page.



(b) Categories labeled at the bottom of JAIST Wikipedia page.

Figure 5.2: The associative information in a Wikipedia article and its source code

Our instance as a structured document is Wikipedia (English version), one of the largest collective intelligence knowledge bases, and contains a huge collection of encyclopedia articles. Each article in Wikipedia consists of a title, content, freelinks (see Figure 5.2a), and category labels (see Figure 5.2b).

5.2.1 Informative Entity Extraction

Informative entity extraction is a function for extracting an informative entity found in a query to retrieve structured documents relating to that query from the knowledge-base. Informative entity extraction assumes a structured document, in which a token is a word. A query is a phrase (i.e., a sequence of tokens), and the searched result is the set of titles (i.e., sequences of tokens) that contain the phrase as its subsequence. We call such titles *informative entities*.

Definition 1. *Let $\mathcal{W}$ be a set of words, and let $\mathcal{T}(\subseteq \mathcal{W}^*)$ be a set of titles (sequences of words). For an input word sequence $\psi \in \mathcal{W}^*$, we define*

$$Subseq(\psi) = \{\psi' \neq \epsilon \mid \exists \psi_1, \psi_2 \,.\, \psi_1 \psi' \psi_2 = \psi\}$$
$$\mathcal{A}(\psi) = Subseq(\psi) \cap \mathcal{T}$$

$Subseq(\psi)$ is the set of non-empty subsequences of ψ (which is sorted with decreasing order in respect to the length) and $\mathcal{A}(\psi)$ is the set of informative entities. For example, from "Reduce electricity usage in Japan Advanced Institute of Science and Technology", "Japan

Advanced Institute of Science and Technology" is extracted, which is at item [2] of Wikipedia. Since $Subseq(\psi)$ is sorted, the extraction reports the longest informative entity first, from which we can expect a more specific matching.

5.3 Association Search

In an association search, an article is regarded as a multiset [65] of tokens (typically, an article is a document and a token is a word). A query is a (multi)set of tokens, and the searched result is a ranking among tokens with respect to a given similarity measure.

Let ID_1 be a set of articles, and let ID_2 be a set of tokens.

Definition 2. *An* association system *is a quadruplet* $\mathcal{A} = (ID_1, ID_2, a, \mathrm{SIM})$ *where* a *is an association function and* SIM *is a similarity function such that*

$$\begin{cases} a: & ID_1 \times ID_2 \to \mathbf{N} \\ \mathrm{SIM}: & ID_2 \times MP(ID_1) \to \mathbf{R}_{\geq 0} \end{cases}$$

where $\mathbf{N}$ *is the set of natural numbers,* $\mathbf{R}_{\geq 0}$ *is the set of non-negative real numbers, and* $MP(X)$ *is the multiset consisting of non-empty subsets of* X*. We say that* $\mathcal{A}^t = (ID_2, ID_1, a^t, \mathrm{SIM}^t)$ *is the transpose of* $\mathcal{A}$ *where* $a^t(y, x) = a(x, y)$ *and a given* $\mathrm{SIM}^t : ID_1 \times MP(ID_2) \to \mathbf{R}_{\geq 0}$*.*

For $X \subseteq ID_1$ *(resp.* ID_2*) and* $n \in \mathbf{N}$*, let* $A(X, n)$ *(resp.* $A^t(X, n)$*) be the function collecting the top* n*-elements in* ID_2 *(resp.* ID_1*) with respect to the similarity* $\mathrm{SIM}(y, X)$ *(resp.* $\mathrm{SIM}^t(y, X)$*) for* $y \in ID_2$*. An* association search *is*

$$A^t(\{y \mid (y, v) \in A(X, m)\}, n)$$

for $m \in \mathbf{N}$*.*

In the definition of an association search, the number m is not specified. From the empirical study, GETA (see Section 5.3.1) sets m to 200 by its developers under the balance of efficiency and precision. Note that during an association search, we first compute $A(X, m)$. Its result is regarded as a *summary* that characterizes X.

Typical examples of association searches are:

- ID_1 is the set of documents, ID_2 is the set of words, and $a(d, w)$ is the number of occurrences of a word w in a document d. In this case, an association search is document-to-document.
- ID_1 is the set of words, ID_2 is the set of documents, and $a^t(w, d) = a(d, w)$. In this case, an association search is word-to-word.

From now on, we fix ID_1 as the set of documents and ID_2 as the set of words. To search a document d with a query q, $d \in ID_1$ and $q \in MP(ID_2)$. Usually $q \notin ID_2$ (i.e., there is no precise matching). In this situation, an associative search is performed as follows:

Step 1. A summary (characteristic keywords) of q is produced. This is performed by taking top-m words among $w \in q$ by $\mathrm{SIM}^t(w, ID_1)$. $ID_1 \in MP(ID_1)$ is regarded as the multiset consisting of all documents appearing precisely once.

[2] http://en.wikipedia.org/wiki/Japan_Advanced_Institute_of_Science_and_Technology

Step 2. For the set $q' \subseteq MP(ID_2)$ of top-m words in q, top-n documents in ID_1 are selected by evaluating $\mathrm{SIM}(d, q')$ for each $d \in ID_1$.

GETAssoc permits a similarity function SIM of the form

$$\mathrm{SIM}(d, q) = \sum_{t \in q} \frac{\mathrm{wq}(t, q) \cdot \mathrm{wd}(t, d)}{\mathrm{norm}(d, q)} \tag{5.1}$$

with the assumptions that $\mathrm{wd}(t, d) = 0$ if $t \notin d$ and $\mathrm{wq}(t, q) = 0$ if $t \notin q$. Typically:

- The value of $\mathrm{norm}(d, q)$ is dependent only on d. (In such cases, SIM^t is obtained by simply swapping wq and wd.)
- Both wq and wd are defined dependent on the association function a.

For an efficient association search implementation (e.g., GETA), we assume $\mathrm{SIM}(y, X) = 0$ if $a(x, y) = 0$ for each $x \in X \subseteq ID_1$ and $y \in ID_2$.

Note that an association search does not require structured documents. The key observation is a dual relationship between words and documents. ID_1 and ID_2 are swapped by regarding a document as a multiset of words and a word as a multiset of a document. The association search ignores the ordering of words; it does not distinguish between "Weather is not always fine" and "Weather is always not fine".

5.3.1 GETA - Generic Engine for Transposable Association Computation

The Generic Engine for Transposable Association Computation [3] (GETA) is an association search engine developed at NII [64]. A key feature of GETA is its scalability; it quickly handles a dynamic association search on more than ten million documents, such as Webcat Plus [4], Imagine [5], and Cultural Heritage Online [6].

Similarity functions

GETA accepts user-defined similarity functions of form in Eq. 5.1 as well as default similarity functions,

- Smart measure [66],
- Okapi BM25 [67],
- Cosine [68],
- Dot Product [68].

These similarity functions are defined as functions of type $ID_2 \times MP(ID_1) \rightarrow \mathbf{R}_{\geq 0}$ as in Figure 5.3, where $d \in ID_1$, $q \in MP(ID_1)$, $t, w \in ID_2$, and

- $w_{q,t} = log(\frac{N}{f_t+1})$

[3] http://getassoc.cs.nii.ac.jp
[4] http://webcatplus.nii.ac.jp
[5] http://imagine.bookmap.info
[6] http://bunka.nii.ac.jp

- $w_{d,t} = log(f_{d,t} + 1)$
- f_t is the number of documents that contain t.
- f_d is the number of words in document d.
- $f_{x,t}$ is the number of occurrences of the word t in x.
- N is the number of documents in ID_1.

GETA sets $\theta = 0.2, k = 0.2$, and $b = 0.75$ for the default Smart Measure in Figure 5.3.

These default similarity functions are directly applied to **Step 2** of an associative search. For **Step 1**, the dual similarity functions of type $ID_1 \times MP(ID_2) \rightarrow \mathbf{R}_{\geq 0}$ are needed, which are obtained by transposing the relation as $d \in w$ (a document d contains a word w) when $w \in d$ (a word w appears in a document d). In practice, there are small differences, e.g., setting of stop words, but in principle, GETAssoc uses the dual functions on both document-to-document and word-to-word similarly.

Structure The key data structure of GETA is a WAM (Word Article Matrix), which represents an association function in Definition 2. WAM is usually a huge sparse matrix of which rows are indexed by names of documents and columns are indexed by words. When ID_1 is a set of words and ID_2 is a set of documents, the cross point of the row of a word w and the column of a document d is $a(w, d)$, which is the number of occurrences of a word w in a document d. Then, the transpose $a^t(w, d)$ is obtained as a transposed WAM. In GETA implementation, a huge and sparse WAM is compressed either vertically or horizontally. These two compressed matrices enable us to compute association functions a and a^t, respectively.

Example The following is an example in the GETA calculation. Before we proceed to the example, the following configurations are used:

- Similarity function = Dot Product[68].
- The number of top-n most related documents, $N = 2$.
- The number of top-m most related keywords, $M = 2$.

First, the WAM is constructed by indexing documents from the knowledge-base. The sample WAM constructed from indexing three documents which contain 12 words of four vocabularies is shown in Figure 5.4a

"twitter dollar", which is a query received from users, is submitted to the GETA. The query is converted as a frequency vector as shown in Figure 5.4b. The dot products between the query vector and each row of WAM are computed. For example, the dot product between the query vector and the IT new row is $[1, 0, 1, 0] \cdot [2, 0, 1, 4] = (1 \times 2) + (0 \times 0) + (1 \times 1) + (0 \times 4) = 2+0+1+0 = 3$. Thus the first row of the associative documents vector as shown in Figure 5.4c is 3. Figure 5.4c shows the set of most related documents of the query.

Since $N = 2$, the similarity scores of the top-2 most related documents are collected. The similarity scores of lower ranks are set to zero. Note that the number between the pair of parentheses in Figure 5.4c shows the similarity score before setting it to zero.

After that, the dot products between a vector of associative documents and each column of WAM are computed. For example, the dot product between the associative documents vector

Smart Measure [66] :

$$\frac{1}{avg(f_d) + \theta(f_d - avg(f_d))} \sum_{t \in q \cap t \in d} log(\frac{N}{f_t}) \cdot \frac{1 + log(f_{d,t})}{1 + log(avg_{\omega \in d}(f_{d,\omega}))} \cdot \frac{1 + log(f_{q,t})}{1 + log(avg_{\omega \in q}(f_{q,\omega}))} \tag{5.2}$$

Okapi BM25 [67] :

$$\sum_{t \in q \cap t \in d} log(\frac{N - f_t + 0.5}{f_t + 0.5}) \cdot \frac{f_{d,t} \cdot (k+1)}{f_{d,t} + k \cdot (1 - b + b \cdot \frac{f_d}{avg(f_d)})} \tag{5.3}$$

Cosine [68] :

$$\frac{\sum_{t \in q \cap t \in d}(w_{q,t} \cdot w_{d,t})}{\sqrt{\sum_{t \in q}(w_{q,t}^2) \cdot \sum_{d \in q}(w_{d,t}^2)}} \tag{5.4}$$

Dot product [68] :

$$\sum_{t \in q \cap t \in d} (w_{q,t} \cdot w_{d,t}) \tag{5.5}$$

Figure 5.3: Four similarity measures and their formulas

	“twitter”	“tennis”	“dollar”	“facebook”
IT news	2	0	1	4
Economic news	0	0	2	0
Sport news	0	2	1	0

(a) A sample WAM

	“twitter”	“tennis”	“dollar”	“facebook”
Query	1	0	1	0

(b) A sample query vector

Documents	Similarity Score
IT news	3
Economic news	2
Sport news	0(1)

(c) Associative documents vectors of the query

Keywords	Similarity Score
“facebook”	12
“dollar”	8
“twitter”	0(6)
“tennis”	0(2)

(d) Summary of the query

Figure 5.4: An example of WAM and sample results.

and the dollar column is $[3, 2, 1] \cdot [1, 2, 1] = (3 \times 1) + (2 \times 2) + (1 \times 1) = 3 + 4 + 1 = 8$. Thus the second row of the summary vector as shown in Figure5.4d is 8. Figure 5.4d shows the set of the most related keywords of the query. Since $M = 2$, "facebook" and "dollar" are used as the summary of the query "twitter dollar". It is the output of the association search.

According to this example, associative keywords, which are significant keywords found in related knowledge domains (the associative documents that are significant to the query), such as "facebook" and "dollar", which might not be found in the KJ-Chart, can be recommended to users for triggering new ideas.

5.4 Divergent Thinking Support Modules

5.4.1 Knowledge-base

We dumped the whole English Wikipedia Website [7] and deployed it into our local knowledge-base, which is used in all experiments.

To expect improvement of retrieval accuracy in GETA, the set of text preprocessing, which is a process for removing noises, appears in the text, and is performed as follows:

1. **Filtering:** filters some punctuation, invisible computers special characters, non-English text, and function words.

2. **Stemming:** reduces the inflected words to their root form (for example, "fishing" and "fisher" to just "fish"), is applied to all Wikipedia articles. We use the Snowball [8] English stemmer.

To be searchable by GETA, a list of words and their frequency pairs appearing for each article called ITB format [69] needs to be constructed. After each article is preprocessed, all sentences in an article are segmented into the list of words by a regular expression program. The occurrences of words are then counted, and only the set of distinct words is kept. At this point, the list of words and their frequency pairs are constructed.

These lists are later read by GETA for constructing WAM. The association search can be performed.

5.4.2 Four Divergent Thinking Support Engines

The following are the association engines used during experiments. All are used under the Eureka! interface (see Chapter 4.2) and only visible differences are produced on the suggestion lists.

GD: GETA's Most Related Documents

A users input sentence is a query for retrieving associated documents by GETA. The most sophisticated Smart Similar Measure [66] is used for this engine. The titles of the search result are suggestions, which are sorted by their similarity scores.

[7] `http://download.wikimedia.org/enwiki/20100730/`
[8] `http://snowball.tartarus.org/`

GK: GETA's Most Related Keywords

A user input sentence is a query for retrieving the summary of the query by GETA. The Smart Similar Measure [66] is used for this engine. The resulting summary (associated keywords) of suggestions is sorted by their similarity scores.

WF: Wikipedia's Freelinks

A user input sentence is a query for extracting informative entities. The freelinks in the contents are suggestions, which are sorted by the length of entities. For instance, a freelink with a longer parent entity comes first. In our current implementation, if the parent entities are the same length, a freelink with a parent entity that appears earlier in an input sentence comes first.

WC: Wikipedia's Categories

A user input sentence is a query for extracting informative entities. All titles in the same category as extracted entities are suggestions, which are sorted by the length of entities. This means that a category with a longer child entity comes first. In our current implementation, if entities are the same length, the order among categories and titles in categories obeys that of Wikipedia.

NE: No engines

No associative information is supplied (no suggestion lists).

5.5 Experimental Setting

5.5.1 Evaluation Methods

The quality of an engine is measured by the degree of creativity that users generate. As proposed by Guilford [70], the degree of creativity can be measured by fluency, flexibility, and originality of ideas. Neupane, et al. proposed the following four measures [71].

- **Number of ideas:** The total number of input ideas.
- **Fluency of ideas:** The total number of input ideas excluding those judged to be off-topic, redundant, errors, impossible, and/or useless.
- **Flexibility of ideas:** The total number of viewpoints found in input ideas. Viewpoints are defined before experiments corresponding to each topic.
- **Originality of ideas:** This is the total number of distinct ideas. When ideas are very similar or identical, they are grouped together.

  For example, consider a problem "Apart from avoiding rain, write down any other uses of an umbrella". Table 5.1 shows the ideas generated by a participant. Table 5.2 shows that there are four viewpoints. The third and fifth ideas in Table 5.1 are grouped.

Three human evaluators, who did not participate in the experiments, judged the fluency of ideas, the flexibility of ideas, and the originality of ideas. A majority vote is taken if a conflict occurs.

Table 5.1: A log of ideas during the experiment.

Idea No.	Generated Idea
1	Avoiding the sun
2	Collect rain water (upside down)
3	Use as a ruler
4	To lock a door by jamming it between the two handles
5	Use as a measuring stick
6	Use the handle to grab something
7	Use as a basket (upside down)
8	A cane for elderly people
9	Dry socks by putting it upside down and hang socks on the frame
10	Disassemble the umbrella and use the frame as sticks for cooking

Table 5.2: A list of viewpoints during the experiment.

Idea Viewpoint	Idea No.
Furniture	9
Tool	1,3,4,5,6,8
Recycle	10
Accessories	
Interior	
Plaything	
Container	2,7
Using Materials	
Clothes	
Social	

Three human evaluators received a printed version of a KJ-Chart produced by a participant via the Eureka! First, they were asked to circle labels that were considered off-topic, redundant, errors, impossible, and/or useless. If a label was circled more than twice it was removed from the KJ-Chart. The number of remaining labels is the fluency of ideas.

Three human evaluators received the list of viewpoints as shown in Table 5.1. For each remaining label, they were asked to circle a viewpoint that most matched with the label. A label that belonged to a viewpoint is circled twice or more. In case three human evaluators had three different opinions, they would discuss and vote for the most matched one.

Finally, three human evaluators find a group of labels that seem to be very close or identical to others. Once a human evaluator found such a group, he/she raises it in the discussions. Three human evaluators vote to gain consensus.

5.5.2 Procedure of Experiment

The experiments are conducted by five groups of users, and each group consists of two users (ten in total). Participants ranged from bachelor to doctoral students of the School of Knowledge Science, and the School of Information Science of the Japan Advanced Institute of Science and Technology. Before the experiments, every participant is informed of the following:

1. System procedure and usage.
2. Divergent thinking, its rules and examples.
3. Q&A

The five following topics are assigned to all participants. In each topic, ten viewpoints (Table 5.3) are prepared in advance, and all resulting ideas are classified into the most related viewpoints, though there are few difficult cases.

1. If all human beings had a third hand on their back, write down the advantages of that hand.
2. Apart from avoiding rain, write down any other uses of an umbrella.
3. What steps should the authorities concerned take to increase the number of foreign tourists in Japan?
4. How could you contribute to power saving at your school?
5. Imagine that you are a product designer. Please design new products likely to be sold to teenagers.

Experiments use four engines and no engine (for comparison) as described in Section 5.4.2, and the maximum number of suggestions is limited to 30 each. The list of suggestions appears at the right side and the bottom bar of the Eureka! groupware screen as shown in Figure 5.5.

To avoid the effect of tool experiences, the topics and the engines are assigned to groups in a different order as in Table 5.4. The timeout for each topic is set to 15 minutes.

5.6 Experimental Result and Observation

5.6.1 Quantitative Result

The quality of a divergent thinking engine is estimated by four measures in Section 5.5.1. The means and standard deviations of the scores collected from ten sessions are shown in Table 5.5.

Figures 5.6, 5.7, 5.8, 5.9 show the descriptive statistics of the results yielded by four creativity evaluators in the proposed five divergent thinking engines. Note that these figures display a box-and-whisker diagram, which displays the sample minimum, lower quartiles (Q1), median (Q2), upper quartiles (Q3), and the sample maximum. The horizontal bars inside boxes are medians. The small circles are outliners.

According to Table 5.5, the number of ideas, the fluency of ideas and the originality of ideas are mostly discovered when supported by GK. Although WF yields the highest flexibility of ideas, it seems to be not as significantly different to others.

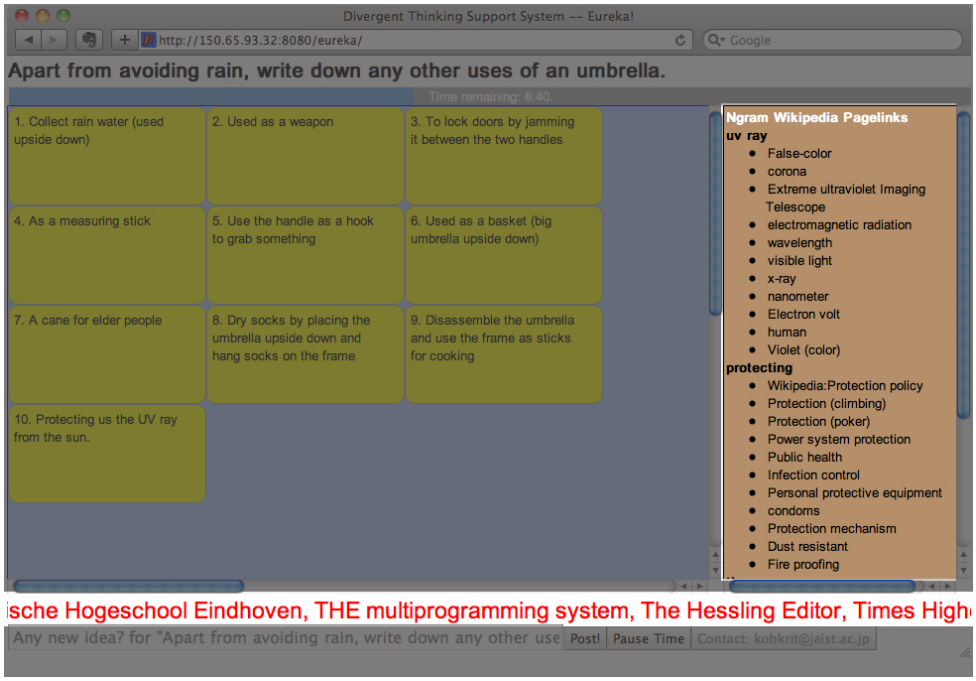


Figure 5.5: The location of the suggestions in the Eureka! screen

Table 5.3: A list of idea viewpoints of Topic 1-5.

Viewpoint No.	**Topic 1**	**Topic 2**	**Topic 3**	**Topic 4**	**Topic 5**
1	Furniture	Security	Information	Habit	Sport
2	Tool	Fame	Economy	Attitude	Electronic
3	Recycle	Production	Quality	Consumption	Entertainment
4	Accessories	Trick	Attitude	Power Sources	Appearance
5	Interior	Social	Regulation	Social Enforcement	Toy
6	Plaything	Education	Quantity	Management	Health
7	Container	Novel Ability	Support	Economy	Security
8	Using Materials	Economy	Security	Regulation	Transportation
9	Clothes	Appearance	Attention	Presentation	Education
10	Social	Health	Promotion	Promotion	Production

Table 5.4: Topic and engine assignment

Group	**Topic 1**	**Topic 2**	**Topic 3**	**Topic 4**	**Topic 5**
A	GD	GK	WF	WC	NE
B	GK	WF	WC	NE	GD
C	WF	WC	NE	GD	GK
D	WC	NE	GD	GK	WF
E	NE	GD	GK	WF	WC

Table 5.5: Means and Standard Deviations of Quantitative Evaluation Results (per user per time period)

Measurement	**GD**	**GK**	**WF**	**WC**	**NE**
Number of ideas	12.8 ± 5.3	$\mathbf{15.9 \pm 10.1}$	11.8 ± 3.6	14.8 ± 6.4	12.7 ± 4.6
Fluency of ideas	12.5 ± 5.3	$\mathbf{15.0 \pm 9.0}$	11.2 ± 3.6	12 ± 6.5	11.2 ± 4.6
Flexibility of ideas	4.7 ± 1.9	4.6 ± 1.3	$\mathbf{4.8 \pm 1.8}$	4.6 ± 1.8	3.9 ± 1.7
Originality of ideas	11.1 ± 4.7	$\mathbf{11.7 \pm 4.8}$	10.2 ± 3.5	11.1 ± 6.2	9.8 ± 4.7

To measure the significance of differences found in the results, the one-way analysis of variances (ANOVA) among five divergent thinking support engines in each creativity evaluator is performed. This is a statistical test of whether or not the means of each creativity evaluator of four divergent thinking support engines are all equal.

The differences found in all creativity evaluators of four divergent thinking support engines are not statistically significant at 0.05 significant levels. The F-ratio of the differences in the number of ideas, the fluency of ideas, the flexibility of ideas, and the originality of ideas are listed as follows:

- Number of ideas: $F(4, 45) = 0.695, p = 0.599, \eta^2 = 0.058$,n.s., $p > 0.05$.
- Fluency of ideas: $F(4, 45) = 0.656, p = 0.626, \eta^2 = 0.055$,n.s., $p > 0.05$.
- Flexibility of ideas: $F(4, 45) = 0.449, p = 0.773, \eta^2 = 0.038$,n.s., $p > 0.05$.
- Originality of ideas: $F(4, 45) = 0.248, p = 0.91, \eta^2 = 0.022$,n.s., $p > 0.05$.

The main reason for the insignificance of differences is too few number of samples, which is only 10 samples each (50 samples in total), and the samples have quite a broad standard deviation. However, from this data, we see that the GK yields produce higher creativity in divergent thinking than the others.

Our observation on the advantage of GK is firstly, that the GETA can accurately summarize keywords. Secondly, each item in the summary is just a word, which requires less time to understand, and can be loosely interpreted in various ways. Thus, users most fluently generate ideas when GK is used.

5.6.2 Qualitative Result

The qualitative evaluation is the survey answered by participants. The survey consists of four questions. The first three questions are multiple choice (Q1-Q3), and the last question is answered by individual comment (Q4). The first question is asked immediately after finishing each topic. All others are asked after finishing all five topics.

Q1. Please evaluate the usefulness of this divergent thinking support engine on a scale from 1(Poor) to 5(Excellent).

GD	**GK**	**WF**	**WC**	**NE**
3.0	**3.9**	2.6	2.7	1.0

Q2. After using all five divergent thinking support engines, which one do you consider is the MOST USEFUL to generate your ideas?

GD	**GK**	**WF**	**WC**	**NE**	**None**
2	**6**	0	0	1	1

Q3. After using all five divergent thinking support engines, which one do you think is the MOST USELESS to generate your ideas?

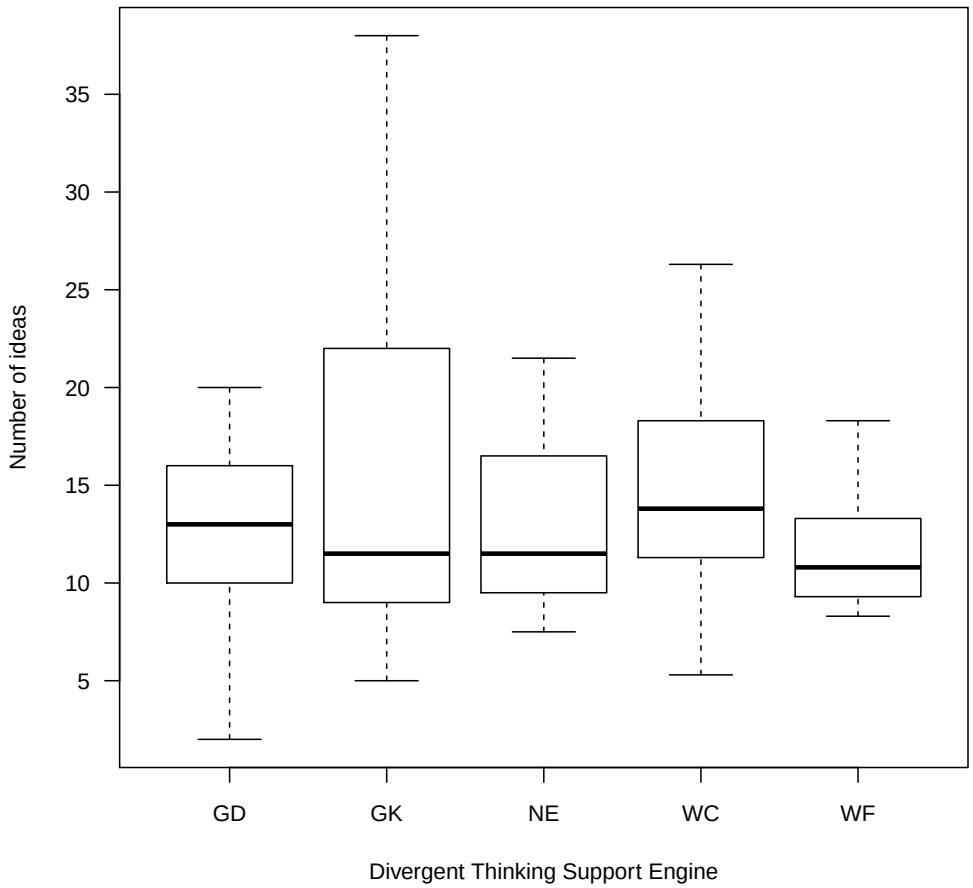


Figure 5.6: Quantitative result: number of ideas (per user per KJ-Chart)

GD	GK	WF	WC	NE	None
2	**0**	2	3	2	1

According to the survey, the GK obtains the most satisfaction from users, which is consistent with the quantitative result.

After analysis, we observed that most users avoid inputting the lengthy informative entities. Instead, they are replaced by sample words or abbreviations. For example, instead of inputting "Japan Advanced Institute of Science and Technology", they input "University" or just "JAIST". Thus, the informative entity extraction (see Section 5.2.1), which reports the longest informative entity first, fails. It is a major difficulty that decreases the quality of both WF and WC.

Q4. Was the content displayed to stimulate your ideas useful to you? If so, please describe how?

- Not useful.(2)
- Useful, when I have no/few ideas.(2)
- Useful, when seeking to trigger the next idea from the existing one.(4)
- Useful, when seeking new fresh ideas.(2)

Users found the suggestions most useful when they feel that the existing ideas could be more sophisticated or developed into new ideas.

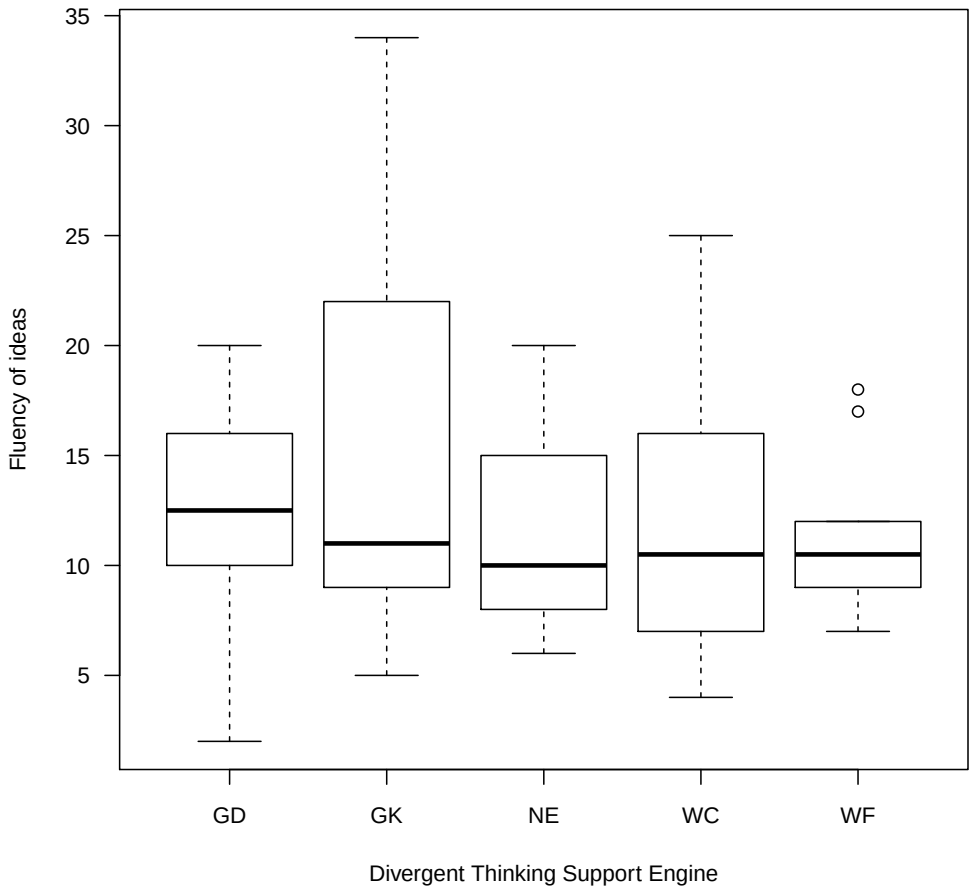


Figure 5.7: Quantitative result: fluency of ideas (per user per KJ-Chart)

5.7 Conclusion

This chapter compared the four divergent thinking engines based on forced association. The first two adopt the association search engine GETA, while the latter use more conventional informative entity extraction. The quality of the four engines is evaluated by experiments on Wikipedia using the Eureka! interface. GK (the most related keywords by GETA, as a summary of an input sentence) is the most effective and gives the highest satisfaction to users.

We think that a dynamic association creation on a large scale database (e.g., GETA) is useful for divergent thinking supports, whereas previous systems used either a statically pre-computed (relatively small) association, or manually prepared (huge number of) freelinks on the Internet. We believe that empirical statistical computation would find unexpected associations beyond biased human thinking.

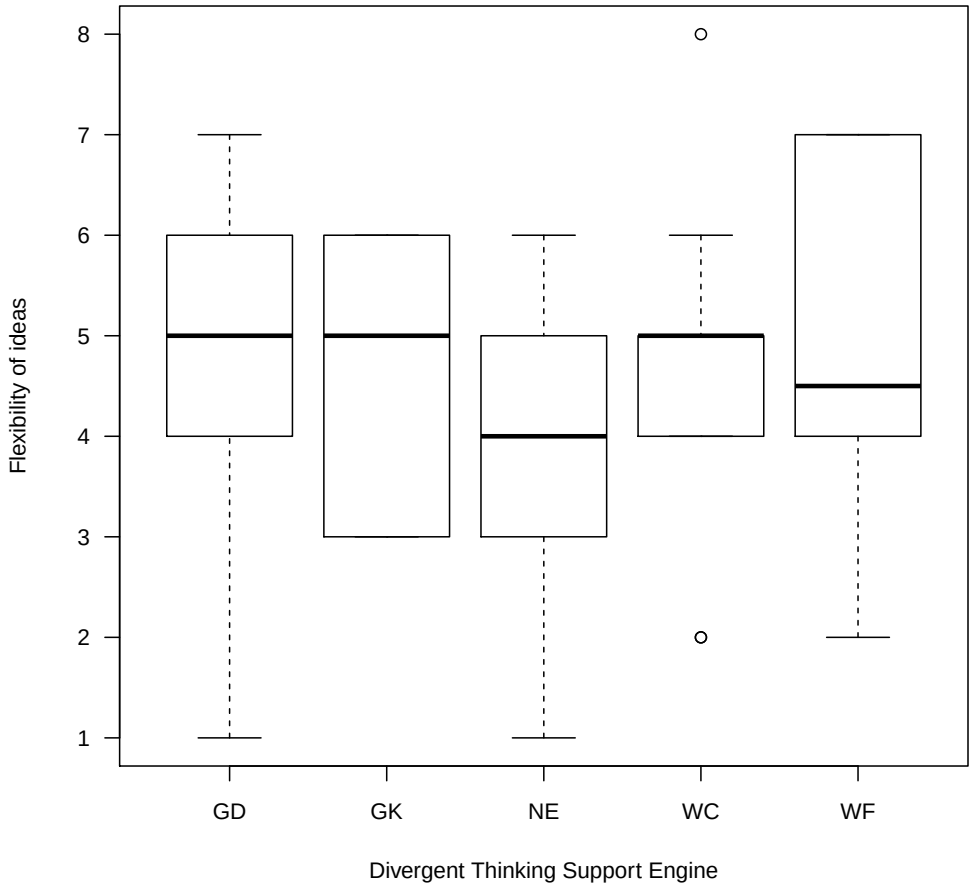


Figure 5.8: Quantitative result: flexibility of ideas (per user per KJ-Chart)

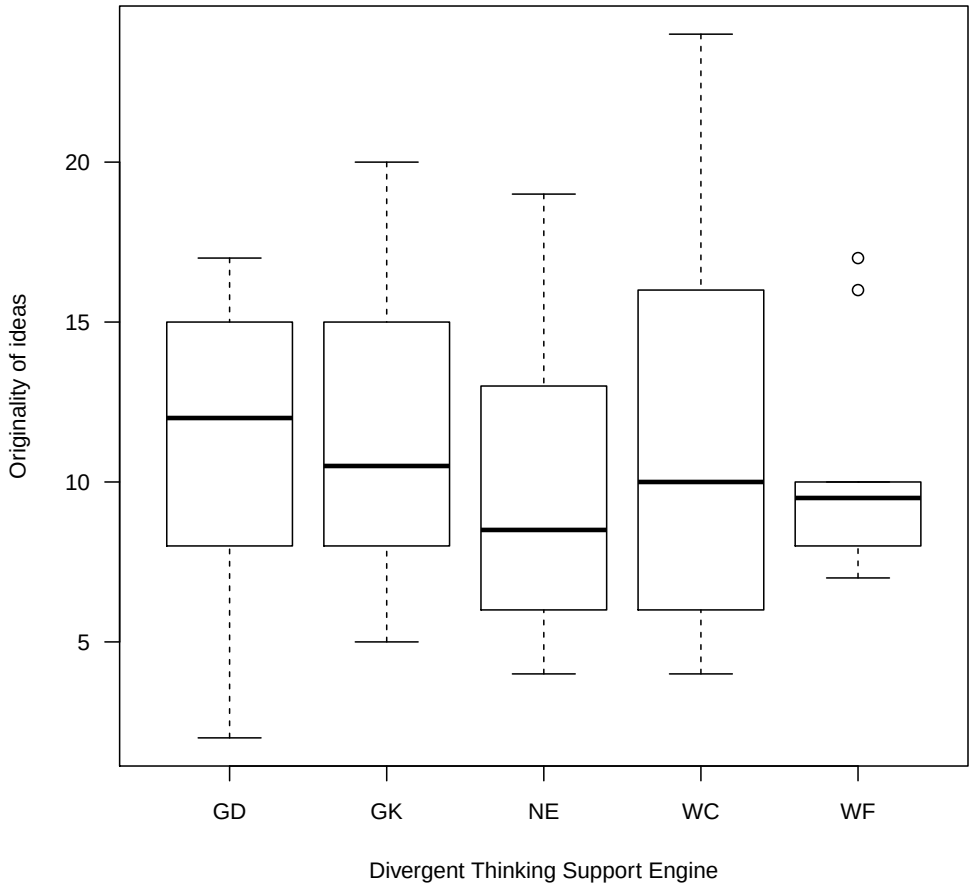


Figure 5.9: Quantitative result: originality of ideas (per user per KJ-Chart)

Chapter 6

Convergent Thinking Support Modules

6.1 Introduction

In the previous chapter, the divergent thinking phase, which is a first thinking phase in the KJ-Method, is supported by providing associative information aimed at promoting idea association performance to the user. Promoting idea association, a key source for generating a new idea, also promotes divergent thinking and accelerates the power of creative thinking while using the support system.

Convergent thinking, which is a phase of thinking right after divergent thinking in the KJ-Method, could also be promoted by the support system. There are several ways to support users during the convergent thinking phase. For example:

1. **Automatic Label Grouping:** The system automatically reads the content of all labels and groups them based on their mental association. With the current state of information technologies, it is easy to group information based on similarity, but not yet for their mental association. The similarity between two entities can be computed by consulting several knowledge-bases such as general ontology, large text corpus, and so on. For mental association, there is still a lack of knowledge base that contains rich mental association between two entities. Moreover, the system must truly understand the problem under consideration, its perspective, and its situation. Two sets of the same labels can be grouped differently if considering a different problem or even considering the same problem but from a different perspective, they can be group differently.

2. **Automatic Category Naming:** The system automatically predicts the title of categories when the label grouping step is finished. It reads the content of all labels in a group, extracting their common characteristics (a form of their mental association) or summarizing, and generating a text title for that group.

3. **Automatic Spatial Arrangement:** The system automatically chooses a suitable position for all labels and categories based on their similarity and relationship. It automatically helps to neatly arrange objects into a tidy order.

4. **Automatic Chart Making:** The system reads the spatial arrangement including the content of all labels and category titles and automatically creates relationships between them.

5. **Automatic Written Explanation:** The system reads the complete KJ Chart and automatically generates a written explanation.

Automatic category naming is computer support for idea processing at the generative support level, which is the highest computer support level for idea processing systems and firstly coined by L.F. Young in 1988 [5]. Current natural language processing technology facilitates automatic category naming in creativity support systems, which may be implemented in practice with acceptable accuracy. Automatic topic identification, a method for identifying the central ideas in a text (extracting the common characteristics between two or more texts), is a major component of automatic category naming and all others supports listed earlier for supporting convergent thinking. Automatic topic identification can be classified into three categories: statistical approaches, knowledge-based approaches, and hybrid approaches. The amount of text that should be produced by idea visualization programs is typically short. Therefore, external knowledge is needed to identify relevant topics. Thus, the proposed method is a knowledge-based approach, which uses two general domain knowledge bases, i.e., WordNet [72] and Yet Another General Ontology (YAGO) [73].

In this chapter, we propose a novel topic selection method for knowledge-based automatic topic identification, which is optimized to identify topics for idea groups in idea visualization programs. Our proposed method is based on a knowledge-based concept counting paradigm from a hierarchical concept taxonomy inspired by Lins work [74]. The performance of our proposed topic selection method was compared with three existing proposed methods on two idea visualization programs, i.e. the KJ Method and mind mapping programs. This chapter is organized as follows. Section 6.2 describes related studies. Section 6.3 explains the idea charts and knowledge bases. Section 6.4 describes how our method automatically identifies topics. Section 6.6 presents our empirical evaluations. Section 6.7 is the discussion and Section 6.8 concludes the chapter.

6.2 Related Works

Knowledge-based automatic topic identification based on concept counting paradigm from a hierarchical taxonomy has been proposed by several researchers.

Lin [74] proposed a concept counting technique that improved an earlier term frequency counting method. Lin also proposed a topic selection method, i.e., the branch ratio, which focused on the degree of generalization of concept, and provided an efficient way to extract multiple topics from a document. Lin used WordNet taxonomy, which contains very little knowledge about proper nouns compared with the YAGO ontology. Tiun et al. [75] proposed a topic selection method, i.e., the ratio balance, which ensures that the accumulated weight is higher than expected by using the Yahoo ontology as the knowledge base.

Automatic topic identification based on information retrieval techniques has been proposed by many authors. He et al. [76] proposed using an unsupervised clustering technique. Özmutlu et al. [77] proposed the use of a neural network classification technique. Clifton et al. [78] proposed the use of an association rule mining technique. Coursey [79] proposed the use of an improved Pagerank algorithm for topic identification with Wikipedia. Muñoz-García et al. [80] proposed the use of semantic relatedness and disambiguation techniques based on DBpedia.

Research studies based on artificial intelligence tools for idea visualization program have been proposed by several researchers. Leake et al. [81] and Kobkrit et al. [19] proposed knowledge-based methods that suggested relevant information to assist the production of idea charts. Vollalon [82] proposed an automatic concept map generation based on short essays.

6.3 Idea Chart and Knowledge Bases

6.3.1 Structure of Idea Chart

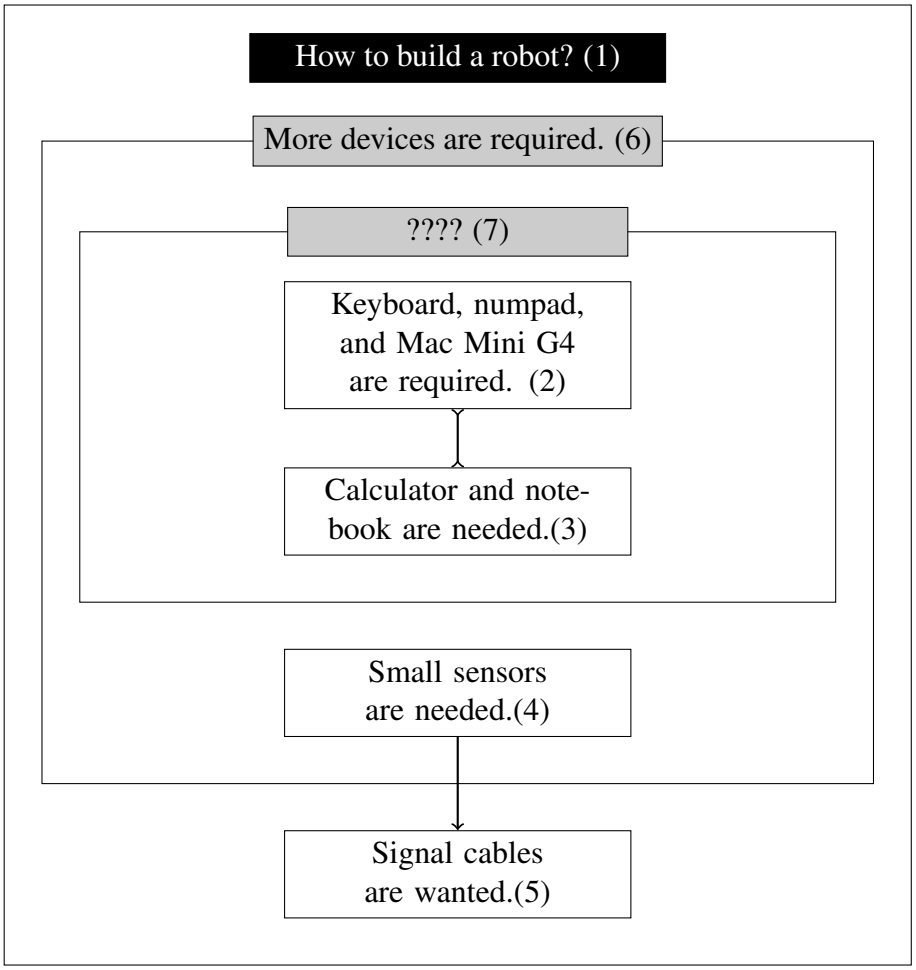


Figure 6.1: A sample idea chart (in the KJ Method program).

An idea chart is a representation of a set of ideas that is typically drawn after performing a creativity technique for an individual memorandum or mutual understanding in a team. An idea chart is defined as a rooted tree as follows.

Let $T = (V, E)$ be a tree. A vertex $v \in V$ is a non-empty word sequence $\psi \in W^*$, where W is a set of words that represents an idea. A v is an internal vertex if its degree is greater than one; otherwise, v is a leaf. Let L_T be the set of leaves of T. An edge $e \in E$ is a hierarchical generalization relationship; a vertex at a higher level is the generalization of a vertex at a lower level.

The tree T has exactly one root vertex that is denoted as R_T, which is the title of an idea chart. The set L_T is the set of idea labels. The subset of vertices $g \subseteq V$ of an idea chart T, which shares the same immediate parent vertex $h \in V - L_T$, is a member of an idea group G_h. The label h is a group header of the idea group G_h. The set of all group headers H where $h \in H$ has the following properties,

- $H = V - L_T$.

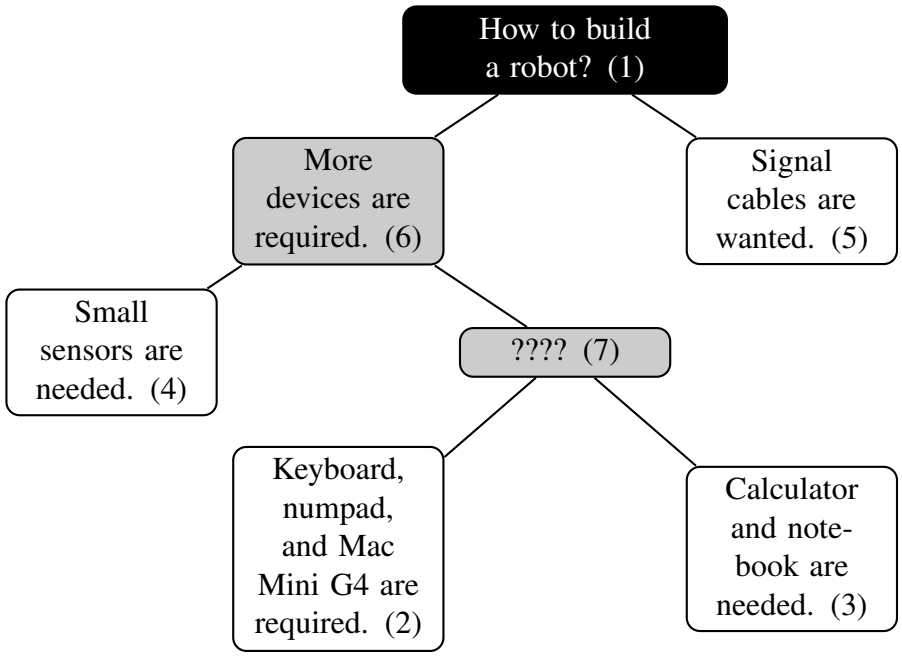


Figure 6.2: Tree representation of the sample idea chart (in the mind map form).

- $\cup_{h \in H}(G_h) = V - R_T$.

The structure of a sample idea chart and its tree representation is illustrated in Figure 6.1 and Figure 6.1 respectively. White, non-white, and black labels/vertices are L_T, H, and R_T, respectively, where $H = \{1, 6, 7\}$, $G_1 = \{6, 5\}$, $G_6 = \{4, 7\}$, and $G_7 = \{2, 3\}$; a number represents a label of the sample idea chart in Figure 6.1.

The research goal is to automatically identify an unlabeled group header h in an idea group G_h, such as the group header 7 (the content of which is "????") in Figure 6.1. Due the tree structure of an idea chart, the group header h should only describe the main idea of all members in G_h and it should not be more abstract than its parent node and siblings.

Definition 3. *Given a group header h in an idea chart T, the function $\mathrm{in}(h)$ returns a set of idea labels and their contents are summarized in h, which is defined as follows.*

$$\mathrm{in}(h) \quad = \quad G_h \tag{6.1}$$

Definition 4. *Given a group header h in an idea chart T, the function $\mathrm{out}(h)$ returns a set of idea labels and their contents should not be mentioned in h, and they are used to limit the abstractness of h, which is defined as follows:*

$$\mathrm{out}(h) \quad = \quad G_{\mathrm{parent}(h)} \cup \mathrm{parent}(h) - h \tag{6.2}$$

where $\mathrm{parent}(x)$ returns the parent of vertex x. For example, if we let h be the group header 7 in Figure 6.2: $\mathrm{in}(h) = \{2, 3\}$ and $\mathrm{out}(h) = \{4, 6\}$.

6.3.2 Knowledge Bases

WordNet is an English lexical database that groups the senses of words into sets of synonyms known as "synsets" and provides their semantic relationships, such as synonymy and hypernymy [72]. YAGO is a general-purposed semantic database with more than 10 million entities

(including common nouns, proper nouns, and name entities), which are derived from Wikipedia, WordNet and Geoname ontologies. YAGO also provides the semantic relationships among them [73]. We used WordNet and YAGO as the external knowledge bases for verbs and nouns, respectively.

WordNet and YAGO are organized as ontologies. Ontology consists of concepts that describe things and facts, which are the relationships between two concepts. A fact is represented as an ordered triplet that represents a domain, a range, and a relationship that associates a domain and a range. A triplet is represented as follows [73]:

$$(\texttt{Domain}, \textsc{relations}, \texttt{Range})$$

For example, to make a statement "An apple is a type of consumable fruit". Its triplet is defined as:

$$(\texttt{Apple}, \textsc{hypernym}, \texttt{ConsumableFruit})$$

Definition 5. *Let $\mathcal{C}$ be a finite set of concepts and $\mathcal{R}$ be a finite set of relations. A set of all possible facts $\mathcal{F}$ and an ontology $\mathcal{O}$ are defined as follow.*

$$\mathcal{F} = \mathcal{C} \times \mathcal{R} \times \mathcal{C} \tag{6.3}$$

$$\mathcal{O} \subseteq \mathcal{F} \tag{6.4}$$

Definition 6. *Given $D \subseteq \mathcal{C}$, $r \in \mathcal{R}$ and $\mathcal{O}$ is an ontology. A function $\mathrm{fact}(c, r, \mathcal{O})$ is defined as follows.*

$$\mathrm{fact}(D, r, \mathcal{O}) = \{(d, r, x) | x \in \mathcal{C} \wedge d \in D \wedge (d, r, x) \in \mathcal{O}\} \tag{6.5}$$

The function $\mathrm{fact}(c, r, \mathcal{O})$ returns the set of facts that has the relation r and the set of domain D that appears in ontology $\mathcal{O}$.

Definition 7. *Let $c \in \mathcal{C}$ and $r \in \mathcal{R}$. Given $F \subseteq \mathcal{F}$, a function $\mathrm{range}(F)$ is defined as follows.*

$$\mathrm{range}(F) = \{x | \exists c, r \,.\, x \in \mathcal{C} \wedge (c, r, x) \in F\} \tag{6.6}$$

The function $\mathrm{range}(F)$ returns the multiset of range concepts from the multiset of facts F.

6.4 Automatic Topic Identification

6.4.1 Extraction of Nouns and Verbs

Given a group header h in an idea chart T, only the $\mathrm{in}(h)$ and $\mathrm{out}(h)$ sets are read as inputs. The set of nouns (including compound nouns and name entities) and a set of verbs appearing in these label sets need to be extracted. POS tagging is used to achieve this, which is the process of tagging a word with a particular lexical category (i.e., noun, pronoun, possessive ending, verb, adjective, or adverb). Only the base form of words are stored in both knowledge bases, so each word is transformed into its base form via lemmatization, e.g., *children* ⇒ *child*, *unsolved problems* ⇒ *unsolved problem*, and *walking* ⇒ *walk*.

Definition 8. *Let W be a set of words and $\psi \in W^*$. Let $Q = \{\texttt{Adjective}, \texttt{Adverb}, \texttt{PossEnding}\}$ and $R = \{\texttt{Noun}, \texttt{ProperNoun}\}$ be sets of lexical categories. Let $\mathcal{N}$ be a set of noun phrases that represents concepts in the YAGO ontology. Let $\mathrm{pos}(X)$ return the order n-tuples lexical*

categories from text X. Let $\mathrm{lemma}(X)$ *returns the lemma sequence from text X. Given a set of labels, a function* $\mathrm{N}(L)$ *is defined as follows.*

$$\mathrm{N}(L) = \bigcup_{l \in L} \mathrm{M}(l) \tag{6.7}$$

$$\mathrm{M}(\psi) = \mathcal{N} \cap \{\mathrm{lemma}(X) \mid X \in \mathrm{sub}(\psi) \wedge \mathrm{pos}(X) \in Q^* R^+\} \tag{6.8}$$

$$\mathrm{sub}(\psi) = \{\psi' \mid \exists \psi_1, \psi_2 \,.\, \psi_1 \psi' \psi_2 = \psi \wedge \psi' \neq \epsilon\} \tag{6.9}$$

The multiset $\mathrm{N}(L)$ is the multiset of all lemmatized noun phrases found in the group of label L. The multiset $\mathrm{M}(\psi)$ is the multiset of lemmatized noun phrases found in a label that exist in YAGO. The set $\mathrm{sub}(\psi)$ is the set of non-empty subsequences of ψ.

Definition 9. *Let W be a set of words and $\psi \in W^*$. Let $\mathcal{V}$ be a set of verbs in the WordNet ontology. Given a set of labels L, a function* $\mathrm{V}(L)$ *is defined as follows.*

$$\mathrm{V}(L) = \bigcup_{l \in L} \mathrm{U}(l) \tag{6.10}$$

$$\mathrm{U}(\psi) = \mathcal{V} \cap \{\mathrm{lemma}(w) \mid w \in \psi \wedge \mathrm{pos}(w) = (\texttt{verb})\} \tag{6.11}$$

the multiset $\mathrm{V}(L)$ is the multiset of all lemmatized verbs found in the group of label L. The multiset $\mathrm{U}(\psi)$ is the multiset of lemmatized verbs found in a label that exists in WordNet.

6.4.2 Mapping of Words to Concepts

Next, the sets of extracted nouns and verbs are mapped to their reference concepts in YAGO and WordNet, respectively. In YAGO, there is a relationship MEANS that maps from a noun phrase to its possible reference concepts [73], for example,

$$(\text{“}\texttt{Apple}\text{”}, \textsc{means}, \texttt{AppleFruit}), (\text{“}\texttt{Apple}\text{”}, \textsc{means}, \texttt{AppleInc.}), \ldots$$

In WordNet, there is a relationship SYNSETS that maps from a verb to its possible concepts [72], for example,

$$(\text{“}\texttt{Require}\text{”}, \textsc{synsets}, \texttt{Demand}), (\text{“}\texttt{Require}\text{”}, \textsc{synsets}, \texttt{Ask}), \ldots$$

Definition 10. *By exploiting both relationships, we can extract the sets of noun and verb concepts from* $\mathrm{N}(L)$ *and* $\mathrm{V}(L)$ *that exist in YAGO and WordNet, respectively. Therefore, the sets of noun and verb concepts are defined as follows.*

$$\mathrm{NC}(L) = \mathrm{range}(\mathrm{NF}(L)) \tag{6.12}$$

$$\mathrm{VC}(L) = \mathrm{range}(\mathrm{VF}(L)) \tag{6.13}$$

$$\mathrm{NF}(L) = \mathrm{fact}(\mathrm{N}(L), \textsc{means}, \mathrm{YAGO}) \tag{6.14}$$

$$\mathrm{VF}(L) = \mathrm{fact}(\mathrm{V}(L), \textsc{synsets}, \mathrm{WordNet}) \tag{6.15}$$

The multisets $\mathrm{NC}(L)$ and $\mathrm{VC}(L)$ contain the noun concepts and verb concepts found in the set of labels L, respectively. The multiset $\mathrm{NF}(L)$ contains the noun facts that map from noun phrases to nouns concepts in YAGO. The multiset $\mathrm{VF}(L)$ contains the verb facts that map from verbs to verb concepts in WordNet.

Both relations are one-to-many, which allows multiple concepts to be extracted from a polyseme. Incorrect word senses may be included in the concept set, which causes ambiguity. This problem will be solved in a later step.

6.4.3 Hierarchical Hypernym Graph Construction

A hierarchical hypernym graph (HHG) is a hierarchical graph of concepts for L that is used during the topic identification process. It consists of two sub-graphs. The first sub-graph is a hypernym tree HT, which provides a pyramidal view of concepts based on their hypernym relationships and HT as a rooted tree. A vertex is a concept. An edge is a leaf-to-root HYPERNYM relationship between two concepts. The root concept is the most abstract concept (e.g., the `Entity` concept in YAGO) and it is the hypernym of every vertex except itself in HT.

The second subgraph is a word to concept graph (CG). This sub-graph is constructed from all facts in $\mathrm{distinct}(\mathrm{NF}(L))$ for nouns and $\mathrm{distinct}(\mathrm{VF}(L))$ for verbs. Note that, $\mathrm{distinct}(X)$ is a function that returns the set of distinct members in the multiset X. A vertex represents a word w and the frequency of word w in set $\mathrm{N}(L)$ if w is a noun, or the frequency of word w in the set $\mathrm{N}(L)$ if w is a verb. An edge represents a relationship between a word w and its reference concepts that appear in the hypernym tree. A sample noun hierarchical hypernym graph is shown in Figure 6.3.

To construct noun and verb hypernym trees for a label group L, all of the hypernym relationships need to be discovered from all concepts in $\mathrm{NC}(L)$ and $\mathrm{VC}(L)$ to the root concept. These are extracted using the following recursive function.

$$\mathrm{H}(X, \mathcal{O}) \;=\; \{F \;\cup\; \mathrm{H}(\mathrm{range}(F), \mathcal{O}) \mid F = \mathrm{fact}(\mathrm{distinct}(X), \textsc{hypernyms}, \mathcal{O})\} \tag{6.16}$$

The function $\mathrm{H}(X, \mathcal{O})$ recursively visits all parents of all concepts in the set of concepts X existing in ontology $\mathcal{O}$, until the root concept is reached. The relationships sets $\mathrm{H}(\mathrm{NC}(L), \mathrm{YAGO})$ and $\mathrm{H}(\mathrm{VC}(L), \mathrm{WordNet})$ contain all of the hypernym relationships from the leaf concept to the root concept in the sets of concepts $\mathrm{NC}(L)$ and $\mathrm{VC}(L)$, respectively. The noun and verb hypernym trees are constructed from the $\mathrm{H}(\mathrm{NC}(L), \mathrm{YAGO})$ and $\mathrm{H}(\mathrm{VC}(L), \mathrm{WordNet})$ relationships sets, respectively. WordNet has multiple root concepts, so the number of verb HTs can be greater than one, whereas the number of noun HTs is always one.

6.4.4 Topic Selection Methods

To determine a plausible topic for h in an idea chart T, the following methods evaluate each concept in the $\mathrm{HT}_{\mathrm{in}(h)\cup\mathrm{out}(h)}$ and selects one of them as a topic for h.

Branch Ratio (BR)

Originally proposed by Lin [74], the branch ratio (BR) selects the concept with the highest generalization power. The BR of concept x is defined as follows.

$$\mathrm{BR}(x) \;=\; \frac{\max_{y\in\mathrm{child}(x)} \mathrm{w}_{\mathrm{in}(h)}(y)}{\sum_{y\in\mathrm{child}(x)} \mathrm{w}_{\mathrm{in}(h)}(y)} \tag{6.17}$$

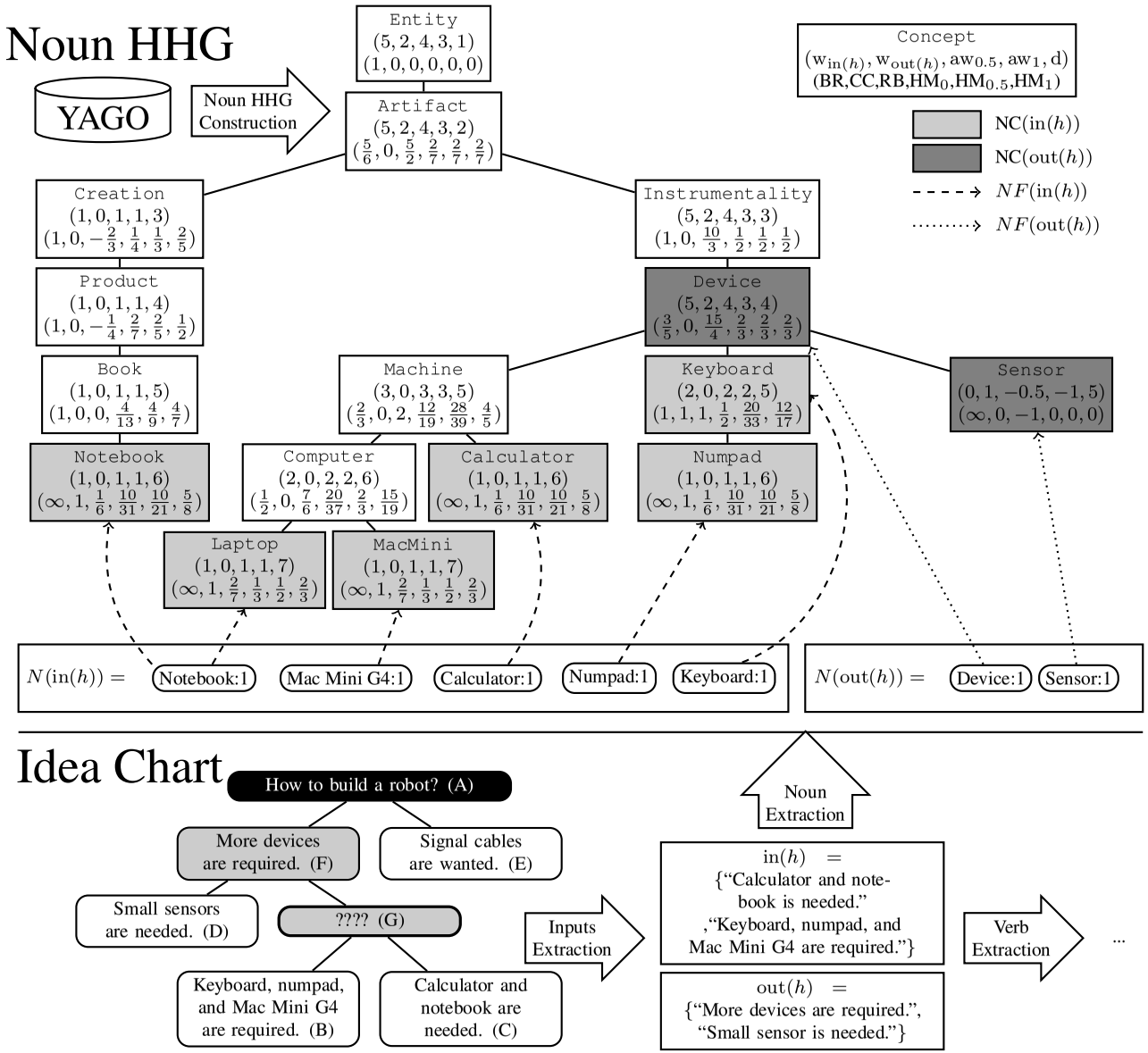


Figure 6.3: In the upper half, the noun hierarchical hypernym graph $HHG_{\mathrm{in}(h)\cup\mathrm{out}(h)}$ where h is the concept "???? (G)" is shown. The solid line graph is the hypernym tree $HT_{\mathrm{in}(h)\cup\mathrm{out}(h)}$. The dashed line graph is the word to concept graph $CG_{\mathrm{in}(h)\cup\mathrm{out}(h)}$. In the lower half, the tree representation of the idea graph is displayed on the left hand side. The diagram on the right hand side shows the result of the input extraction $\mathrm{in}(h)$ and $\mathrm{out}(\mathrm{h})$. The numbers in parentheses are the values of variables and six topic selection measures corresponding to each concept where they are located. The order of variables is shown at the legend box located at the top-right corner of this figure.

where

$$
\begin{aligned}
\mathrm{w}_L(x) &= \begin{cases} \sum_{w \in \mathrm{allword}_L(x)} \mathbf{1}_{\mathrm{N}(L)}(w) & \text{if x is a noun.} \\ \sum_{w \in \mathrm{allword}_L(x)} \mathbf{1}_{\mathrm{V}(L)}(w) & \text{if x is a verb.} \end{cases} \\
\mathrm{allword}_L(x) &= \{\mathrm{word}_L(x) \cup \mathrm{word}_L(y) \mid y \in \mathrm{allchild}(x)\} \\
\mathrm{word}_L(x) &= \begin{cases} \{d \mid \exists d, r \,.\, (d, r, x) \in \mathrm{NF}(L)\} & \text{if x is a noun.} \\ \{d \mid \exists d, r \,.\, (d, r, x) \in \mathrm{VF}(L)\} & \text{if x is a verb.} \end{cases} \\
\mathrm{allchild}(x) &= \{a \cup \mathrm{allchild}(a) \mid a \in \mathrm{child}(x)\} \\
\mathbf{1}_A(x) &= \begin{cases} 1 & x \in A \\ 0 & \text{otherwise} \end{cases}
\end{aligned}
$$

The function $\mathrm{child}(x)$ returns the set of children of x. The function $\mathrm{allchild}(x)$ returns the concepts set that have upward paths to x. The function $\mathrm{word}_L(x)$ returns the word set in L that refers to x. The function $\mathrm{allword}_L(x)$ returns the word set in L that refers to x or its children. The weight function $\mathrm{w}_L(x)$ returns the total word frequency of $\mathrm{allword}_L(x)$ that appear in $\mathrm{N}(L)$ or $\mathrm{V}(L)$. In other words, the function $\mathrm{w}_L(x)$ returns the total frequency of words that refer to x and its children. Figure 6.3 shows the $w_{\mathrm{in}(h)}$ and the value of the BR in the sample hierarchical hypernym graph. The value of BR in the `Device` concept is computed based on the $w_{\mathrm{in}(h)}$ of the `Machine` concept divided by the total $w_{\mathrm{in}(h)}$ for the `Machine`, `Keyboard`, and `Sensor` concepts, i.e., $\frac{3}{3+2+0} = \frac{3}{5}$.

Note that, the leaf concepts always yield ∞. A concept that yields a low BR is a suitable topic choice. A lower BR implies a higher generalization power for its children concepts. A concept that is > 0.68 is considered insignificant [74] and they are ignored. The concept that yields the lowest branch ratio is selected.

Concept Counting (CC)

Originally proposed by Lin [74], the concept counting (CC) is a simple and efficient topic selection method. CC is the frequency of concept x in $\mathrm{in}(h)$, which is defined as follows:

$$
CC(x) = \begin{cases} \mathbf{N}_{\mathrm{NC}(\mathrm{in}(h))}(x) & \text{if x is a noun.} \\ \mathbf{N}_{\mathrm{VC}(\mathrm{in}(h))}(x) & \text{if x is a verb.} \\ 0 & \text{otherwise} \end{cases} \tag{6.18}
$$

where

$$
\mathbf{N}_A(x) = \sum_{w \in A} \begin{cases} 1 & w = x \\ 0 & \text{otherwise} \end{cases}
$$

A higher CC is better. The concept with the highest frequency is selected as the topic. Figure 6.3 shows the CC values for each concept. The CC of the `Device` concept is 0 because there no word in $\mathrm{N}(\mathrm{in}(h))$ is mapped to this concept.

Ratio Balance (RB)

Originally proposed by Tiun et al. [75], the ratio balance (RB) requires that the accumulated weight is higher than that expected for each level. The RB of concept x is defined as follows:

$$
\mathrm{RB}(x) = \mathrm{w}_{\mathrm{in}(h)}(x) - \frac{\mathrm{w}_{\mathrm{in}(h)}(\mathrm{R}_{HT})}{\mathrm{d}(x)} \tag{6.19}
$$

where the depth function $\mathrm{d}(x)$ returns the depth of x in HT (the depth of a root concept is one). A higher RB is better. The concept with the highest RB is selected as the topic. Figure 6.3 shows the RB values for each concept. The RB of the `Device` concept is computed from its $\mathrm{w}_{in(h)}$ value based on the expected value, which is computed from the $\mathrm{w}_{in(h)}$ of the `Entity` concept divided by the depth of the `Device` concept, i.e., $5 - \frac{5}{4} = \frac{15}{4}$.

Harmonic Mean (HM)

All previous methods have identified topics based on a consideration of $\mathrm{in}(h)$ alone, which is the set of labels a topic describes. The set of idea labels that are not mentioned in h (i.e., $\mathrm{out}(h)$) have not been used before. The $\mathrm{out}(h)$ set provides clear boundaries for the topic of h. We propose that the harmonic mean (HM) can be used to evaluate each topic based on a consideration of the $\mathrm{in}(h)$ and $\mathrm{out}(h)$ sets.

Given a concept x, the adjusted weight of the concept x is defined as follows.

$$\mathrm{aw}_\alpha(x) \quad = \quad \mathrm{w}_{\mathrm{in}(h)}(x) - \alpha\, \mathrm{w}_{\mathrm{out}(h)}(x) \tag{6.20}$$

The constant α is the penalty rate that indicates the degree of deduction for each occurrence of words in $\mathrm{out}(h)$ that map to the children of the concept x or concept x itself.

A concept in a hierarchical hypernym tree with a high adjusted weight and a high depth is considered to be a plausible topic because it covers a high proportion of the words in $\mathrm{in}(h)$ compared with the number of words in $\mathrm{out}(h)$ and it has high specificity. HM considers the adjusted weight and the depth of a concept equally, which can be interpreted as the weighted average of the adjusted weight and depth.

Given the concept x, HM of x is defined as follows.

$$\mathrm{HM}_\alpha(x) \quad = \quad \frac{2 \times \mathrm{naw}_\alpha(x) \times \mathrm{nd}(x)}{\mathrm{naw}_\alpha(x) + \mathrm{nd}(x)} \tag{6.21}$$

where

$$\begin{aligned} \mathrm{naw}_\alpha(x) &= \frac{\mathrm{aw}_\alpha(x) - \mathbf{min}_{m \in HT}\, \mathrm{aw}_\alpha(m)}{\mathbf{max}_{m \in HT}\, \mathrm{aw}_\alpha(m) - \mathbf{min}_{m \in HT}\, \mathrm{aw}_\alpha(m)} \\ \mathrm{nd}(x) &= \frac{\mathrm{d}(x) - 1}{\mathbf{max}_{m \in HT}\, \mathrm{d}(m) - 1} \end{aligned}$$

HM is undefined for negative numbers, so the weight and depth need to be normalized in the range $[0, 1]$. The HM values with three different α values ($\alpha = 0$, $\alpha = 0.5$, $\alpha = 1$) are shown in Figure 6.3. To compute $\mathrm{HM}_{0.5}(\texttt{Device})$, we compute $\mathrm{naw}_{0.5}(\texttt{Device}) = \frac{4-(-0.5)}{4-(-0.5)} = 1$ and $\mathrm{nd}(\texttt{Device}) = \frac{4-1}{7-1} = \frac{1}{2}$. Thus, $\mathrm{HM}_{0.5}(\texttt{Device}) = \frac{2 \times 1 \times \frac{1}{2}}{1+\frac{1}{2}} = \frac{2}{3}$ is yielded. A concept with a high HM is plausible. The concept with the highest HM is selected.

6.4.5 More Non-overlapping Topics

Any text usually mentions one or more topics. To identify multiple topics, the most naïve approach is to select the top-n concepts evaluated by a topic selection method. As shown in Figure 6.3, the first and the second ranks are `Machine` and `Device` based on the identification of the first and second topics using HM_1. Both concepts are similar and `Device` is the hypernym

of `Machine`, so `Device` should not be chosen as the second topic. Instead, another concept should be selected that does not overlap with the previously selected topics.

We follow the method proposed by Lin [74]. After the first topic is selected, we extract the next most significant non-overlapping topic, which does not share any commonalities with the first topic, by removing all of the words in the CG that refer to the children of the first topic in the HT. Next, the HHG is re-constructed and we re-evaluate every concept using the same topic selection method. The best concept in the new graph is chosen as the second topic. This process can be repeated to select the third topic, the fourth topic, and so on. Figure 6.4 shows the pruned HHG produced by extracting the second non-overlapping topic from the original HHG (Figure 6.3). The `Keyboard` is chosen as the second topic using the HM_1 method.

This process performs word sense disambiguation for polysemes simultaneously. Figure 6.3 shows that the "Notebook" word has two word senses, i.e., `Laptop` and `Notebook`. `Laptop` is much more likely to be the correct sense because the majority of the content relates to the `Machine`. This process removes all other senses (e.g., `Notebook`), after the correct sense (e.g., `Laptop`) is selected.

6.5 Usage in Automatic Category Naming

Automatic topic identification can be used as a component of automatic category naming, which is an algorithm for automatically predicting the title of a category. Automatic category naming has two stages. Automatic topic identification is the first stage of automatic category naming and followed by the category title generator (a natural language generator) for constructing a plausible category title, which can be sentences or noun phrases. The noun and verb topics yielded from the automatic topic identification are fed into the category title generator. A category title generator can be as simple as concatenating the noun and verb topics, for example, the noun topic is a "Machine" and the verb topic is a "Need"; the category title "Machine need" is automatically produced and replaces the title "????" of an untitled category.

If the automatic category naming is not accurate enough, instead of directly replacing an "????" untitled category, the noun and verb topics yielded from the automatic topic identification can be displayed for assisting the user in naming a category as shown in Figure 6.5 (in the category naming assistant box). Users can freely type a category title that he/she wants.

6.6 Empirical Evaluations

The proposed topic selection methods (HM_0, $HM_{0.5}$, and HM_1) and existing topic selection methods (BR, CC, and RB) were evaluated empirically using idea charts produced by two creativity techniques, i.e., the KJ Method and mind mapping. Automatic category naming based on the proposed automatic topic identification system was implemented for the evaluation. The POS tagging and lemmatization of nouns and verbs during the extraction step were performed using Stanford CoreNLP[1].

[1] http://nlp.stanford.edu/software/corenlp.shtml

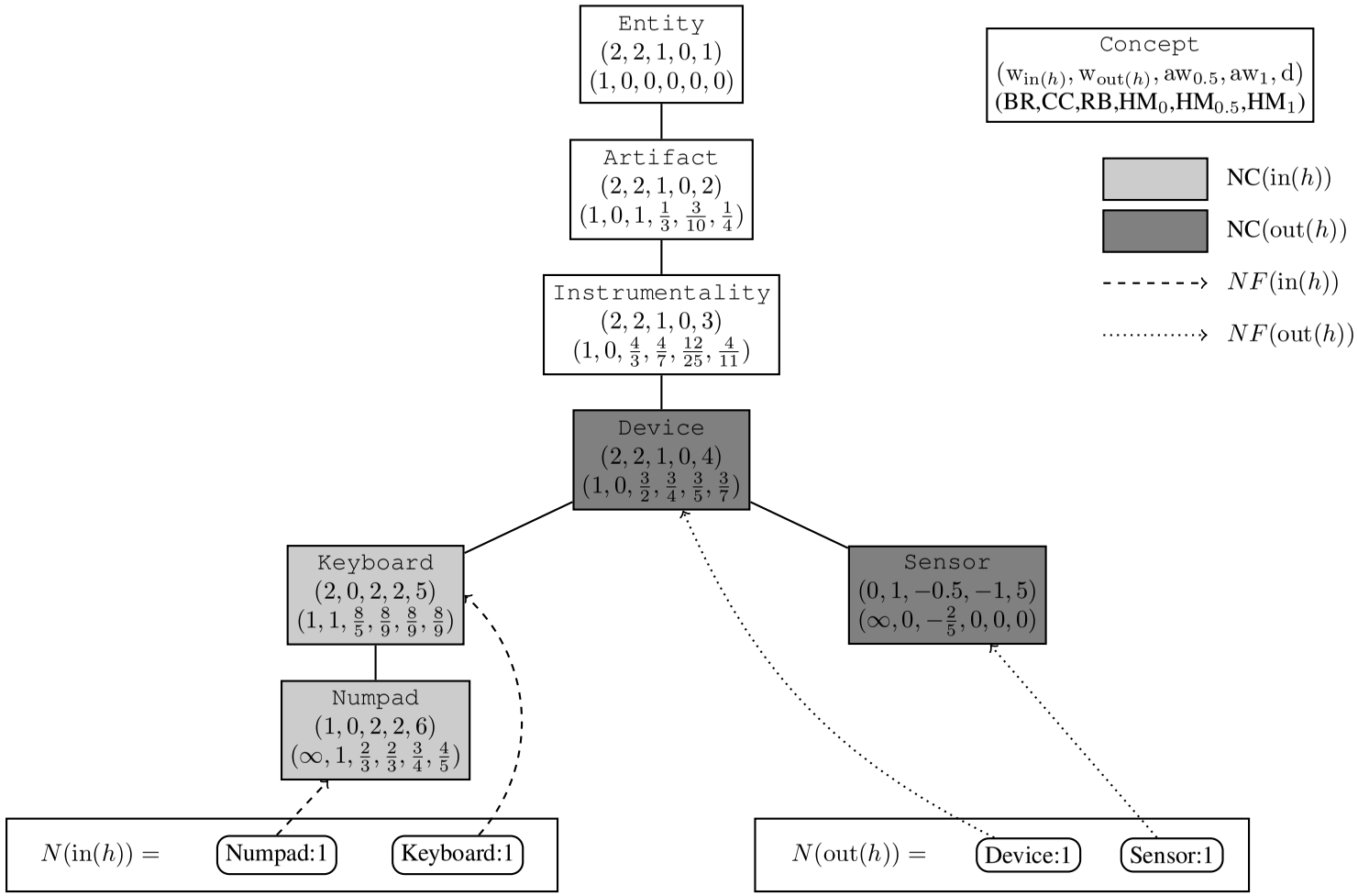


Figure 6.4: The pruned hierarchical hypernym graph (original is Fig 2) used to extract the second topic.

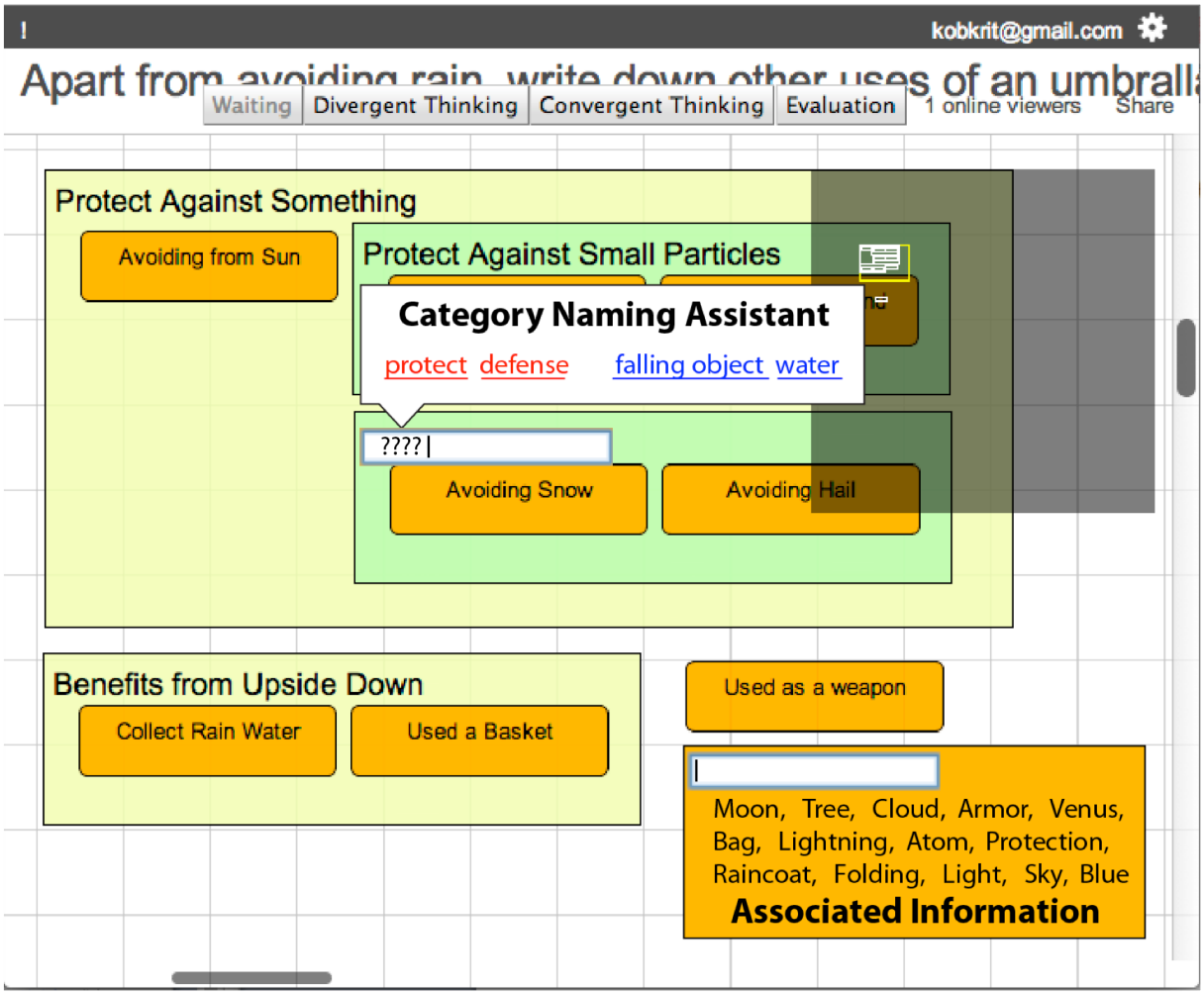


Figure 6.5: A screen of Gugeek groupware when the creativity assistants are activated.

6.6.1 KJ Method

The experiments were conducted with 87 participants who were Masters or Ph.D. students and who understand the KJ Method well. The Gugeek, a KJ-Method support groupware developed as a Web application platform (see Section 4.3), is used for allowing participants to construct an idea chart (KJ chart) using their computer. Participants were asked to create at least one idea chart for any problem of interest.

The participants input their ideas into the system to create virtual labels and performed the drag-and-drop operation between two labels to form a new group. When a group was formed, a blank group header was automatically created. When a blank group header was clicked, the tree representation of the current chart and the position of the clicked group header were fed into the automatic topic identification module. Next, automatic topic identification as described in Section 6.4 for automatic category naming was performed.

Lists of pairs of the first and the second identified noun topics produced by the five selection methods (as described in Section 6.4.4 were displayed inside a pop-up dialog on the screen as shown in Figure 6.6. The initial order of pairs was random. Participants were asked to rank these pairs based on their relevance to the group's content by clicking on the pairs from the first rank to the last rank on the left queue. When a pair was clicked, it was dequeues from the left queue and enqueues into the right queue. The queue was sorted according to the pairs' ranks ascending from the top to the bottom.

According to the Figure 6.6, the green and red text on the right sidebar was the label content appeared as information to the user. The green text showed the content of labels that are inside the naming category. The red text showed the content of labels that are outside the naming category by one level (siblings and their parent). It supported participants to accurately choose the correct word to describe the content of labels that are inside the naming category and avoid mentioning the content of labels that are outside the naming category. When all pairs were ranked (the left queues were empty), participants were asked to input the most suitable topic noun in their mind in the text box at the bottom of the dialog to assess their opinions and measure system satisfaction, and then click the next button.

In the next screen, list of pairs of the first and the second identified verb topics are displayed. Participants rank these pairs according to their relevance to the group content and input the most suitable topic verb in their mind, as in the previous screen for the noun topics. The results of the ranking and system satisfaction assessment were analyzed to evaluate the five topic selection methods.

In total, 1,934 labels and 583 group headers were drawn in 112 idea charts by the participants. The average numbers of HTs in an HHG were 1 in noun topics and 24.89 in verb topics. The average depth of HTs was 5.12 and 1.56 for noun and verb topics, respectively. The average number of concepts found in the HTs was 97.80 and 86.17 for the noun and verb topics, respectively.

Based on the preliminary requirement that $BR \leq 0.68$, some group headers with no concepts satisfied this requirement. To ensure fair comparison, these groups were removed from the evaluation. Only 339 and 373 group headers remained for evaluation in the noun and verb topics, respectively.

Topic Ranks

Table 6.1 shows the ranking results produced by the five topic selection methods for the noun and verb topics (lower results were better). Wilcoxon's rank sum test [83], which is a non-

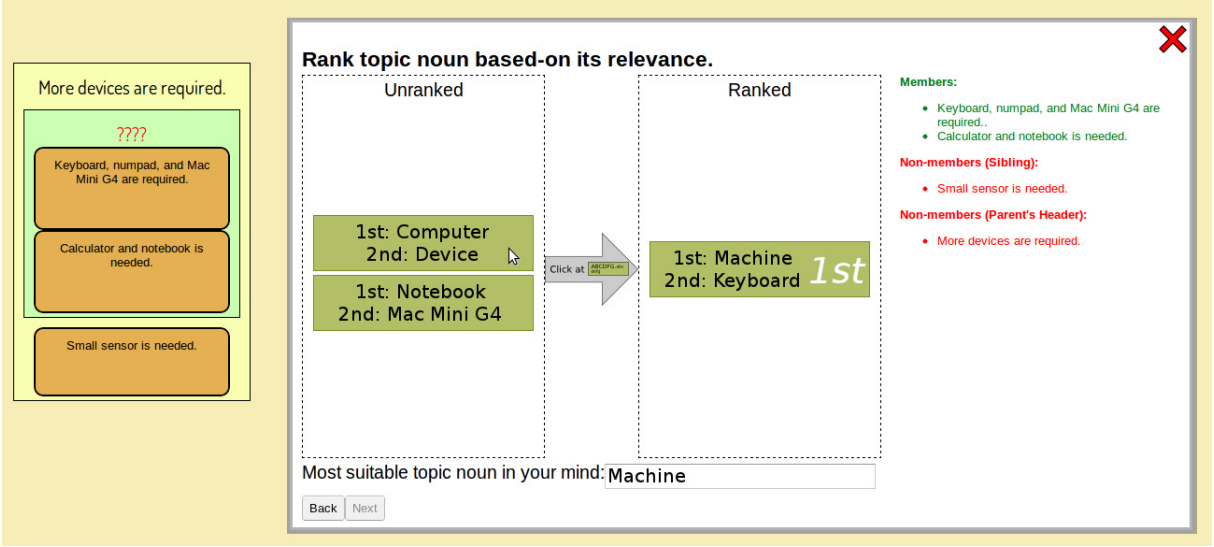


Figure 6.6: Ranking screen during the automatic category naming experiment

Table 6.1: Rank results for all topic selection methods except the RB in the KJ Method experiment (the lower is the better)

POS	BR	CC	HM		
			$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$
Noun	3.904 ± 1.423	$2.863 \pm 1.613^{*}$	$2.763 \pm 1.019^{*}$	$\mathbf{2.721 \pm 0.877}^{*}$	$2.749 \pm 1.013^{*}$
Verb	3.182 ± 1.480	$\mathbf{2.489 \pm 1.461}^{*}$	$3.676 \pm 1.393^{*}$	$2.838 \pm 0.984^{*+}$	$2.815 \pm 1.026^{*+}$

$^{*,+}$ Significance $P \leq 0.05$

parametric statistical hypothesis test, was used to test whether the rank results for each topic selection method were significantly different. The performance with BR was the baseline. Any ranks were significantly better than the baseline rank are marked with the * symbol. $HM_{0.5}$ and HM_1 with significantly higher ranks than HM_0 are tagged with the $^{+}$ symbol. Wilcoxon's rank sum test was used to compare all of the experimental results. * and $^{+}$ symbols are used throughout the results. Note that RB is absent from some experiment results due to a technical issue.

Table 6.2: Suggestion Accuracy in the KJ Method experiment

POS	# Headers	BR	CC	RB	HM		
					$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$
Noun Top-1	339	20.944%	$43.363\%^{*}$	$33.038\%^{*}$	$38.348\%^{*}$	$42.478\%^{*+}$	$\mathbf{43.658\%}^{*+}$
Noun Top-2	339	23.894%	$49.263\%^{*}$	$42.773\%^{*}$	$48.378\%^{*}$	$51.622\%^{*}$	$\mathbf{52.802\%}^{*+}$
Verb Top-1	373	22.520%	$\mathbf{49.062\%}^{*}$	23.592%	23.324%	$36.729\%^{*+}$	$37.265\%^{*+}$
Verb Top-2	373	28.418%	$55.764\%^{*}$	25.201%	25.737%	$\mathbf{56.568\%}^{*+}$	$\mathbf{56.568\%}^{*+}$

$^{*,+}$ Significance $P \leq 0.05$

Suggestion Accuracy

Suggestion accuracy is the percentage of noun and verb topics predicted by our algorithm matched with the correct noun and verb topics. The correct noun and verb topics provided manually by participants were matched with the five identified topics generated by the five topic selection methods. For Top-1, only the first identified topic was compared with the correct topic. For Top-2, the first and the second identified topics were compared with the correct topic. In comparison, if at least one identified topic is found as a word of the correct topics (word-by-word comparison, not the substring matching), the proposed algorithm is considered to suggests correctly.

Stemmer is a software that derived the root forms of words (stems) by chopping off the ends of words using crude heuristic rules, which is the same purpose of the lemmatization. Lemmatization derived the root forms of words by language and morphological analysis which requires much more high computation power and is generally slower than the stemmer. But the stemmer is sufficient enough for related words mapped to the same root form [84]. Therefore, before making comparisons, porter stemmer[85], was applied to predict and correct topics for broadening the matches. Table 6.2 shows the suggestion accuracy with the five topic selection methods.

Average Word Similarity

Instead of comparing the suggestion accuracy directly, we computed the average word similarity score of the Cartesian product of the set of all words in the first/second identified concepts and the set of all words in the correct topic. Lin's word similarity algorithm [86] was applied. For Top-1, only the first identified concept was computed. For Top-2, the average score of both the first and the second identified concepts was computed. The average score ranges from 0 to 1. Table 6.3 shows the average similarity scores with the five topic selection methods.

6.6.2 Mind Mapping

Xmind is a mind mapping program that was released in 2007, which allows users to export mind maps as computer files. Xmind.net Share [2] is a mind mapping sharing Website that makes approximately 40,000 idea charts (mind maps) publicly available for download. In the experiment, only English idea charts were downloaded. We extracted the tree representation, the positions of all group headers, and the content written in all of the group headers in each idea chart. The tree representation and the positions of all group headers were input into our implemented system and the automatic topic identification process was performed.

The pairs of the first and the second predicted topics identified using the five topic selection methods were output and compared with the content of the group headers. The suggestion accuracy and the average word similarity used in the KJ Method experiment were also applied with the same settings in this experiment.

We extracted 339,765 labels and 197,505 group headers from 9,575 English idea charts. Due to the preliminary requirement that $BR \leq 0.68$, the group headers were reduced to 96,468 and 62,378 for nouns and verbs, respectively. Table 6.4 shows the suggestion accuracy using

[2] `http://www.xmind.net/share/`

the five selection methods in the mind mapping experiment. Table 6.5 shows the average word similarity scores for the five selection methods in the mind mapping experiment.

6.7 Discussion

The experimental results showed that, CC, RB, and HM delivered significantly higher topic identification performance than BR (* symbols). The $HM_{0.5}$ and the CC yielded the best ranked evaluations by participants during noun and verb topic identification, respectively. CC yielded the highest suggestion accuracy in most cases, while $HM_{0.5}$ yielded the highest average word similarity score in most cases. The inclusion of the set of negative labels in the calculations ($HM_{0.5}$ and HM_1) delivered a significantly higher performance compared to its exclusion (HM_0) from verb topics identification ($^+$ symbols).

The performance of HM was not significantly different from CC, but HM had some advantages over CC. HM could discover topic concepts that were represented by words not appearing in the set of labels. In our opinion, these concepts were more plausible than those in the more frequent words because they are the generalization of idea groups using different words.

In contrast to BR, which never considered leaf concepts or any concepts that yield $BR > 0.68$ in the HHG as a topic candidates, HM lacked these requirements. The amount of text per idea group is usually short in idea visualization programs. Most are simple phrases, using a small amount of extracted concepts. BR failed to make any suggestions in most cases, whereas HM always managed to generate a suggestion even when only one concept was found.

Lastly, the difficulties of our proposed method found in the experiment and the experimental results are illustrated as a KJ-Chart in Figure 6.7.

6.7.1 Criticism

The proposed algorithm can be found to be different from the intention of the label grouping step in the KJ-Method described in Chapter 2. Label grouping in the KJ-Method is based on any mental association (connections) between two or more labels. These associations can be anything from referring to the same thing, belonging to the same category, sharing the same object, sharing the same action, causing the same effect, having the same value, having a similar pattern, etc.

According to the preliminary observation, the most frequent connections found during the label grouping steps is labels having the same category. Therefore, I started to focus on this kind of connection for automatic category naming. The proposed automatic category naming, based on the hypernym structures found in ontologies, is fitted to the scenario of groups of labels belonging to the same category connection only.

For other kinds of connections, it can be future works. The most challenging is the lack of knowledge base for supplying information about how two concepts are related. For example, if we focus on the causing of the same effect relationship, the knowledge base of the cause-and-effect for two concepts must be supplied, and they are unavailable at this time, and also seem unlikely to be available in the near future.

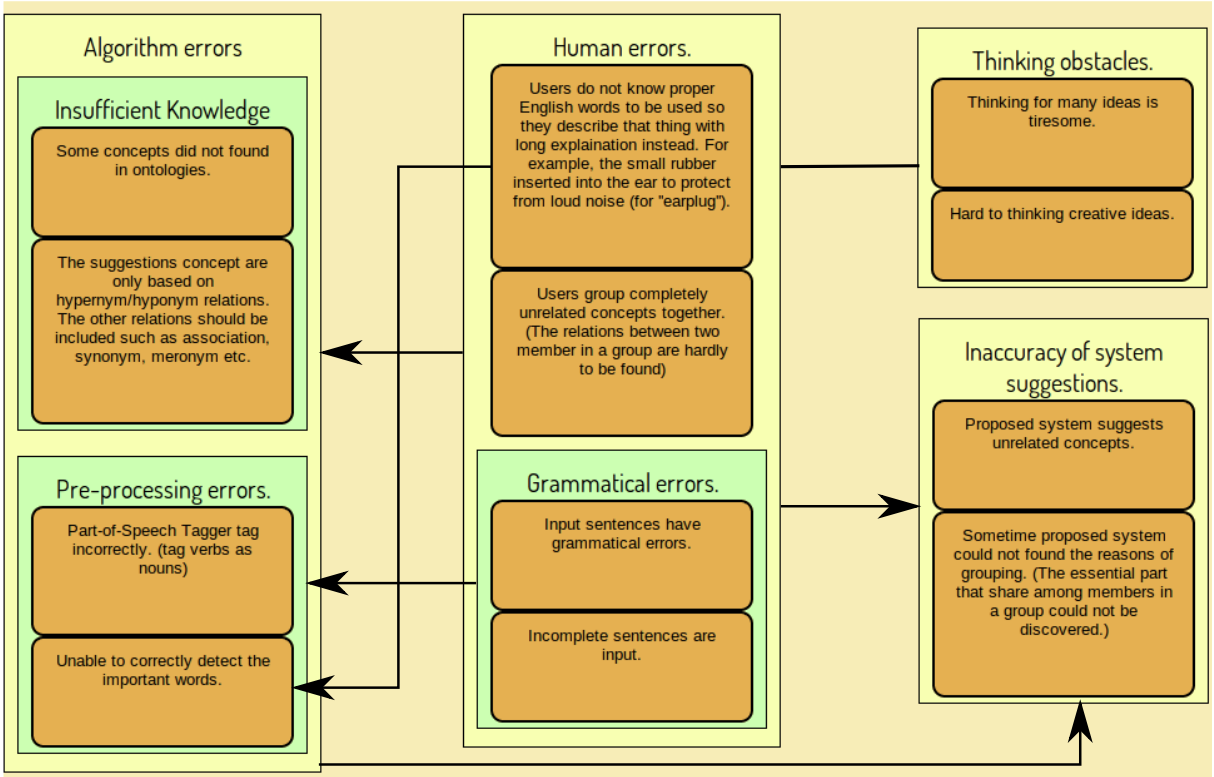


Figure 6.7: Difficulties found in the proposed convergent thinking support.

Table 6.3: Average word similarity score [0,1] between predicted topics and actual topics in the KJ Method experiment $(\times 10^{-2})$

POS	BR	CC	RB	HM		
				$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$
Noun Top-1	16.503 ± 22.256	$19.459 \pm 24.128^{*}$	$20.672 \pm 26.291^{*}$	$20.929 \pm 26.806^{*}$	$22.271 \pm 28.305^{*}$	$\mathbf{22.553} \pm \mathbf{28.485}^{*}$
Noun Top-2	9.871 ± 11.966	$14.105 \pm 13.862^{*}$	$15.112 \pm 15.614^{*}$	$15.441 \pm 16.205^{*}$	$15.596 \pm 15.639^{*}$	$\mathbf{15.673} \pm \mathbf{15.616}^{*}$
Verb Top-1	4.735 ± 7.430	$\mathbf{8.967} \pm \mathbf{13.804}^{*}$	3.930 ± 7.924	4.307 ± 8.710	$7.816 \pm 11.814^{*+}$	$7.717 \pm 11.630^{*+}$
Verb Top-2	3.669 ± 4.974	$5.607 \pm 7.475^{*}$	2.241 ± 4.112	2.534 ± 4.851	$\mathbf{6.347} \pm \mathbf{7.542}^{*+}$	$6.252 \pm 7.233^{*+}$

*,$^{+}$ Significance at $P \leq 0.05$

Table 6.4: Suggestion Accuracy in the Mind Mapping Experiment

POS	**# Group Headers**	**BR**	**CC**	**RB**	**HM**		
					$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$
Noun Top-1	96,468	1.254%	$\mathbf{7.838}\%^{*}$	$3.527\%^{*}$	$3.819\%^{*}$	$4.619\%^{*+}$	$4.850\%^{*+}$
Noun Top-2	96,468	1.727%	$\mathbf{11.469}\%^{*}$	$6.687\%^{*}$	$6.909\%^{*}$	$7.755\%^{*+}$	$7.869\%^{*+}$
Verb Top-1	62,378	0.891%	$\mathbf{2.830}\%^{*}$	$1.132\%^{*}$	$1.380\%^{*}$	$1.917\%^{*+}$	$1.844\%^{*+}$
Verb Top-2	62,378	1.459%	$\mathbf{4.050}\%^{*}$	$1.999\%^{*}$	$1.757\%^{*}$	$3.413\%^{*+}$	$3.339\%^{*+}$

*,$^{+}$ Significance $P \leq 0.05$

Table 6.5: Average word similarity score [0,1] between predicted topics and actual topics in the Mind Mapping Experiment ($\times 10^{-2}$)

POS	**BR**	**CC**	**RB**	**HM**		
				$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$
Noun Top-1	4.816 ± 6.374	4.622 ± 8.910	4.959 ± 7.525	$\mathbf{5.006} \pm \mathbf{7.750}$	4.810 ± 7.943	4.723 ± 8.021
Noun Top-2	3.671 ± 4.282	$4.364 \pm 6.504^{*}$	$4.735 \pm 5.560^{*}$	$\mathbf{4.787} \pm \mathbf{5.687}^{*}$	$4.487 \pm 5.668^{*}$	$4.374 \pm 5.665^{*}$
Verb Top-1	0.786 ± 2.266	0.933 ± 3.286	$1.035 \pm 3.314^{*}$	0.775 ± 2.652	$\mathbf{1.038} \pm \mathbf{3.551}^{*+}$	$1.020 \pm 3.506^{*+}$
Verb Top-2	0.694 ± 1.696	0.757 ± 2.332	$0.839 \pm 2.365^{*}$	0.532 ± 1.702	$\mathbf{0.903} \pm \mathbf{2.548}^{*+}$	$0.891 \pm 2.543^{*+}$

$^{*,+}$ Significance at $P \leq 0.05$

6.8 Conclusion

This chapter proposed a novel selection method, harmonic mean, for knowledge-based automatic topic identification, based on Lins framework. It is invented for an automatic idea summarization system, which can support users in the convergent thinking phase. HM is designed to identify topics in extremely short text extracts, which is usual in idea visualization programs. HM was compared with BR, which was proposed by Lin, and CC, which is a powerful statistical-based method.

Two experiments were performed using datasets from two idea visualization programs, i.e., the KJ Method and mind mapping. CC and HM significantly outperformed BR in terms of performance and consistency. CC and HM differed little in terms of performance, but HM discovered different plausible topics that did not appear in the text, whereas CC could not.

According to the experimental result, our proposed automatic category naming can correctly suggest a category title around 43.658% for nouns and around 49.062% in verbs in the KJ-Method (as shown in Table 6.2). In the experiment, we used a simple text concatenation method for making titles. We believed that there is room for more sophisticated techniques in this research problem which might yielded higher performance in acceptable accuracy in the near future.

Part III

Social Influences in KJ Method Groupware

Chapter 7

Team Communications

7.1 Introduction

Every innovation team in an organization regardless of size, motive, and goal is regularly pressured to create products or service offerings. Team creativity is the key resource driving the development process to achieve successful products. The effective techniques or systems for stimulating team creativity are strongly desired in many organizations.

With the recent developments in information and communication technology, a creativity support groupware is invented to support team members to communicate, discuss, and perform group creative techniques over geographical distances and different time availabilities to yield creative, well-established solutions for accomplishing team goals. To yield such creative solutions, a team is required to collect, combine and integrate ideas from members which need intensive discussion and team communication. The communication among team members is a crucial factor that has a high impact on the effectiveness of creativity support groupware.

Due to increasing high-level intellectual activities nowadays, creativity support groupware is gaining attention from many researchers. There are many studies revealing factors that affect team creativity performance through groupware use such as team size [24], degree of idea exposure [87], communication frequency and patterns [88], and experience of the teams past successes [89].

Although there are many research works in creativity support groupware, little is known about the effects of communication methods on team creativity. This chapter studies three communication methods commonly found in typical collaborative software, such as task interactions only (chart interactions), text chat, and face-to-face methods. The effectiveness of these communication methods is measured by both quantitative and qualitative evaluation experiments in a KJ Method creativity support groupware carried out by 30 innovative teams. Lastly, the results obtained are examined for statistical significance and a conclusion is drawn.

This chapter is organized as follows: Section 7.2 explains the three communication methods; Section 7.3 shows the degree of creativity evaluators and experimental settings; Section 7.4 gives the experimental results; Section 7.5 discusses the experimental results; Section 7.6 concludes this chapter.

7.2 Team Communication Methods

The following three communication methods are commonly found in most collaborative software ordered from highly restricted to unrestricted.

1. **Chart Interaction Only (CI):** This method allows members to communicate via chart interaction in a shared KJ chart only. Every action performed by a member is propagated to all members in the same team. All members see the changes happening to all labels and all categories in real time. Note that the name of a member who performs an action is not shown.

2. **Chart Interaction + Text Chat (TC):** This method allows members to communicate via chart interaction (Method 1) and online group chat. The online group chat is always shown at the right sidebar of the screen as shown in Figure 7.1. Once a message is received, the system beeps to notify a user. Note that only text can be transmitted.

3. **Chart Interaction + Face-to-Face (FF):** This method allows members to communicate via chart interaction (Method 1) and face-to-face communication. Members are met in person to verbally discuss each other as shown in Figure 7.2. Note that other communication channels are not allowed (such as pointing another members screen or drawing a figure on a whiteboard).

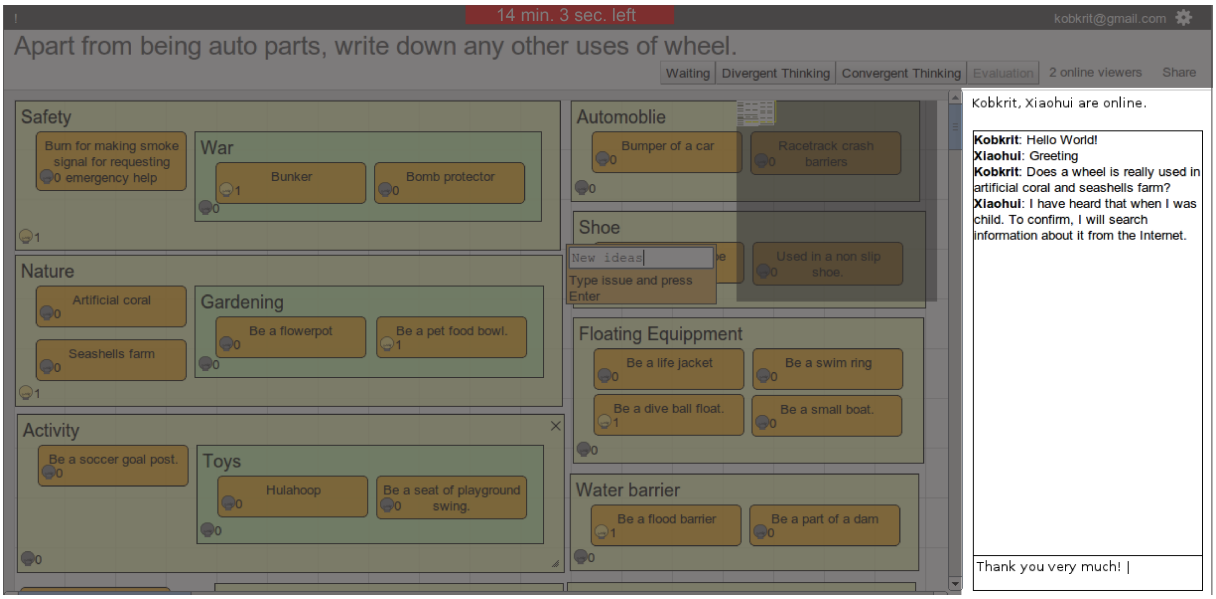


Figure 7.1: Text chat box located at the right sidebar of the Gugeek screen.

7.3 Experimental Settings

7.3.1 Experimental Procedure

Gugeek, a KJ-Method support groupware, is used for the experiment. The total number of participants in the experiment is 90. Participants consist of master and doctoral students who study

Figure 7.2: Team members perform the KJ-Chart and discuss the problem by using Chart Interaction and Face-to-Face (FF) communication methods during an experiment

at the School of Knowledge Science and School of Information Science of Japan Advanced Institute of Science and Technology. All participants must have English testing scores equivalent to a TOEFL score of 550 or more. The experiment consists of ten sessions. In each session, nine participants are divided into three teams of three participants each.

All members in a team were seated at the same table and were asked to access the KJ Method creativity support groupware via their own portable computers (one computer per participant). Before the experiment, every participant is informed of the following:

1. KJ Method's purposes, procedures, and rules.
2. Three communication methods and Gugeek instructions.
3. Examples and Q&A.

For each session, every team performed the group KJ method on the same set of three problems with different problem types. The problem types are listed as follows:

1. **Artifact Synthesis/Unusual used (AS):** "Imagine the new kitchenware used in the future" and "Apart from avoiding rain, write down any other uses of an umbrella".
2. **Problem Solving (PS):** "How can we solve political corruptions?".
3. **Knowledge Gathering on a Topic (KG):** Members contribute their knowledge of a given topic and help each other to construct a KJ chart representing the team's knowledge of that topic, for example, "Fast foods".

Three team communication methods were used as described in Section 7.2. A log of their chart interaction and text chats was collected. The face-to-face conversations among members were recorded by video camera. To avoid the effects of tool experiences, the group communication methods and problems were assigned to teams in a different order as shown in Table 7.1.

For evaluation purposes, the ten-minute and five-minute time-outs were set for the divergent and convergent thinking phases respectively. Once the time-out is reached, the current thinking

Table 7.1: Group communication methods and problems types assignment.

Team	Problem 1	Problem 2	Problem 3
1	CI	TC	FF
2	TC	FF	CI
3	FF	CI	TC

phase moves forward to the next automatically. In the evaluation stage, instead of using the original voting system, we used "Participant-cross evaluation" to prevent the bias in the experimental result described in Section 7.3.3. At the end of the session, the questionnaire was shown on their computer screens. Every participant input his/her opinions and was dismissed.

7.3.2 Quantitative Evaluation

The effectiveness of a communication method is measured by the degree of creativity of KJ charts constructed by teams by that communication method. As proposed by Guilford [70] and being used in Torrance test of creativity performance [90], the degree of creativity can be measured by fluency, flexibility, originality, and elaboration of ideas. These are described as follows:

- **Fluency of Ideas:** The total number of non-redundant ideas.
- **Flexibility of Ideas:** The breadth of ideas.
- **Originality of Ideas:** The statistical rarity of the ideas.
- **Elaboration of Ideas:** The amount of detail in ideas.

We decided to evaluate the degree of creativity in divergent and convergent thinking separately. It allows us to understand the effects of three communication methods in each way of thinking. The following measures adapted from Guilford's work [70] and that of Jack A. Goncalo's work [89] are proposed to evaluate the effectiveness of divergent and convergent thinking in the experiment.

Divergent Thinking

- **Fluency of Ideas:** The total number of labels excluding those judged to be off-topic, redundant, impossible, and/or useless.
- **Originality of Ideas:** The total number of labels that did not occur in other charts under the same problem generated by other teams.
- **Flexibility of Ideas:** The total number of viewpoints appearing in labels.
- **Elaboration of Ideas:** The average length of text (in characters) of labels.

The fluency of ideas, flexibility of ideas and originality of ideas are decided after the experiment by the majority vote of three human evaluators.

Convergent Thinking

- **Category Fluency:** The total number of categories (sub-categories are counted).
- **Within-Category Fluency:** The average amount of labels per category.
- **Average Depth of Top Categories:** The average depth of top categories.

The within-category fluency is proposed for solving some problematic cases. If titles of categories are defined too abstractly, the amount of labels in such categories tends to be high, and the number of categories in that chart tends to be small, they compensating breadth of ideas with greater detail. Such detailed categories should be considered as a product of quality convergent thinking. The within-category fluency indicates the strength of derived solutions (categories) in terms of the amount of support ideas (labels), which also reflects the quality of convergent thinking.

The average depth of categories is proposed. This is an average number of induction steps in deriving a solution from the collection of ideas under a top category in a KJ chart. This number reflects the depth of derived solutions, which is intuitively proportionate to the performance of convergent thinking.

7.3.3 Qualitative Evaluation

Participant Cross-Evaluation

In each problem, after three teams finish the convergent thinking stage, the participant cross-evaluation stage is performed. Instead of evaluating their own KJ chart, all members of a team are asked to evaluate two other charts created by the other two teams in the same session. All participants are guided to vote for their favorite labels on the aspect of their creativeness by announcing: "Please click on bulb icons of ideas that you do not expect to see". They are also guided to vote for their favorite categories on the aspect of their completeness by announcing: "Please click on bulb icons of categories that almost/completely contain all possible viewpoints". Note that a user cannot repeatedly vote on the same label or category. There are no lower and upper limits in the number of votes per user.

7.4 Experimental Result

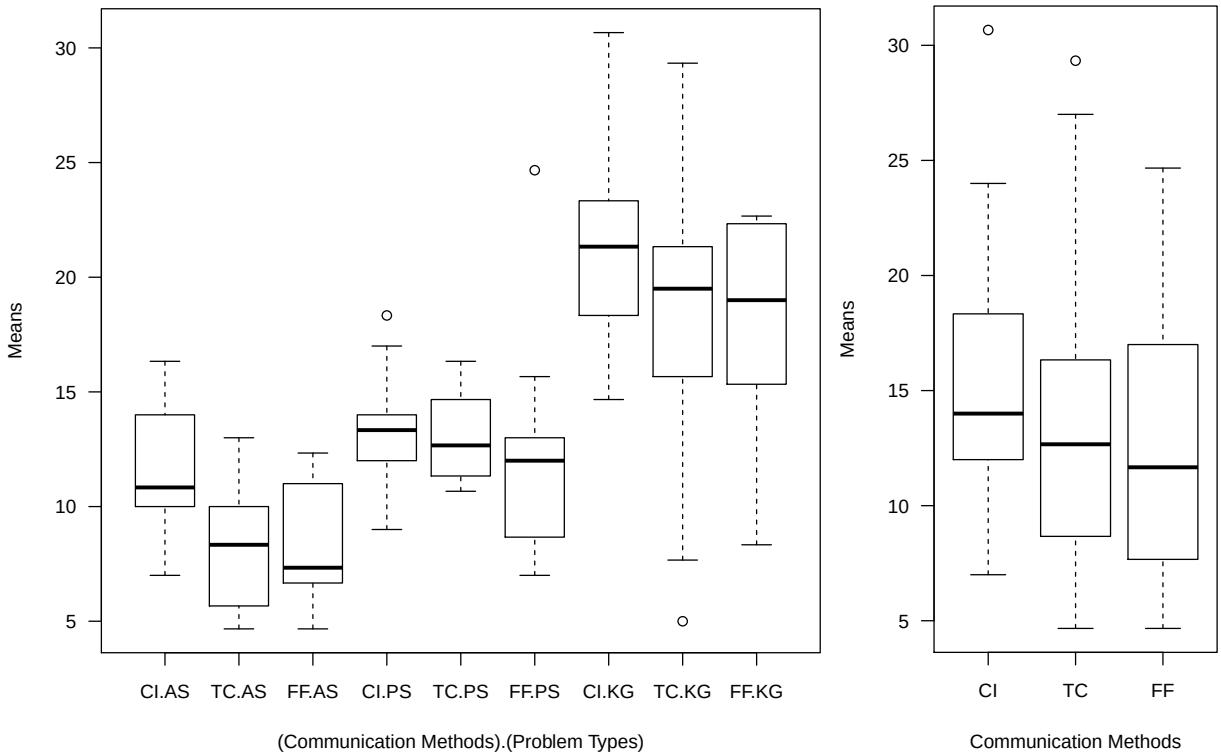
The data consist of 3,828 labels and 886 categories in 90 KJ charts constructed by 90 team members in 30 innovation teams. The settings in 90 KJ charts are yielded by crossing 30 problems from three problem types with three communication methods. To avoid the within-subject effects from the different creative performance of teams, we ask a team to construct three KJ charts by using three different communication methods in each problem type.

After the log file of the system and the recorded videos are examined, the average number of actions per team per chart is 193.2556 ± 60.8297 actions. The average number of action per user per chart is 64.4185 ± 29.3517. Note that only actions causing a change in charts are considered, such as, create, edit, move, resize, and remove. The average number of chat messages per chart is 31.5334. The average length of discussion dialogues per chart is roughly six minutes.

7.4.1 Quantitative Result

Figures 7.3, 7.4, 7.5, 7.6, 7.7, 7.8, and 7.9 show the descriptive statistics of the effectiveness of both divergent and convergent thinking in three communication methods (CI, TC, FF) in seven quantitative evaluators. Each figure consists of left and right subfigures. The left sub-figures present creativity performance under the influence of two social factors, namely, communication methods (CI,TC,FF) and problem types (AS, PS, KG). Since both factors have three possible categories, the number of possible categories in the crossing of both factors is nine. The right sub-figure presents creativity performance when considering only the communication method (CI, TC, FF). Note that the small circles represent the outliers of data.

The effectiveness of both divergent and convergent thinking was subjected to statistical testing by a two-way between-subject analysis of variance having three levels of team communication methods (CI, TC, FF) and three levels of problem types (AS, PS, KG). Since this chapter focuses on the effects of communication methods, the effects from problem types are not reported.

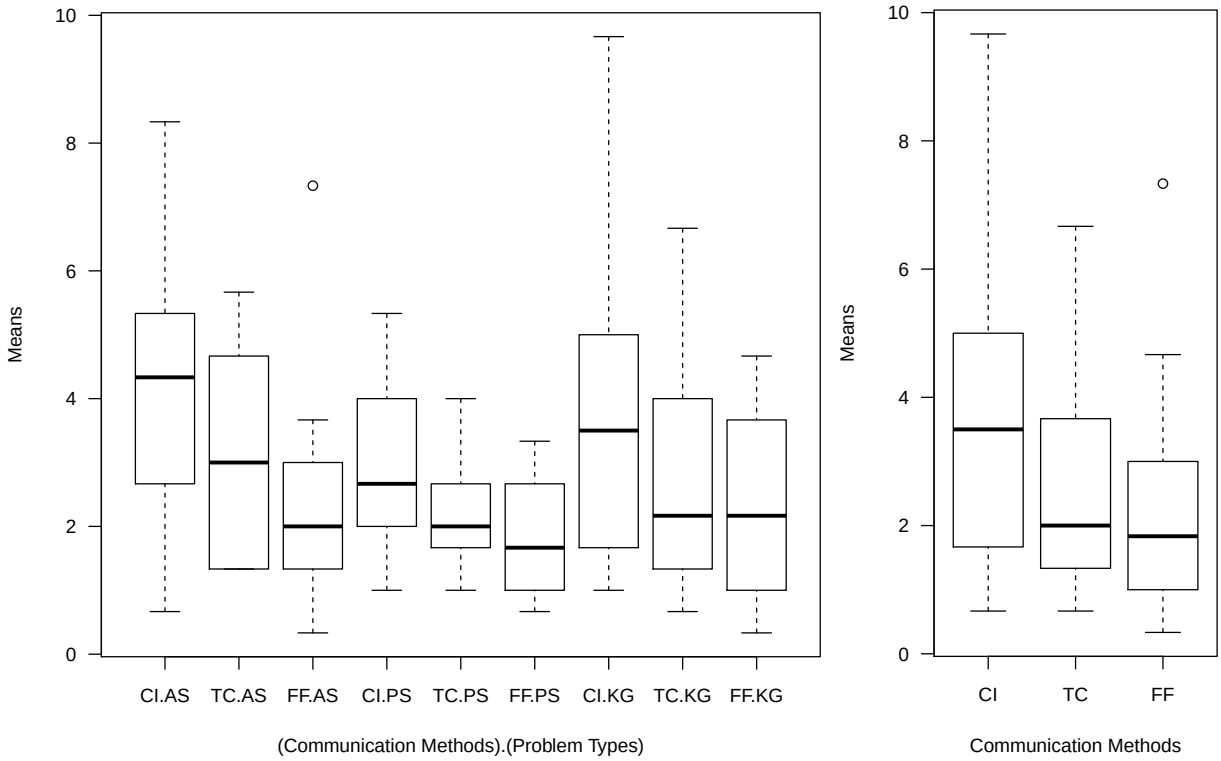


(a) Both Communication Methods and Problem Topics (b) Only Communication Methods

Figure 7.3: Quantitative result: fluency of ideas (per user per KJ-Chart)

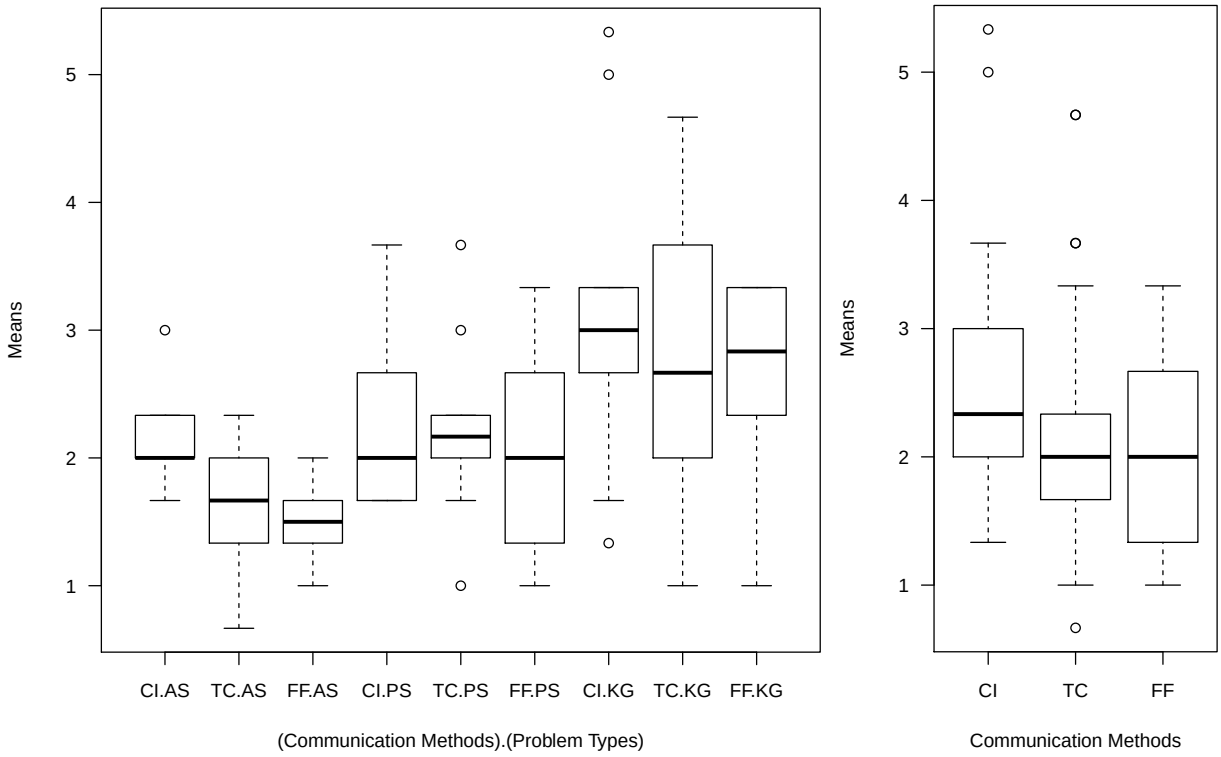
Divergent Thinking

The effects of communication methods were statistically significant to the fluency of ideas at 0.05 significant level, $F(2, 85) = 3.5659, p = 0.0325, \eta^2 = 0.0401$. A post hoc Tukey test showed that the fluency of ideas was significantly greater at $p = 0.04$ for CI (46.5 ± 16.4584) than for FF (38.4667 ± 17.6357). TC was not significantly different from the other methods lying somewhere in the middle near to CI. For the originality of ideas, the effects of communi-



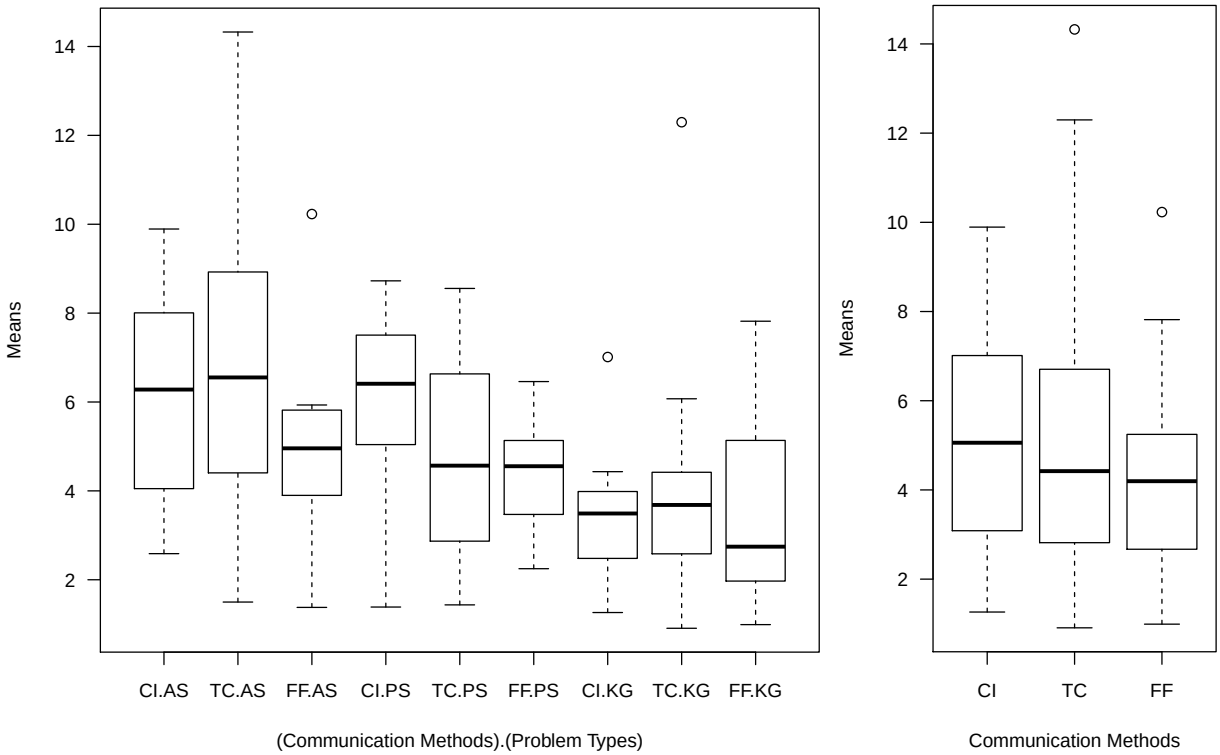
(a) Both Communication Methods and Problem Topics (b) Only Communication Methods

Figure 7.4: Quantitative result: original of ideas (per user per KJ-Chart)



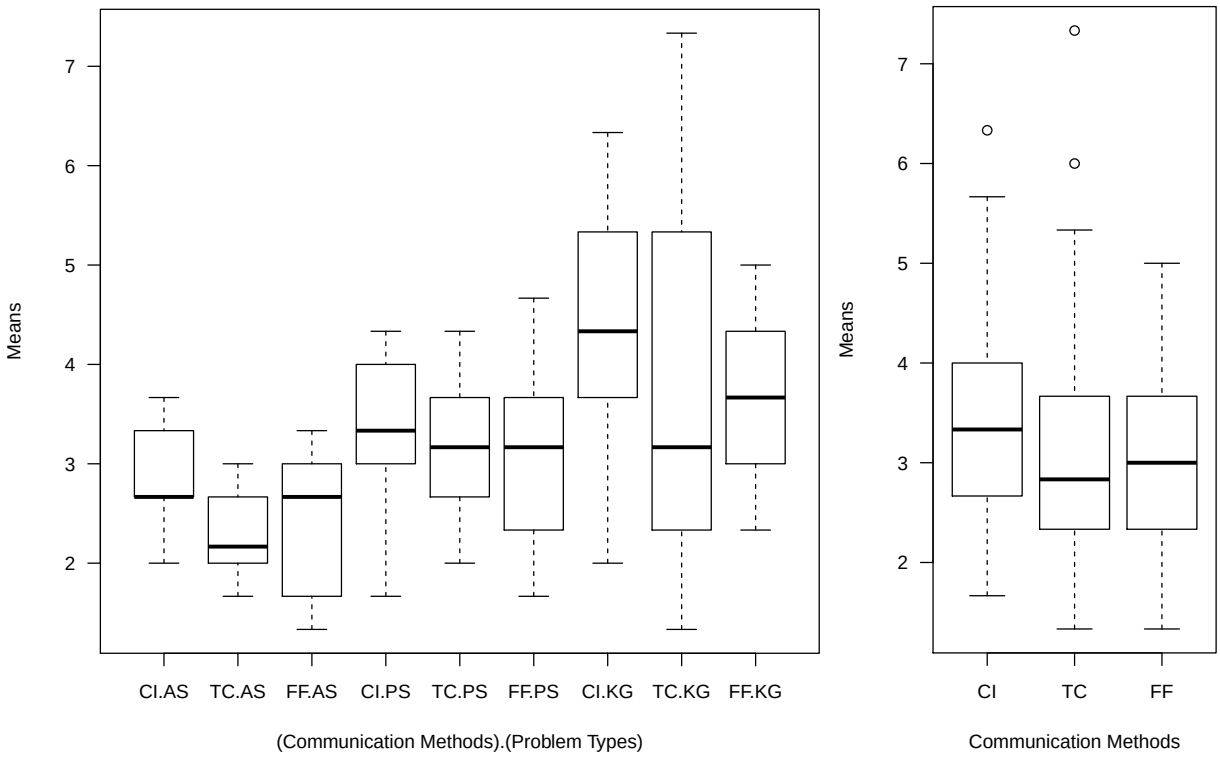
(a) Both Communication Methods and Problem Topics (b) Only Communication Methods

Figure 7.5: Quantitative result: flexibility of ideas (per user per KJ-Chart)



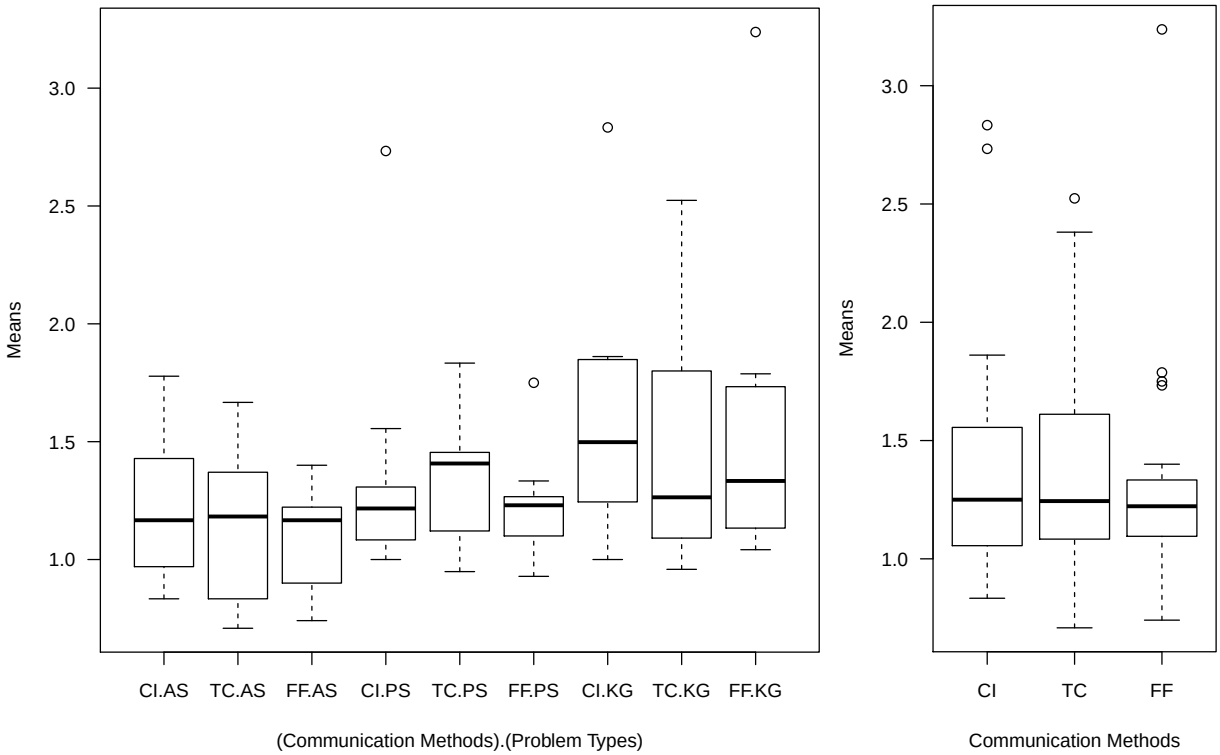
(a) Both Communication Methods and Problem Topics (b) Only Communication Methods

Figure 7.6: Quantitative result: elaboration of ideas (per user per KJ-Chart)



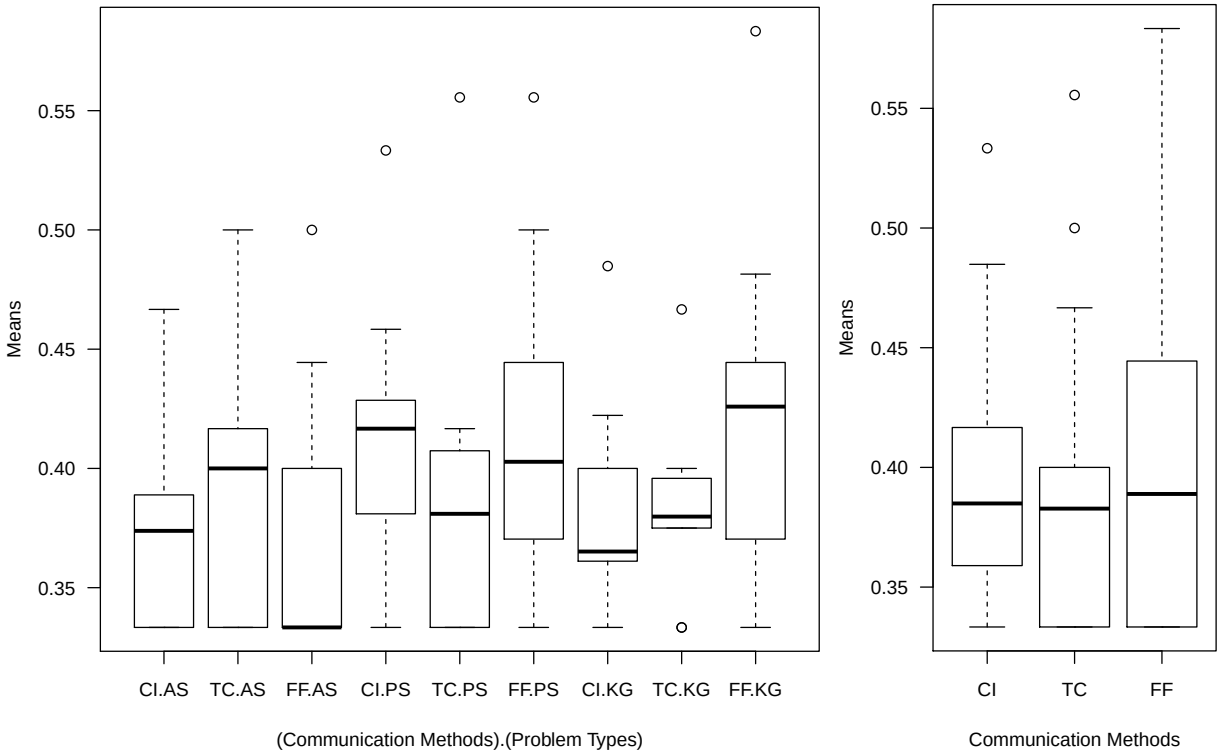
(a) Both Communication Methods and Problem Topics (b) Only Communication Methods

Figure 7.7: Quantitative result: category fluency (per user per KJ-Chart)



(a) Both Communication Methods and Problem Topics (b) Only Communication Methods

Figure 7.8: Quantitative result: within-category fluency (per user per KJ-Chart)



(a) Both Communication Methods and Problem Topics (b) Only Communication Methods

Figure 7.9: Quantitative result: average depth of top categories (per user per KJ-Chart)

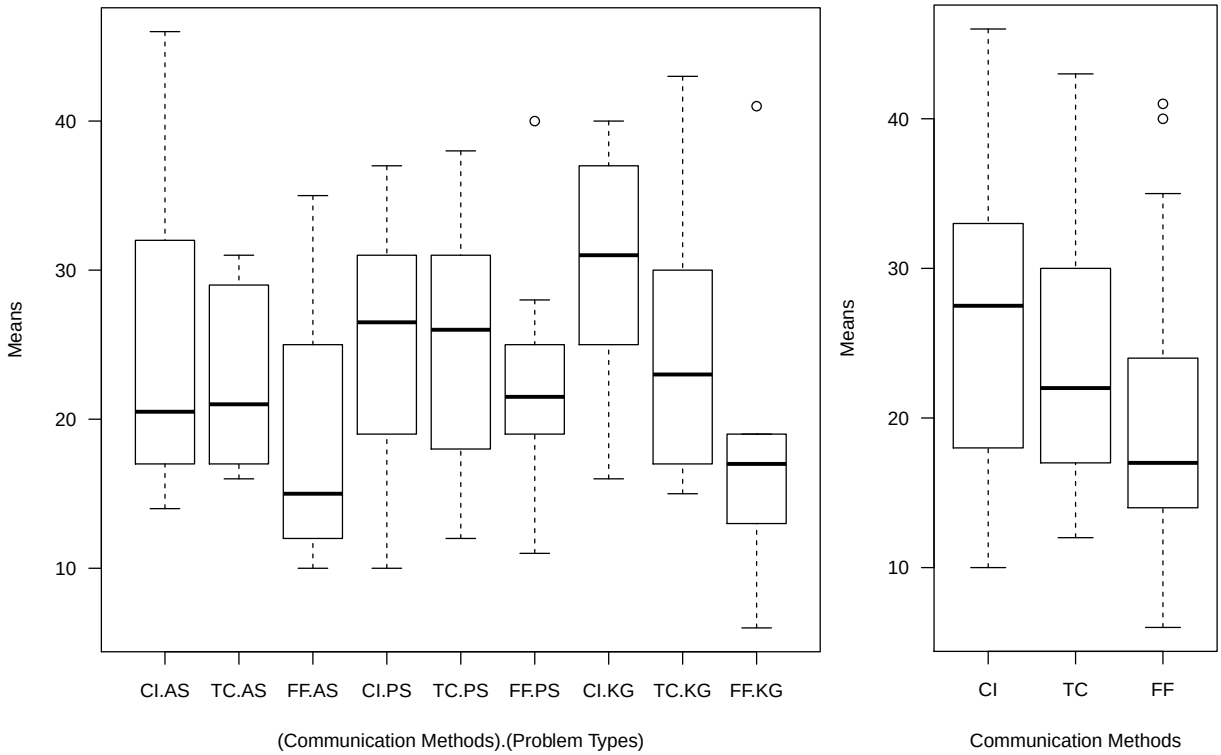
cation methods obtained significant variation ($p < 0.01$), $F(2, 85) = 5.6567, p = 0.0049, \eta^2 = 0.1119$. A post hoc analysis using the Tukey test indicated that the originality of ideas was significantly higher at $p = 0.004$ for CI (11.1 ± 6.4880) than for FF (6.6667 ± 4.7149). TC was almost significantly different ($p = 0.05417$) from FF and was very close to CI.

In the flexibility of ideas and the elaboration of ideas, the effects of communication methods were not statistically significant. The F ratio of the flexibility of ideas and the elaboration of ideas are $F(2, 85) = 2.2939, p = 0.1071, \eta^2 = 0.0379$, n.s., $p > 0.05$ and $F(2, 85) = 1.5057, p = 0.2277, \eta^2 = 0.0301$, n.s., $p > 0.05$ respectively.

Convergent Thinking

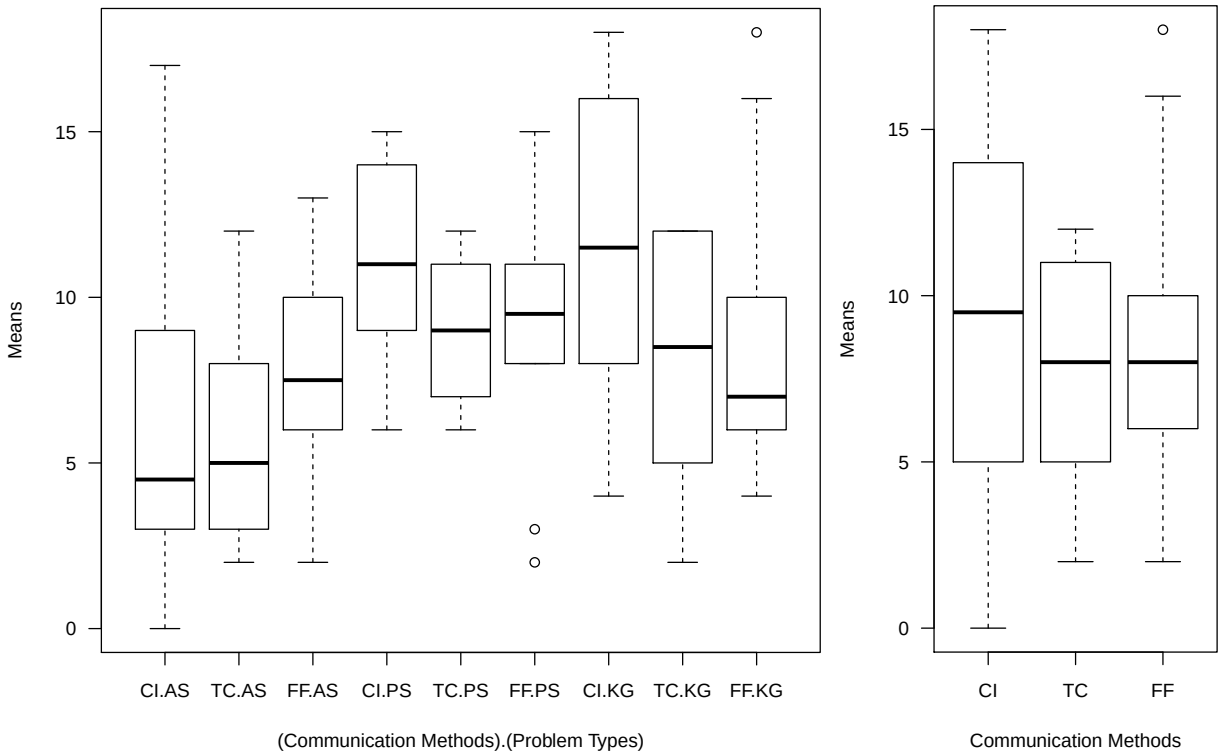
With all three convergent thinking performance measures, which are the category fluency, the within-category fluency, and the average depth of top categories, the effects of communication methods were not statistically significant in these measures. F ratios of the category fluency, the within-category fluency, and the average depth of category are $F(2, 85) = 1.7699, p = 0.1766, \eta^2 = 0.0299$, n.s., $p > 0.05$; $F(2, 85) = 0.3870, p = 0.6802, \eta^2 = 0.0078$, n.s., $p > 0.05$, and $F(2, 85) = 0.7540, p = 0.4736, \eta^2 = 0.0168$, n.s., $p > 0.05$ respectively.

7.4.2 Qualitative Result



(a) Both Communication Methods and Problem Topics (b) Only Communication Methods

Figure 7.10: Qualitative result: participant cross-evaluation in divergent thinking phase (per user per KJ-Chart)



(a) Both Communication Methods and Problem Topics (b) Only Communication Methods

Figure 7.11: Qualitative result: participant cross-evaluation in divergent thinking phase (per user per KJ-Chart)

Participant Cross-Evaluation

There are 3,026 votes on 1,721 labels and 856 votes on 579 categories in 90 KJ charts. Figures 7.10 and 7.11 show the descriptive statistics of the participant cross-evaluation results in three problem types (AS, PS, KG), and three communication methods (CI, TC, FF) in both ways of thinking. Note that the small circles represent the outliers of data. As in the quantitative result, the participant cross-evaluation result was subjected to a two-way between-subject analysis of variance. Only the effects of communication methods are focused on.

In divergent thinking, the effects of communication methods yielded significant variation to the number of favored ideas at 0.05 significant level, $F(2, 85) = 5.6438, p = 0.0049, \eta^2 = 0.1151$. A post hoc analysis using the Tukey test yielded that the number of favored ideas is significantly higher for CI (26.6 ± 8.7636) than for FF (19.5 ± 8.4924) at $p = 0.0037$. TC was not significantly different from the other methods lying somewhere in the middle.

In convergent thinking, the effects of communication methods were not statistically significant to the number of favored categories, $F(2, 85) = 1.4073, p = 0.2504, \eta^2 = 0.0283$,n.s., $p > 0.05$.

Questionnaire

The questionnaire about the effectiveness of three communication methods is answered by 90 participants. The questionnaire is illustrated in Figures 7.12. It consists of ten questions presented using 1 (Very Poor) - 5 (Very Good) Likert scale. For the first six questions, the effec-

tiveness of three communication methods (CI, TC, FF) in the divergent and convergent thinking phases is required for evaluation. The result is shown in Table 7.2.

Table 7.2: Qualitative results on the effectiveness of three communication methods.

	CI	**TC**	**FF**
Divergent Thinking	3.6333 ± 0.9416	3.3222 ± 0.7908	$\mathbf{3.7556 \pm 1.009}$
Convergent Thinking	2.6222 ± 0.9189	3.2333 ± 0.7039	$\mathbf{4.4111 \pm 0.6161}$

For the last four questions, answers to the following questions are required: "Which communication model that do you like/dislike most in collaborative creativity thinking and why?". The following are the survey's results (only most frequent reasons are displayed):

- Like
  - CI(27) No discussion allows self-creativity working to full potential.
  - TC(10) Allow discussion with the fewest interruptions.
  - FF(53) The easiest and fastest way to communicate and collaborate.
- Dislike
  - CI(48) Can not communicate/collaborate, too many overlapped ideas.
  - TC(35) Wasting time in typing text for chatting.
  - FF(7) Allow others to limit self-creativity and waste time in discussion.

7.5 Discussion

According to the experimental result, CI was more appropriate than others because no team communication, mutual understanding, and consensus building were needed. Since these activities consumed a moderate amount of time, CI provided more thinking time than other methods. In addition, since creativity requires people to think differently, the different ideas generated from individual thinking are contrary to getting and gaining consensus. CI bypassed this barrier by asking the individual to input his/her ideas into the system directly.

Comparing TC and FF, TC was more restrictive in communication than the FF; thus, TC provided more thinking time. Moreover, it is an asynchronous communication, which does not need to respond simultaneously, so it does not interrupt users while they are thinking.

7.6 Conclusion

We examined the effects and the effectiveness of three communication methods for a KJ-Method groupware. These are the chart interactions only, text chat, and face-to-face communication methods. Their effectiveness was measured by the degree of team creativity evaluated from the 90 constructed KJ charts and by the opinions of 90 participants.

According to the experimental result, the communication methods significantly affected the degree of team creativity on divergent thinking, but not on convergent thinking. The chart interaction only method yielded a higher degree of fluency and originality of ideas than the face-to-face method. The chart interaction only method also generated a higher number of favored ideas than the face-to-face method. Even though the face-to-face communication method

Post-session questionaire

This is the post-session questionaires about the effectiveness of each communication method.
* Required

Your full name: *
For example, Joe Lin.

Divergent Thinking

Effective of each communication model in Divergent Thinking *
In perspective of group creativity collaboration

	Very poor	Poor	Fair	Good	Very Good
"KJ chart interaction only" communication model	○	○	○	○	○
"KJ chart interaction only + Text Chat" communication model	○	○	○	○	○
"KJ chart interaction only + Face2Face" communication model	○	○	○	○	○

Convergent Thinking

Effective of each communication model in Convergent Thinking *
In perspective of group creativity collaboration

	Very poor	Poor	Fair	Good	Very Good
"KJ chart interaction only" communication model	○	○	○	○	○
"KJ chart interaction only + Text Chat" communication model	○	○	○	○	○
"KJ chart interaction only + Face2Face" communication model	○	○	○	○	○

Favoritism

Which communication model that you LIKE most for collaborative creativity thinking? *
KJ chart interaction only

Why? *
Explain its good points

Which communication model that you DISLIKE most for collaborative creativity thinking? *
KJ chart interaction only

Why? *
Explain its bad points

Submit

Powered by Google Docs

Report Abuse - Terms of Service - Additional Terms

Figure 7.12: Questionnaire about the effectiveness of team communication method

yielded the worst degree of team creativity, it is most favored by participants due to it providing the easiest way to communicate.

Further research may attempt to analyze the relationship between group creativity and closeness among team members and/or the level of background knowledge in a given topic. Finally, we can better understand the relationship between team communication and creativity.

Chapter 8

Team Characteristics

8.1 Introduction

Every innovation company, regardless of its size is regularly under pressure to create products or service offerings. Team creativity is the key resource driving the development processes to achieve successful products. The effective techniques for stimulating team creativity are greatly desired by many organizations.

With the recent developments of information and communication technologies, Creativity support groupware is developed to a level that can support an innovation team to communicate, discuss, and perform group creative techniques over geographical distances, and different time availabilities. Creativity support groupware helps an innovation team to yield creative and well-established solutions for accomplishing team goals. Creativity support groupware is increasingly important for competitive creative economies. Many research studies discovered factors that significantly affect team creativity through groupware use, e.g., the team size [24], degree of idea exposure [87], communication patterns [88], and the teams past successful experiences [89]. A research study proposed a proper way to measure such effects, such as the IPL (Island formation for Pseudo Label) for measuring the effect of individual knowledge [91].

Although there are many studies in creativity support groupware, little is known about the impact of team characteristics on team creativity. Traits of a team, e.g., the background knowledge, background knowledge difference, friendship closeness, and the difference in closeness of the friendship among team members, are proven to significantly affect team creativity in a traditional working environment [92, 93, 94], but not yet in a creativity support groupware.

This report studies the effects of team characteristics on the team creativity while performing group creativity tasks via a KJ Method groupware. The KJ Method, which is an excellent individual/group creativity technique, is applied to our creativity support groupware due to its popularity especially in Japan. The effects of these team characteristics are measured on the basis of the creativity of the teams proposed ideas (KJ Chart). The obtained results are examined for correlation analysis, regression analysis, and statistical significance.

This study is aimed at helping companies that use creativity support groupware to make a decision on assigning individual members to a team for achieving higher team creativity. This report is organized as follows: Section 8.2 explains four team characteristics that impact on team creativity; Section 8.3 describes two control variables that affect team creativity; 8.4 shows the experimental settings and how to evaluate team creativity and characteristics; Section 8.5 gives the experimental results; Section 8.6 concludes this chapter.

8.2 Team Characteristics

Based on the previous studies, team characteristics that were proven to significantly affect team creativity are investigated as follows:

8.2.1 Background Knowledge

Background knowledge was proven to be a positive significant factor in the degree of individual creativity [95, 96] and team creativity [92]. Background knowledge was also an essential key for successful creative work, e.g., musical knowledge in Jazz performances [97] and art knowledge in design work [98]. The impact of background knowledge to team creativity was reflected through the ability of learning new knowledge [99]. Creative thinkers who have high background knowledge, tend to consider problems in holistic ways (looking for an overall board scope and then moving into detail) [98]. They understand a problem as a whole and develop new ideas more fluently and more flexibly. A group of physics professors is more likely to have higher team creativity on the explanation of a black hole than a group of children.

8.2.2 Difference of Background Knowledge

The difference of background knowledge among team members was studied as a positive factor to team creativity [93]. A difference in high background knowledge forces the members who have low background knowledge to adapt knowledge from different fields to produce ideas to a given problem. They usually come up with novel ideas, which contribute to team creativity. These ideas will be well refined later by the members who have high background knowledge. The team that consists of the professors and the kindergarten students is more likely to have higher team creativity than the team with professors only.

8.2.3 Friendship Closeness

New perspectives or novel ideas are less likely to be heard from close friends, but more likely from strangers. The friendship closeness among team members was discovered to be a negative factor to team creativity because people tend to be friends with the individuals similar to themselves [94]. The length of time they have known each other, their knowledge, and attitudes are converged over time. A group of strangers is more likely to have higher team creativity than the group of family members.

8.2.4 Difference of Friendship Closeness

The high degree of the difference of friendship closeness in a team splits a team into sub-groups; for example, a team consisting of a married couple and a stranger. Sub-groups tend to create their local orientation, intra-language, and communication encoding/decoding rules, which makes overall team communication less efficient. These are negative factors to team creativity [88]. The high difference in friendship closeness also tends to disunite a team, which discourages team creativity.

8.3 Control Variables

The following control variables are considered to have a significant impact on team creativity while performing the group KJ Method via a KJ Method support groupware.

8.3.1 Team Communication Methods

The following three communication methods are commonly found in most groupwares.

1. **Chart Interaction Only (CI):** This method allows members to communicate via actions in a shared KJ chart only. Every action performed by a member is propagated to all other members in the same team. Members see all changes happening to all label categories in real time. Note that the name of a member who performs an action is not shown.

2. **Chart Interaction + Text Chat (TC):** This method allows members to communicate via chart interaction and online group chat. Once a message is received, the system beeps to notify a user. Note that only text can be transmitted.

3. **Chart Interaction + Face-to-Face (FF):** This method allows members to communicate via chart interaction and face-to-face communication. Members met in person while using the system and discussing verbally with each other. Note that, other communication channels are not allowed (such as pointing at another members screen or drawing a figure on a whiteboard).

8.3.2 Problem types

The following three problem types, commonly found in group creativity tasks, are given in the experiments.

1. **Unusual Used (UU):** "Apart from being auto parts, write down any other uses of a wheel".
2. **Problem Solving (PS):** "How can we solve political corruptions?".
3. **Knowledge Gathering on a Topic (KG):** Members contribute their knowledge on a given topic, for example, "Fast foods".

8.4 Experimental Settings

8.4.1 Experimental Procedure

The Gugeek, a KJ-Method support groupware, is used for the experiment. The total number of participants for the experiment is 90. Participants consist of master and doctoral students who study at the School of Knowledge Science and School of Information Science of Japan Advanced Institute of Science and Technology. All participants must have English testing scores equivalent to a TOEFL score of 550 or more.

The experiments consisted of ten sessions. In each session, nine participants are divided into three teams of three participants each. All members in a team were seated at the same table and were asked to access our KJ Method groupware via their own portable computers (one laptop per participant). Before the experiments, every participant is informed of the following:

Table 8.1: Communication methods and problem type assignment

Team	Problem 1	Problem 2	Problem 3
1	CI	TC	FF
2	TC	FF	CI
3	FF	CI	TC

1. The purposes, procedures, and rules of the KJ Method.
2. Groupware instructions.
3. Examples, and Q&A.

Every team performed the group KJ Method on the same set of three problems (one problem per problem type). The experiments use three team communication methods as described in Section 8.3.1. To equally distribute control variables and avoid the effect of tool experience, both problem types and team communication methods were assigned to teams in a different orders as shown in Table 8.1.

For evaluation purposes, ten-minute and five-minute time-out sessions were set for the divergent and convergent thinking stages respectively. Once the time-out is reached, the current stage is automatically forwarded to the next. In the experiment, the votes in the evaluation stage are disregarded since a participant can vote for his/her ideas with bias.

At the end of the session, a questionnaire was shown on their computer screen. The questionnaire consists of two questions, using the 1-5 Likert scale (1=*Very Poor* to 5=*Very Good*). Every participant gives the degree of his/her background knowledge on three given problems and provides the degree of friendship closeness between himself/herself and the rest of the team.

8.4.2 Team Creativity Evaluation

To understand the effects of team characteristics on the team creativity in each stage of thinking, we evaluate the degree of creativity in divergent and convergent thinking separately. The following measures adapted from Guilford's work [70] are used to evaluate the performance of both ways of thinking.

Divergent Thinking

- **Number of Ideas:** The total number of labels.
- **Fluency of Ideas:** The total number of labels excluding those judged to be off-topic, redundant, impossible, and/or useless.
- **Originality of Ideas:** The total number of labels that did not occur in other charts under the same problem generated by other teams.
- **Elaboration of Ideas:** The average length of text (in characters) on labels.

Convergent Thinking

- **Flexibility of Ideas:** The total number of categories (sub-categories are counted).

- **Within-Category Fluency:** The average amount of labels per category.
- **Depth of Top Categories:** The average depth of top categories.

The fluency and the originality of ideas is decided after experiments by the majority vote of three human evaluators.

8.4.3 Team Characteristic Evaluation

We derived the traits of a team by asking individual members to complete a questionnaire. The questionnaire is illustrated in Figures 8.1.

Let's k_i be a background knowledge in a given problem of members i of a team, $f_{i,j}$ be the degrees of friendship closeness between member i and member j told by member i, $f_{i,i} = 0$, and n is the amount of members in a team where $k, f, n, i, j \in I^+$ and $n > 1$. Each variable in team characteristics (see Section 8.2) is computed as follows:

Average Background Knowledge (BK) is the average of all elements in K.

Average Difference of Background Knowledge (DBK) is the average of the background knowledge difference in all member pairs of a team.

$$\mathrm{DBK} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} \mid k_i - k_j \mid}{n^2 - n} \tag{8.1}$$

Average Friendship Closeness (FC) is the average of mutual friendship closeness of two members of a team.

$$\mathrm{FC} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} \mid f_{i,j} \mid}{n^2 - n} \tag{8.2}$$

Average Difference of Friendship Closeness (DFC) is the average of the friendship closeness difference of two member pairs selected from all member pairs of a team.

$$m(i,j) = \frac{f_{i,j} + f_{j,i}}{2} \tag{8.3}$$

$$M = \{m(i,j) | 1 \leq i \leq n-1, i+1 \leq j \leq n\} \tag{8.4}$$

$$\mathrm{DFC} = \frac{\sum_{i=1}^{\frac{n^2-n}{2}}\sum_{j=i+1}^{\frac{n^2-n}{2}} \mid M_i - M_j \mid}{\frac{(n^2-n)(n^2-n-2)}{8}} \tag{8.5}$$

8.4.4 Control Variables Evaluation

Since the control variables (see Section 8.3) contain categorical data, they converted to numeric data before performing the correlation analysis.

Post-session questionnaire

This is the post-session questionnaires for determining individual and team characteristics while using Gugeek KJ-Method support groupware.
* Required

Your full name: *
For example, Joe Lin.

Friendship Closeness

Input the name of the first team member of your team

How much is your friendship closeness with the first team member?

	1	2	3	4	5	
Unknown	○	○	○	○	○	Best Friend

Input the name of the second team member of your team.

How much is your friendship closeness with the second team member?

	1	2	3	4	5	
Unknown	○	○	○	○	○	Best Friend

Background Knowledge

How much do you have background knowledge about the first topic problem? *
Imagine to the usage of a brick other than using in construction.

	1	2	3	4	5	
Novice	○	○	○	○	○	Expert

How much do you have background knowledge about the second topic problem? *
If you are a statesman, How could you solve the poverty problem?

	1	2	3	4	5	
Novice	○	○	○	○	○	Expert

How much do you have background knowledge about the third topic problem? *
Social Networking

	1	2	3	4	5	
Novice	○	○	○	○	○	Expert

Submit

Powered by Google Docs

Report Abuse - Terms of Service - Additional Terms

Figure 8.1: Questionnaire for deriving team characteristics

Table 8.2: Descriptive statistics of all variables in this study.

Variable	Minimum	Maximum	Mean	SD	N
Dependent Variables					
Divergent Thinking					
Number of Ideas	14.000	94.000	42.530	18.080	90
Fluency of ideas	14.000	92.000	41.540	17.750	90
Originality of ideas	1.000	29.000	8.556	5.617	90
Elaboration of ideas	2.724	42.978	14.624	7.780	90
Convergent Thinking					
Flexibility of ideas	4.000	22.000	9.644	3.413	90
With-in category fluency	2.125	9.714	4.025	1.337	90
Depth of top categories	1.000	1.750	1.180	0.175	90
Team Characteristics					
Background knowledge	1.667	5.000	3.263	0.725	90
Difference of background knowledge	0.000	2.667	0.881	0.615	90
Friendship closeness	2.500	3.833	3.200	0.316	90
Difference of friendship closeness	0.000	2.333	0.633	0.601	90
Control Variables					
Communication methods	1.000	3.000	2.000	0.821	90
Problem types	1.000	3.000	2.000	0.821	90

Team Communication Method (TCM)

The communication method variable is assigned by the following mapping, CI = 1, TC = 2, and FF = 3. These are in the order from very restricted to unrestricted in communication.

Problem Type (PT)

The problem type variable is assigned by the following mapping, UU = 1, PS = 2, and KG = 3. These are in the order from a small amount of possible solutions (a hard problem) to a high amount of possible solutions (an easy problem).

8.5 Experimental Result

The data consisted of 3,828 labels and 886 categories in 90 KJ charts in 30 predefined problems constructed by 90 team members in 30 innovation teams. All possible values of two control variables are equally distributed by the Cartesian product of three problem types and three communication methods in 90 KJ charts.

Table 8.2 shows the descriptive statistics of all variables in this study including creativity measures, team characteristics, and two control variables.

Table 8.3: Bivariate correlations and significant test of four divergent thinking measures, three convergent thinking measures, four team characteristics, and two control variables.

Variables	1	2	3	4	5	6	7	8	9	10	11	12	13
Number of ideas	–												
Fluency of ideas	0.985*	–											
Originality of ideas	0.456*	0.416*	–										
Elaboration of ideas	−0.386*	−0.399*	0.042	–									
Flexibility of ideas	0.736*	0.753*	0.447*	−0.301*	–								
Within category fluency	0.356*	0.381*	−0.114	−0.268*	−0.212*	–							
Depth of top categories	0.063	0.045	−0.103	−0.112	0.175*	−0.200*	–						
Background knowledge	0.474*	0.495*	0.120	−0.419*	0.362*	0.286*	0.096	–					
Diff. of background knowledge	0.013	0.044	−0.107	−0.076	−0.024	0.154	0.080	−0.022	–				
Friendship closeness	0.021	0.007	0.074	0.162	0.105	−0.143	−0.012	−0.090	−0.089	–			
Diff. of friendship closeness	−0.007	−0.008	−0.186*	−0.420*	−0.130	0.220*	0.130	0.120	0.256*	−0.132	–		
Communication methods	−0.207*	−0.186*	−0.324*	−0.140	−0.160	−0.088	0.099	0.000	−0.030	0.000	0.000	–	
Problem types	0.643*	0.686*	−0.061	−0.345*	0.501*	0.359*	0.128	0.459*	0.030	0.000	0.000	0.000	–

* Significance at 0.1

Table 8.3 shows the Pearson product-moment correlation coefficient among divergent thinking and convergent thinking performances, team characteristics (Section 8.4.3), and two added control factors (Section 8.4.4). The numbers in Table 8.3 are the bivariate correlations between two variables. If the absolute value of numbers is greater than 0.2 (yielded by rough approximation), it is considered as statistically significant at the 0.1 level.

According to the correlation coefficients, the average background knowledge had a significant positive relationship with two divergent thinking measures and two convergent thinking measures, i.e., the number of ideas, fluency of ideas, flexibility of ideas, and with-in category fluency. Since there was a significant negative relationship between the average background knowledge and the elaboration of ideas, a team who had the higher background knowledge giving the shorter description in each proposed idea, but instead focusing more on the number of ideas.

The average difference of background knowledge and the average friendship closeness of team members had no significant relationship with all creativity measures. The average difference of friendship closeness yielded significant negative relations to the originality and elaboration of ideas, but obtained a significant positive relationship within category fluency. Subgroups discouraged the uniqueness and verbosity of generated ideas, and provided too many ideas for supporting a category, which might be due to the lack of team communication.

Linear least-square regression, which searches for best linear models to fit all social factors, was applied. Linear models yielded from the regression can be used for predicting scores of team creativity performance yielded from seven creativity measures from six social factors. The interactions among variables were assumed trivial. The regression analysis describing the team creativity from the control variables only (base model) is shown in Table 8.4. The regression analysis mapping from the control variables and the team characteristics to the team creativity (full model) is shown in Table 8.5. Note that, each number in the table is an estimated coefficient followed by standard error of each variables. Any coefficients of social factors in linear models that have an absolute value higher than significant thresholds, which are different for each creativity measure, are considered statistically significant in influencing the score of each creativity measure. The statistical significant levels are set to 0.1 for both tables.

According to the result of regression analysis shown in both tables, the average background knowledge of team members had a positive significant relationship with three divergent thinking measures, i.e., the number of ideas, fluency of ideas, and originality of ideas. In contrast, the background knowledge had a negative significant relationship with the elaboration of ideas. In convergent thinking, the background knowledge significantly related to the flexibility of ideas in a positive way. If a team had higher background knowledge, a higher amount of ideas, unique ideas and viewpoints were obtained.

The average difference of friendship closeness had negative relationships with the originality of ideas and the elaboration of ideas in the divergent thinking. If team members have a smaller gap in friendship closeness, the uniqueness and the verbosity of ideas is increased.

For the control variables, the method of communication significantly related to the divergent and convergent thinking measures in a negative way. Providing easier ways to communicate among team members significantly reduced team creativity performance. In addition, the types of given problems yielded significant relations with most of the creativity measures. In giving an easier problem to a team, the fluency and flexibility of ideas increases, but the originality and verbosity decreases.

Including the team characteristics into the regression analysis in the full model explained the overall variance of team creativity better than the base model by three percent. Moreover,

the variance of the elaboration of ideas described by the full model increased from the base model by 11%.

8.6 Conclusion

We studied the effects of team characteristics, i.e., the background knowledge, difference of background knowledge, friendship closeness, and the difference of friendship closeness among team members, on the team creativity through creativity support groupware use. The team creativity can be evaluated through the creativeness of their proposed ideas (KJ charts) measured by the seven creativity measures in both divergent and convergent thinking.

According to the experimental result, the higher background knowledge significantly helped team creativity in terms of fluency, uniqueness, flexibility, and conciseness of the proposed ideas. The difference in background knowledge and friendship closeness did not significantly influence team creativity. The difference in friendship closeness significantly reduced the uniqueness and verbosity of proposed ideas. Finally, the team characteristics that yield the highest team creativity are: having a high average background knowledge in the problem under consideration; and having the same degree of friendship closeness regardless of the level.

Table 8.4: Regression analysis on the divergent and convergent thinking performances: base model

Variables	Constant	Communication methods	Problem types	Adjusted R^2
Number of ideas	$23.367 \pm 5.127^*$	$-4.567 \pm 1.741^*$	$14.15 \pm 1.741^*$	0.443
Fluency of ideas	$19.911 \pm 4.799^*$	$-4.017 \pm 1.63^*$	$14.833 \pm 1.63^*$	0.494
Originality of ideas	$13.822 \pm 2.039^*$	$-2.217 \pm 0.692^*$	-0.417 ± 0.692	0.088
Elaboration of ideas	$23.809 \pm 2.776^*$	-1.325 ± 0.943	$-3.267 \pm 0.943^*$	0.119
Flexibility of ideas	$6.811 \pm 1.116^*$	$-0.667 \pm 0.379^*$	$2.083 \pm 0.379^*$	0.260
Within category fluency	$3.145 \pm 0.478^*$	-0.144 ± 0.162	$0.585 \pm 0.162^*$	0.117
Depth of top categories	$1.084 \pm 0.067^*$	0.021 ± 0.023	0.027 ± 0.023	0.004

* Significance at 0.1

Table 8.5: Regression analysis on the divergent and convergent thinking performances: including four team characteristic factors.

Variables	Constant	BK	DBK	FC	DFC	TCM	PT	Adjusted R^2
Number of Ideas	2.235 ± 17.027	$5.909 \pm 2.217^*$	0.24 ± 2.378	2.206 ± 4.506	-0.987 ± 2.454	$-4.561 \pm 1.711^*$	$11.748 \pm 1.935^*$	0.463
Fluency of ideas	0.661 ± 15.857	$5.847 \pm 2.064^*$	1.159 ± 2.214	1.469 ± 4.196	-1.278 ± 2.285	$-3.991 \pm 1.594^*$	$12.436 \pm 1.802^*$	0.517
Originality of ideas	7.517 ± 6.738	$1.719 \pm 0.877^*$	-0.476 ± 0.941	1.138 ± 1.783	$-1.784 \pm 0.971^*$	$-2.227 \pm 0.677^*$	-1.103 ± 0.766	0.129
Elaboration of ideas	$26.357 \pm 8.153^*$	$-2.79 \pm 1.061^*$	0.342 ± 1.138	2.229 ± 2.157	$-4.959 \pm 1.175^*$	-1.318 ± 0.819	$-2.143 \pm 0.926^*$	0.335
Flexibility of ideas	1.325 ± 3.711	$0.948 \pm 0.483^*$	0.052 ± 0.518	1.134 ± 0.982	-0.809 ± 0.535	$-0.666 \pm 0.373^*$	$1.698 \pm 0.422^*$	0.284
Within category fluency	$3.556 \pm 1.576^*$	0.222 ± 0.205	0.202 ± 0.22	-0.428 ± 0.417	0.374 ± 0.227	-0.14 ± 0.158	$0.49 \pm 0.179^*$	0.159
Depth of top categories	$1.01 \pm 0.228^*$	0.008 ± 0.03	0.015 ± 0.032	0.006 ± 0.06	0.033 ± 0.033	0.021 ± 0.023	0.024 ± 0.026	-0.023

* Significance at 0.1

Part IV

Conclusion

Chapter 9

Summary and Conclusion

9.1 Summary

The KJ Method is a popular data organization method invented by JIRO Kawakita in 1951 and is still popular, especially in Japan, until the present day. It is a method for organizing vague information into understandable knowledge, which helps the practitioner to unbiasedly discover the truth from the observed information. The KJ Method consists of four basic steps, namely: (1) Label Making; (2) Label Grouping; (3) Chart Making; and (4) Explanation. The KJ Method has been proven to be an excellent tool for organizing and prioritizing ideas and data by many companies and organizations in the past. Performing the KJ Method seems to need few requirements, but people can be prevented from using the KJ Method due to its requirements and inconvenience of performance. To eliminate or reduce the requirements such as gathering participants to the same time and place, wasting stationery products, and the inconvenience in distributing ideas after a meeting, the KJ-Method support groupware is proposed to provide a convenient way to perform the group KJ Method.

In this dissertation, we proposed a novel KJ-Method support groupware called Gugeek based on a prototype divergent thinking support system called Eureka!. The Gugeek is an ambitious attempt to provide an accessible platform and idea processing support for the group KJ- Method. The Gugeek allows a group of people to perform the group KJ-Method together via the Web even though they have different time availabilities and are in different locations. The Gugeek is exciting and important because it promises to provide a significantly more effective way of working compared to the traditional face-to-face meeting. The Gugeek significantly reduces operating costs including transport , time wasted in transportation, time wasted for waiting for other participants, and so on. The Gugeek significantly boosts creativity of output ideas via the variety of creativity assistants.

The Gugeek is a Web application developed from JavaScript and HTML5 programming languages, which is the latest Internet technology of the present day. The Gugeek can be easily accessed by most devices including mobile and personal computers without any pre-installed software. The only requirement for using this software is an Internet-connected device that has an Internet browser installed.

9.1.1 Creativity Assistants

The Gugeek is a well-designed platform, fitting well with participants needs for performing remote group KJ-Method activity. The objective of this research is to discover and invent an

efficient KJ-Method groupware, to support a group of users to maximize their individual and team creativity performance. By the way, depending solely on the Gugeek groupware architecture, the Gugeek level of support is just at the framework-paradigm level (according to Youngs classification [5]) that store, display, and distribute the input ideas and guide the user to think according to the procedure of the group KJ-Method. To increase the Gugeek groupware capability, authors may wish to pursue the next level of support for Gugeek, which is the generative support level. This can be done by adding Gugeek to the creativity assistant modules, which automatically suggest appropriate associative information for promoting creativity of team members.

We developed creativity assistant modules in both divergent and convergent thinking. For the divergent thinking module, four divergent thinking support engines using associative information extracted from Wikipedia are proposed. The four divergent thinking support engines are: (1) GETAs most related Documents (GD); (2) GETAs most related Keywords (GK); (3) Wikipedias Freelinks (WF); and (4) Wikipedias Category (WC). The first two engines adapt the association search engine GETA [64], and the last two engines find the association by using the document structure. Their quality is compared by experiments in both quantitative and qualitative evaluation using the Eureka! divergent thinking support system. According to the quantitative experimental result, GK, a summary of the users input sentences, yields the highest number of ideas, highest fluency of ideas, and highest originality of ideas, while WF yields the highest flexibility of ideas. In the qualitative evaluation, the GK also yields the highest satisfaction to users.

For the convergent thinking module, we focus on automatic category naming (or automatic topic identification) capabilities for supporting category naming in the label grouping step of the KJ-Method. The automatic topic identification capabilities can be used in different creativity techniques other than the KJ-Method, such as mind mapping. We proposed a knowledge-based method for automatically generating a short piece of English text about a topic to each idea category in the idea charts. The proposed automatic topic identification used Yet Another General Ontology (YAGO), and WordNet as its knowledge bases. A novel topic selection method, Harmonic Mean (HM), is proposed, and we compared its performance with three existing methods, namely: (1) Branch Ratio (BR); (2) Concept Counting (CC); and (3) Ratio Balance (RB). The comparisons have been made by using two experimental datasets constructed from the KJ-Method support system (Gugeek) and the mind mapping software (XMind), which contain the set of idea categories and their idea members and the correct topic (title) of that category annotated by humans. The human ranking on a predicted topic, direct string comparison (suggestion accuracy), the word similarity between the predicted topic and the actual topic are computed and used as evaluators for measuring the performance of the proposed topic selection method, Harmonic Mean (HM) and its variance, and compared against the other three existing topic selection methods. Both noun and verb topics are examined.

According to the empirical evaluation in the Gugeek KJ-Method groupware, participants ranked $HM_{0.5}$ as the highest rank for the noun topic and CC as the highest rank for the verb topic. For suggestion accuracy, HM_1 is the most accurate selector for the noun topic at 43.658% accuracy, whereas CC is the most accurate selector for the verb topic at 49.062% accuracy. For average word similarity between the predicted topic and the actual topic annotated by humans, HM_1 yielded the most similar score for the noun topic, but CC yielded the most similar score for the verb topic. According to the empirical evaluation in the Xmind software, CC yielded the highest accuracy in direct comparison of both noun and verb topics at around 8% and 3% respectively. For average word similarity, the HM_0 and $HM_{0.5}$ yielded the highest word similarity score in noun and verb topics respectively.

9.1.2 Social Influences

Although the creativity assistant modules can improve team creativity performance dramatically, another factor that has affected team creativity during the use of the KJ-Method groupware is social influence among team members. If the relationships between such social influences and team creativity can be discovered, we can fix or reshape the creativity support groupware to avoid unfavorable social influences among participants that depress team creativity. We focused on two groupwares design keys, which directly affect social interaction and influence among members, which are: (1) the team communication method; and (2) team characteristics.

Since intensive discussion among members is a crucial activity in idea creation, communication methods in creativity support groupware have a significant influence on team creativity. Three team communication methods are proposed as follows: (1) Chart Interaction only (CI); (2) Chart Interaction + Text Chat (TC); (3) Chart Interaction + Face-to-Face (FF). They are sorted from hard to easy communication. Three communication methods are compared by both quantitative and qualitative evaluation. For quantitative evaluation, the result is based on the creativity performance on 90 KJ charts constructed by 30 innovation teams supported by the Gugeek KJ-Method creativity support groupware. For qualitative evaluation, the result is concluded based on participant cross-evaluation (pressing like buttons of favorite ideas and categories) and the result of questionnaires. According to the quantitative evaluation, CI was statistically significantly better than other communication methods in the fluency of ideas and the originality of ideas in divergent thinking followed by TC and FF. In contrast, the effect of the communication method was not statistically significant to team creativity in convergent thinking. According to the qualitative evaluation, CI yielded the highest team creativity performance in participant cross-evaluation, but FF is the most favored based on the questionnaire result.

Since the properties of a team have a vital role in collaboration and cooperation, team characteristics, which are the traits of a team combined with those of individual members, have a strong impact on team creativity through groupware use. Team characteristics can be wisely adjusted to achieve a higher degree of team creativity. Four team characteristics, namely: (1) Background Knowledge (BK); (2) Difference in Background Knowledge (DBK); (3) Friendship Closeness (FC); and (4) Difference in Friendship Closeness (DFC), and their relationships with team creativity are studied and examined. Based on the statistical test on the data of 90 KJ charts constructed by 30 innovation teams supported by the Gugeek KJ Method groupware, we discovered the relationships between four team characteristics and team creativity, and also discovered the best team formation to yield the highest creativity performance. According to the Pearson product-moment correlation coefficient test, BK has strong positive relationships with most divergent and convergent thinking measures. DBK and FC have no strong influence on team creativity. Finally, DFC has strong negative relationships with two divergent thinking measures, which are the originality of ideas and the elaboration of ideas, but has no strong effect on convergent thinking measures.

9.2 Conclusion

With the popularity and excellence of the KJ-Method, we developed a KJ-Method creativity support groupware to enable people regardless of their location and time constraints to perform group KJ-Method activities on the largest computer network nowadays, the Internet. Moreover, the proposed groupware "Gugeek" significantly removes many barriers for performing the group KJ-Method such as the cost of stationery, the difficulty in distributing the final KJ-

Chart, and providing persistent searchable idea storage in digital format. It is more convenient to operate, and increases the rate of idea sharing among team members. The Gugeek KJ-Method groupware provides the framework-paradigm support capability. It can store and organize discovered ideas and guide the group of users to think systematically by following the framework of the KJ-Method. For pursuing the highest level of support, the generative level, the Gugeek enhances team creativity performance by using various creativity assistant modules in both divergent and convergent thinking.

For enhancing the divergent thinking capability of a user, we proposed a divergent thinking assistant module, based on the associative search engine GETA queried through the Wikipedia knowledge-base. We found that presenting information to a user generated from the GETAs most related keyword associated engine can promote the idea association performance in divergent thinking at around 34% in the number of useful ideas, and can improve the number of perspectives and original ideas at around 19%.

We also proposed a novel technique for enhancing the convergent thinking capability of a user as well. Automatic category naming based on the knowledge-based automatic topic identification of general ontology, such as YAGO, is proposed. We found that using the proposed harmonic mean, and the existing concept counting topic selection techniques in the automatic category naming assistant module, can automatically predict the category title of the untitled group of labels at around 44% and 49% for noun and verb topics respectively.

Achieving the generative level of support cannot guarantee the efficiency of the proposed KJ-Method groupware, but the social influences among team members, which have a strong impact on team creativity, should also be carefully considered. The questions are focused on two major factors of team characteristics, which are the communication method and characteristics. We found that the task interaction only team communication method, which disallows text chat and face-to-face communication among team members, is the most suitable method for the KJ-Method creativity support groupware due to its performance on team creativity. We also performed experiments to understand how the formation of team members contributes to the characteristics of the team to yield the highest creativity. We found that the best formation for the highest team creativity performance consists of team members who have high background knowledge in problem solving, and the same degree of friendship regardless of its level.

According to these discoveries, the Gugeek KJ-Method creativity support system can improve dramatically in creativity assistants for both the divergent and convergent thinking phases, and the groupware design for positive social influence dimensions. According to the traditional design of Gugeek, the input textbox should display the GETAs most related keywords for promoting idea association performance in the divergent thinking phase. For the titles of new created categories, these should show the choice of category titles automatically generated by using the harmonic mean topic selection method in the convergent thinking phase. The team communication method should rely on the chart interaction only for obtaining the highest team creativity performance, but because of the favored face-to-face meeting of most participants, the real-time video or voice chat should be included in the system. To yield the highest team creativity, the automatic team member selection or automatic team construction could be performed, if a company wants to establish multiple teams to perform the Gugeek KJ-Method groupware on the same or different problem topics simultaneously. All participants will be surveyed or tested for background knowledge in the problem topic under consideration and the friendship closeness between individuals. The automatic team member sections can establish the team formation to yield the maximum overall creativity performance.

9.3 Future Research

Even though there are many research works about the KJ-Method creativity support groupware, there is still a lot of scope in this field as shown in Figure 9.1 For example:

1. **Supporting Each Step of the KJ-Method:** Research for novel creativity assistants to promote human divergent and convergent thinking.
2. **Traits of a Team and Team Creativity:** Understand which traits of an individual or team significantly affects team creativity.
3. **KJ-Method Groupware Design and Process:** Research for the best groupware designs that aids a team to yield their highest creativity performance.
4. **Different Type of Media:** The primary media for conveying information in the KJ-Method Groupware is text. How about other types of media?
5. **Usage in Complex Situations that Involve Many People:** Understand the advantages, disadvantages, and problems of creativity support groupware in such complex situations.
6. **Creativity Motivation:** Understand and find ways to increase the motivation of users and their willingness to propose creative ideas.
7. **Computer Creative Thinking as a Human:** Discover the nature of humanity in the thought process and create, mimic and transfer it into a machine.
8. **Converting the Representation of Other Knowledge Sources to the KJ-Chart, or Vice Versa:** Convert an existing mundane knowledge source such as a research article into an exciting KJ- Chart.

Note that idea labels and categories that have a turn-on light bulb in Figure 9.1 are considered by the author to be interesting and challenging research topics. In the authors opinion, the most interesting research topics for the future are creativity motivation and the ability of computers to think as humans.

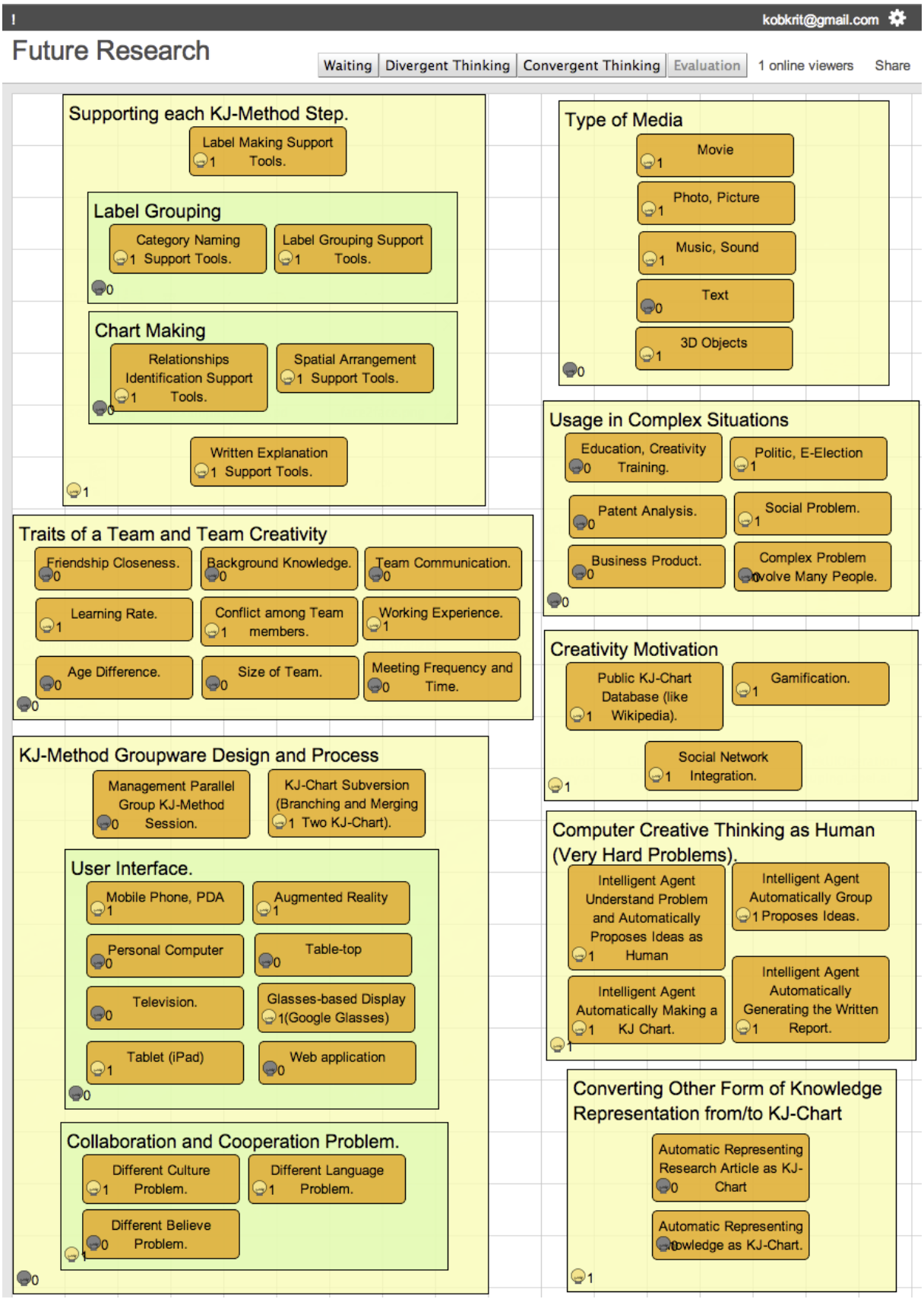


Figure 9.1: A KJ-Chart considering future research for KJ-Method groupware.

Publications

International Journal

[1] Kobkrit Viriyayudhakorn, and Susumu Kunifuji: "Automatic Topic Identification for Idea Summarization in Idea Visualization Programs," IEICE Transaction Information and System, Vol. E96-D, No.1, pp. 64-72, January 2013. (Peer-reviewed)

International Conferences

[2] Kobkrit Viriyayudhakorn, and Susumu Kunifuji: "The Impact of the Background Knowledge and the Friendship Closeness on the Team Creativity in a Realtime Creativity Support Groupware," In Proceedings of *The Seventh International Conference on Knowledge, Information and Creativity Support Systems (KICSS2012)*, pp. 89-96, Australia, 8-9 November 2012. (Peer-reviewed)

[3] Kobkrit Viriyayudhakorn, and Susumu Kunifuji: "The Effectiveness of Three Communication Methods in a Realtime KJ Method Creativity Support Groupware," In Proceedings of *The 16th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems (KES2012)*, pp. 1837-1846, Spain, September 2012. (Peer-reviewed)

[4] Kobkrit Viriyayudhakorn, Susumu Kunifuji, and Mizuhito Ogawa: "A Comparison of Four Association Engines in Divergent Thinking Support Systems on Wikipedia," In Proceedings of *The Fifth International Conference on Knowledge, Information and Creativity Support Systems (KICSS2010)*, pp. 226-237, Thailand, 25-27 November 2010. (Peer-reviewed)

Bibliography

[1] Osamu Mimura. KJ Method and W-typed Problem Solving Model, 2011.

[2] J. Munemori and Y. Nagasawa. Development and trial of groupware for organizational design and management: distributed and cooperative KJ method support system. *Information and Software Technology*, 33(4):259–264, 1991.

[3] J. MUNEMORI and Y. NAGASAWA. GUNGEN: groupware for new idea generation system. *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, 75(2):171–178, 1992.

[4] T.I. Lubart and C. Mouchiroud. Creativity: A source of difficulty in problem solving. *The psychology of problem solving*, pages 127–148, 2003.

[5] Lawrence F. Young. Idea Processing Support: Definitions and Concepts, chapt. 8. *Decision Support and Idea Processing Systems*, pages 243–268, 1988.

[6] A. Pinar Saygin, I. Cicekli, and V. Akman. Turing test: 50 years later. *Minds and Machines*, 10(4):463–518, 2000.

[7] F.C. Pereira. *Creativity and artificial intelligence: a conceptual blending approach*, volume 4. Walter de Gruyter, 2007.

[8] B. Shneiderman. Creativity support tools: accelerating discovery and innovation. *Communications of the ACM*, 50(12):20–32, 2007.

[9] J. Kawakita. The KJ method: a scientific approach to problem solving. Technical report, Kawakita Research Institute, Tokyo, 1975.

[10] P. Wilson. *Computer Supported Cooperative Work:: An Introduction*. Springer, 1991.

[11] H. Chen, P. Hsu, R. Orwig, L. Hoopes, and J. F. Nunamaker. Automatic concept classification of text from electronic meetings. *Commun. ACM*, 37(10):56–73, October 1994.

[12] J. Kawakita. The original KJ method. *Tokyo: Kawakita Research Institute*, 1991.

[13] Tadao Umesao. *The Art of Intellectual Production*. Iwanami Shoten, Tokyo, Japan, 1969.

[14] Jiro Kawakita. *An Idea Development Method*. Chuuo Kouronsha, Tokyo, Japan, 1967.

[15] Susumu Kunifuji, Naotaka Kato, and Andrzej Wierzbicki. Creativity Support in Brainstorming. In Andrzej Wierzbicki and Yoshiteru Nakamori, editors, *Creative Environments*, volume 59 of *Studies in Computational Intelligence*, pages 93–126. Springer, 2007.

[16] J. Spool. The KJ-technique: a group process for establishing priorities. *User Interface Engineering*, 2004.

[17] Susumu Kunifuji and Naotaka Kato. Consensus-Making Support Systems Dedicated to Creative Problem Solving. *International Journal of Information Technology and Decision Making*, 6(3):459–474, 2007.

[18] Alex Osborn. *Your Creative Power*. New York : Charles Scribner's Son, 1948.

[19] K. Viriyayudhakorn, S. Kunifuji, and M. Ogawa. A Comparison of Four Association Engines in Divergent Thinking Support Systems on Wikipedia. *Knowledge, Information, and Creativity Support Systems*, pages 226–237, 2011.

[20] K. Misue, K. Sugiyama, and Fujitsu Laboratories. International Institute for Advanced Study of Social Information Science. *An overview of diagram based idea organizer: D-abductor*. International Institute for Advanced Study of Social Information Science, Fujitsu Ltd., 1993.

[21] T. Kawaji and S. Kunifuji. A prototyping and an evaluation of an idea generation support groupware Hasso-tobi. *Journal of Japan Creativity Society*, 4:18–36, 2000.

[22] H. Ohiwa, N. Takeda, K. Kawai, and A. Shiomi. KJ Editor: a card-handling tool for creative work support. *Knowledge-Based Systems*, 10(1):43–50, 1997.

[23] T. Shigenobu, T. Yoshino, and J. Munemori. Idea generation support system GUNGEN DX II beyond papers. In *Knowledge-Based Intelligent Information and Engineering Systems*, pages 741–747. Springer, 2003.

[24] R.B. Gallupe, A.R. Dennis, W.H. Cooper, J.S. Valacich, L.M. Bastianutti, and J.F. Nunamaker Jr. Electronic brainstorming and group size. *Academy of Management Journal*, pages 350–369, 1992.

[25] U. Neupane. Qualitative, Quantitiative Evaluation of Ideas in Brain Writing Groupware. *IEICE TRANSACTIONS on Information and Systems*, 2007.

[26] H. Ohiwa, K. Kawai, and M. Koyama. Idea processor and the KJ method. *Journal of Information Processing*, 13(1):44–48, 1990.

[27] N. Takeda, K. Kawai, M. Koyama, A. Shiomi, and H. Ohiwa. KJ-Editor: an index-card style tool. In *Visual Languages, 1992. Proceedings., 1992 IEEE Workshop on*, pages 255–257. IEEE, 1992.

[28] H. Ohiwa. KJ editor for creative work support and collaboration. In *Creating, Connecting and Collaborating Through Computing, 2003. C5 2003. Proceedings. First Conference on*, pages 104–109. IEEE, 2003.

[29] M. Takahashi. The bible of creativity, 2002.

[30] K. Sugiyama and K. Misue. A generic compound graph visualizer/manipulator: D-ABDUCTOR. In *Graph Drawing*, pages 500–503. Springer, 1996.

[31] K. Misue, K. Nitta, K. Sugiyama, T. Koshiba, and R. Inder. Enhancing D-ABDUCTOR towards a diagrammatic user interface platform. In *Knowledge-Based Intelligent Electronic Systems, 1998. Proceedings KES'98. 1998 Second International Conference on*, volume 1, pages 359–368. IEEE, 1998.

[32] M. Miura, T. Sugihara, and S. Kunifuji. GKJ: Group KJ method support system utilizing digital pens. *IEICE TRANSACTIONS on Information and Systems*, 94(3):456–464, 2011.

[33] Motoki Miura Takahiro Nyu. Evaluation of gesture-command input method for pen-based group KJ system. In *the 16th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems (KES2012)*, September 2012.

[34] Takahiro Nyu and Motoki Miura. Editing the Visual Properties of Collaborative KJ Diagram Decoration by User-Defined Handwritten Gestures. In *WMUTE*, pages 208–210, 2012.

[35] J. Munemori and Y. Nagasawa. GUNGEN: groupware for a new idea generation support system. *Information and Software Technology*, 38(3):213–220, 1996.

[36] T. Yuizono, J. Munemori, and Y. Nagasawa. GUNGEN: groupware for a new idea generation consistent support system. In *Computer Human Interaction, 1998. Proceedings. 3rd Asia Pacific*, pages 357–362. IEEE, 1998.

[37] T. Yuizono, T. Yoshino, and J. Munemori. GUNGEN-GO: Real-Time Groupware Development Environment for a Hypermedia System. In *Knowledge-Based Intelligent Information and Engineering Systems*, pages 779–785. Springer, 2003.

[38] Shigenobu Tomohiro, Yoshino Takashi, and Munemori Jun. GUNGEN DX II : Idea generation support system with grouping-labels function for hundreds of labels. *IPSJ Journal*, 46(1):2–14, 2005-01-15.

[39] T. Shigenobu, T. Yoshino, and J. Munemori. Evaluation and application of creativity collaboration support system GUNGEN DX II for consensus-building among users. *International Journal of Information Technology & Decision Making*, 6(03):475–490, 2007.

[40] T. Yuizono, J. Munemori, and Y. Nagasawa. Application of groupware for a new idea generation consistent support system using PDA for input device. In *Parallel Processing, 1999. Proceedings. 1999 International Workshops on*, pages 394–399. IEEE, 1999.

[41] T. Yoshino, J. Munemori, and K. Yunokuchi. Development of a PDA based idea collecting system and its application to an idea generation consistent support system. In *Information Networking, 2001. Proceedings. 15th International Conference on*, pages 737–742. IEEE, 2001.

[42] Takaya Yuizono, Jun Munemori, and Tomohiro Shigenobu. Effects of groupware for an idea generation with a large collaboration interface. *IPSJ Journal*, 49(7):2574–2588, 2008-07-15.

[43] Takaya Yuizono and Zeying Yu. The Effects of a Second Language on the Distributed and Cooperative KJ Method in Text-shared Collaboration. In *KES*, pages 1877–1886, 2012.

[44] Jun Munemori, Hiroshi Fukuda, and Junko Itou. Application of a Web-Based Idea Generation Consistent Support System. In *KES*, pages 1827–1836, 2012.

[45] G.S. Sidhu, R.F. Andrews, and A.B. Oppenheimer. *Inside AppleTalk*. Addison-Wesley Reading (MA) etc, 1990.

[46] P. Vossen and J. Hofmann. Using Timbuktu and Guide for Computer Supported Group Learning. *Cognitive tools for learning*, 81:185, 1992.

[47] D. Goodman. The Complete HyperCard Handbook. 1987, 1995.

[48] M. Stefik, G. Foster, D.G. Bobrow, K. Kahn, S. Lanning, and L. Suchman. Beyond the chalkboard: computer support for collaboration and problem solving in meetings. *Communications of the ACM*, 30(1):32–47, 1987.

[49] G. Foster and M. Stefik. Cognoter: theory and practice of a colab-orative tool. In *Proceedings of the 1986 ACM conference on Computer-supported cooperative work*, pages 7–15. ACM, 1986.

[50] Youji Kohda, Isamu Watanabe, Kazuo Misue, Shinichi Hiraiwa, and Motoo Masui. Group Idea Processing System: GrIPS. *Transactions of the Japanese Society for Artificail Intelligence*, 8:601–601, 1993.

[51] Isamu Watanabe. Idea Editor: Overview. Technical Report Research Report IIAS-RR-90-2E, Fujitsu Laboratories Limited, October 1990.

[52] I. Nonaka, K. Umemoto, and D. Senoo. From information processing to knowledge creation: a paradigm shift in business management. *Technology in society*, 18(2):203–218, 1996.

[53] Watanabe Isamu. Divergent thinking support system : keyword associator. *Proceedings of SICE Joint Symposium*, pages 411–418, 1991. (in Japanese).

[54] H. Ueda and S. Kunifuji. GRAPE: knowledge acquisition support groupware for the classification-choice problemthe design principles, groupware functions, and a suggestion to extend for the planning problem. *Modern Tools for Manufacturing Systems*, pages 119–135, 1993.

[55] A.F. Osborn. *Applied Imagination; Principles and Procedures of Creative Problem-solving: Principles and Procedures of Creative Problem-solving*. Scribner, 1963.

[56] E.B. Herreras. Cognitive neuroscience, The biology of the mind. *Cuadernos de Neuropsicología*, 4(1):87–90, 2010.

[57] Takahiro Kawaji and Susumu Kunifuji. Divergent Thinking Supporting Groupware by Using Collective Intelligence. In *Proceedings of The Third International Conference on Knowledge, Information and Creativity Support Systems (KICSS2008)*, pages 192–199, 2008.

[58] Hao-Chuan Wang, Dan Cosley, and Susan R. Fussell. Idea Expander: Supporting group brainstorming with conversationally triggered visual thinking stimuli. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, pages 103–106. ACM, 2010.

[59] Lawrence F. Young. The metaphor machine: A database method for creativity support. *Decision Support Systems*, 3(4):309 – 317, 1987.

[60] U.M. Borghoff and J.H. Schlichter. *Computer-supported cooperative work: Introduction to distributed applications*. Springer, 2000.

[61] World Wide Web Consortium (W3C). Same origin policy, January 2010.

[62] J. Oikarinen and D. Reed. RFC 1459. *Internet Relay Chat Protocol*, 1993.

[63] C. Lueg. *From Usenet to CoWebs: Interacting with social information spaces*. Springer Verlag, 2003.

[64] Akihiko Takano. Association computation for information access. In *Discovery Science*, pages 33–44. Springer, 2003.

[65] Nachum Dershowitz and Zohar Manna. Proving Termination with Multiset Orderings. *Communication of the ACM*, 22(8):465–476, 1979.

[66] Amit Singhal, Chris Buckley, Mandar Mitra, and Ar Mitra. Pivoted Document Length Normalization. In *Document length normalization*, pages 21–29. ACM Press, 1996.

[67] S.E. Robertson and S. Walker. Okapi/Keenbow at TREC-8. *NIST Special Publication SP*, pages 151–162, 2000.

[68] Ross Wilkinson, Justin Zobel, and Ron Sacks-Davis. Similarity measures for short queries. *NIST Special Publication SP*, pages 277–286, 1996.

[69] GETAssoc ITB File Format | GETAssoc. http://getassoc.cs.nii.ac.jp/en/?itb%20File%20Format.

[70] Joy Paul Guilford. *Traits of Creativity*, chapter Creativity and its Cultivation. Harper, 1959.

[71] Ujjuwal Neupane, Motoki Miura, Tessai Hayama, and Susumu Kunifuji. Distributed Environment for Brain Writing Support Groupware and its Evaluation, Tree and Sequences. *Journal of Japanese Creativity Society*, 10:74–86, 2006. (in Japanese).

[72] G.A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K.J. Miller. Introduction to WordNet: an on-line lexical database. *International journal of lexicography*, 3(4):235–244, 1990.

[73] F.M. Suchanek, G. Kasneci, and G. Weikum. Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, pages 697–706. ACM, 2007.

[74] Chin-Yew Lin. Knowledge-based automatic topic identification. In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, ACL '95, pages 308–310, Stroudsburg, PA, USA, 1995. Association for Computational Linguistics.

[75] S. Tiun, R. Abdullah, and T. Kong. Automatic topic identification using ontology hierarchy. *Computational Linguistics and Intelligent Text Processing*, pages 444–453, 2001.

[76] X. He, C.H.Q. Ding, H. Zha, and H.D. Simon. Automatic topic identification using Webpage clustering. In *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, pages 195–202. IEEE, 2001.

[77] H.C. Özmutlu, F. Çavdur, S. Özmutlu, and A. Spink. Neural network applications for automatic new topic identification on excite Web search engine data logs. *Proceedings of the American Society for Information Science and Technology*, 41(1):310–316, 2004.

[78] C. Clifton, R. Cooley, and J. Rennie. TopCat: data mining for topic identification in a text corpus. *Knowledge and Data Engineering, IEEE Transactions on*, 16(8):949–964, 2004.

[79] K. Coursey, R. Mihalcea, and W. Moen. Using encyclopedic knowledge for automatic topic identification. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, pages 210–218. Association for Computational Linguistics, 2009.

[80] O. Muñoz-García, A. García-Silva, O. Corcho, M. de la Higuera Hernández, and C. Navarro. Identifying Topics in Social Media Posts using DBpedia. *Proceedings of the NEM Summit 2011*, pages 81–86, 2011.

[81] D.B. Leake, A. Maguitman, and T. Reichherzer. Topic extraction and extension to support concept mapping. In *Proc. of the Sixteenth FLAIRS*, 2003.

[82] J. Villalon and R.A. Calvo. Concept extraction from student essays, towards concept map mining. In *Advanced Learning Technologies, 2009. ICALT 2009. Ninth IEEE International Conference on*, pages 221–225. IEEE, 2009.

[83] F. Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83, 1945.

[84] C.D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*, volume 1. Cambridge University Press Cambridge, 2008.

[85] M. Porter. Snowball: A language for stemming algorithms, 2001.

[86] D. Lin. Automatic retrieval and clustering of similar words. In *Proceedings of the 17th international conference on Computational linguistics-Volume 2*, pages 768–774. Association for Computational Linguistics, 1998.

[87] Bernard A Nijstad, Wolfgang Stroebe, and Hein F.M Lodewijkx. Cognitive stimulation and interference in groups: Exposure effects in an idea generation task. *Journal of Experimental Social Psychology*, 38(6):535 – 544, 2002.

[88] J. Kratzer, O.T.A.J. Leenders, and J.M.L. Engelen. Stimulating the potential: Creative performance and communication in innovation teams. *Creativity and Innovation Management*, 13(1):63–71, 2004.

[89] J.A. Goncalo. Past success and convergent thinking in groups: The role of group-focused attributions. *European Journal of Social Psychology*, 34(4):385–395, 2004.

[90] E.P. Torrance. *Torrance tests of creative thinking*. Personnel Press, Incorporated, 1968.

[91] S. Sugiura, J. Munemori, T. Kinoshita, and N. Shiratori. A proposal of an evaluation method IPL for effects of individual quantity of domain knowledge on an intuitive classification in a distributed and cooperative KJ method and its application. In *Information Networking, 1998. (ICOIN-12) Proceedings., Twelfth International Conference on*, pages 581–586. IEEE, 1998.

[92] A.B.R. Shani, J.A. Sena, and M.W. Stebbins. Knowledge work teams and groupware technology: learning from Seagates experience. *Journal of Knowledge Management*, 4(2):111–124, 2000.

[93] Z. Gang and N. Xudong. The Impact of Knowledge Difference and Knowledge Conflict on the Creativity of Team. *Acta Psychologica Sinica*, 5, 2007.

[94] R.K. Sawyer. *Explaining creativity: The science of human innovation*. Oxford Univ Pr, 2012.

[95] S. Lau and G.Y.C. Ng. *Creativity: When East Meets West*. World Scientific, 2004.

[96] M.A. Runco, J. Nemiro, and H.J. Walberg. Personal explicit theories of creativity. *The Journal of Creative Behavior*, 32(1):1–17, 1998.

[97] G. Ramalho and J.G. Ganascia. Simulating creativity in jazz performance. In *Proceedings of the National Conference on Artificial Intelligence*, pages 108–108. Citeseer, 1994.

[98] E. Edmonds and L. Candy. Creativity, art practice, and knowledge. *Communications of the ACM*, 45(10):91–95, 2002.

[99] S.K. Sim and A.H.B. Duffy. Knowledge transformers: a link between learning and creativity. *AI EDAM: Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 18(03):271–279, 2004.