

We approve the thesis of (Name Goes Here).

Date of Signature

Name of Associate Professor Advisor
Associate Professor of Astronomy and Astrophysics
Thesis Adviser
Chair of Committee

Associate Prof. Member1
Associate Professor of Astronomy and Astrophysics

Assistant Prof. Member2
Assistant Professor of Astronomy and Astrophysics

Associate Prof. Member3
Associate Professor of Physics

Evan Pugh Prof. Member4
Evan Pugh Professor of Astronomy and Astrophysics

Assistant Prof. Member5
Assistant Professor of Astronomy and Astrophysics

Professor and Department Head Member6
Professor of Astronomy and Astrophysics
Head of the Department of Astronomy and Astrophysics

The Pennsylvania State University

The Graduate School

Department of Astronomy and Astrophysics

SUPPORTING COLLABORATIVE INFORMATION ANALYSIS
USING INTERACTIVE VISUALIZATION: A DESIGN
RESEARCH

A Thesis in

Astronomy and Astrophysics

by

(Name Goes Here)

© 2002 (Name Goes Here)

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

May 2002

The thesis of (Name Goes Here) was read and approved¹ by the following:

Name of Associate Professor Advisor
Associate Professor of Astronomy and Astrophysics
Thesis Adviser
Chair of Committee

Associate Prof. Member1
Associate Professor of Astronomy and Astrophysics

Assistant Prof. Member2
Assistant Professor of Astronomy and Astrophysics

Associate Prof. Member3
Associate Professor of Physics

Evan Pugh Prof. Member4
Evan Pugh Professor of Astronomy and Astrophysics

Assistant Prof. Member5
Assistant Professor of Astronomy and Astrophysics

Professor and Department Head Member6
Professor of Astronomy and Astrophysics
Head of the Department of Astronomy and Astrophysics

¹Signatures on file in the Graduate School.

Abstract

Abstract

Table of Contents

List of Tables	viii
List of Figures	ix
Acknowledgments	xii
Chapter 1. Introduction	1
1.1 Introduction	1
1.1.1 Motivation	1
1.1.2 Problem	2
1.1.3 Research methods	4
1.1.4 Summary of research result	4
1.1.5 Dissertation structure	4
Chapter 2. Literature Review	6
2.1 Task of Collaborative Information Analysis	6
2.1.1 Modeling the task of information analysis	7
2.1.2 Structured techniques	13
2.2 Collaboration and Awareness	19
2.3 Technology to Support Collaborative Information Analysis	22
Chapter 3. CAnalytics: Tool Supporting Collaborative Information Analysis	28

3.1	Task Analysis	28
3.1.1	Major activities: data annotation, data analysis, and hypothesis development	29
3.1.2	Collaboration	32
3.2	Design Objectives	33
3.2.1	Objective 1. Provide an integrated, dynamic analytic workspace	33
3.2.2	Objective 2. Support structured data annotating and visualization techniques	34
3.2.3	Objective 3. Support information sharing and team awareness	35
3.2.4	Objective 4. Enabling collaborative hypotheses development .	35
3.2.5	Objective 5. Capture team behavior	36
3.3	CAnalytics Features	36
3.3.1	An integrated workspace: annotation, visualization, and hypothesis	37
3.3.2	Multiple coordinated views of annotated evidence	38
3.3.3	Information sharing and team awareness	40
3.3.4	User activity log	42
3.3.5	Collaborative hypothesis development	42
3.4	Implementation	44
3.4.1	Managing team analytic states	44
3.4.2	Conflict resolution	49
3.4.3	View restoration and sharing	50

Chapter 4. The First Classroom Study	51
4.1 Problem Statement	51
4.2 Setting and Method	51
4.3 System	55
4.4 Results	55
4.4.1 Filtering vs. Accretion	56
4.4.2 Situational logic vs. Comparison	58
4.4.3 Inferential connection vs. Factual discrete	59
4.4.4 Interleaving data annotation and analysis	64
4.4.5 Collaboration and awareness	65
4.4.6 Interaction with team performance	69
4.5 Discussion	70
4.5.1 Scaffold a structured interleaved workflow	73
4.5.2 Represent uncertainty in collaborative analysis	75
4.5.3 Build collapsible views for multi-level modeling	77
4.5.4 Share views as team resource	77
4.5.5 Distinguish visible vs. valuable contributions	79
Chapter 5. The second Classroom Study	80
5.0.1 Problem Statement	80
5.0.2 CAnalytics@v2: supporting collaborative hypothesis development	82
5.1 Use case demonstration	85

5.1.1 Classroom study setting	85
5.2 Result	86
5.2.1 Confirming result from study one	86
5.2.2 Feedback on hypothesis development tool	89
5.3 Discussion	93
Chapter 6. Reflection on Evolution of CAnaytics:	
A Design Research	96
6.1 Reflection on method	96
6.1.1 Classroom study	96
6.1.2 A design research of groupware	98
6.1.3 Design research	100
6.2 Reflection on collaborative information analysis support	100
6.2.1 Analyst in control	100
6.2.2 The cost vs. benefits of structured techniques	101
6.2.3 The cost vs. benefits of collaboration	102
6.2.4 Education in collaborative information analysis	103
Chapter 7. Conclusion	105
Bibliography	106

List of Tables

3.1 Specification of user activities	43
--	----

List of Figures

2.1 Pirolli and Card’s Think Loop Model)	9
2.2 Wheaton’s multiphasic model of information analysis process)	10
3.1 Participant reported intelligence analysis workflow)	32
3.2 CAnalytics user interface	37
3.3 Annotation in document view	39
3.4 Utility for location annotation	39
3.5 Utility for location annotation	43
3.6 State management in a dynamic system.	47
3.7 Diagrams showing two state management approaches. a) Event-driven approach: each interface component manages its own state, and updates itself driven by events. b) Our approach: a central state manager takes care of all events and notifies interface components whenever there is a need to update.	48
3.8 Event flow in a collaborative system.	49
4.1 Classroom setting	53
4.2 Survey responses (box shows Q1-Q3 and median; ends of whiskers show maximum and minimum)	56
4.3 Network artifact comparison: filtering (a) vs. accretion (b)	57

4.4 Pie charts showing different labor division strategies. Each pie chart corresponds to one team member. (a) Document-based collaboration. The pie chart shows the documents one team member annotated, color coded by document ID. (b) Entity-based collaboration. The pie chart shows the entity types one team member created, color coded by entity types.	60
4.5 Graph showing the timeline of one team creating annotations. Each row corresponds to one team member. Each bar represents an annotation, color coded by document ID. The red blocks highlights the periods when all teammates worked on the same documents simultaneously.	60
4.6 Network artifact comparison: separate clusters (a) vs. connected clusters (b). The parts highlighted in red squares in (b) are key evidence that connects clusters	61
4.7 Three different representations of uncertainty for entity relationships . .	63
4.8 (a) Visualization of interaction logs of Team 107. Each row of colored marks indicates the sequence of top level activities a participant performed. (b) State transition diagram of interaction logs of Team 107. Each node is an activity, whose size represents the time spent on the it (M: data modeling; A: data analysis; H: hypothesis development; C: coordination); a link represents a switch from one activity to another, whose width encodes the number of switches. We see highly frequent transitions between data modeling and data analysis	65

4.9 The relationship between collaboration characteristics and team performance.	71
5.1 The new timeline displays events schematized by people	84
5.2 The second classroom study setting	87
6.1 Task-artifact co-evolution structured with design methodology from ? . .	100

Acknowledgments

Here is where the acknowledgments go if you have any.

Chapter 1

Introduction

1.1 Introduction

1.1.1 Motivation

Collaborative information analysis is a form of sensemaking wherein a team analyzes a complex information space of facts and relationships to identify and evaluate causal hypotheses. It is critical in various domains. A common example is crime investigation; a variety of putative facts are assembled, including financial records, witness observations and interviews, and social connections of various sorts among persons of interest, from which investigators collaboratively assess means, motives, and opportunities, articulate and investigate further hypotheses and deductions, and develop one or more theories of the crime. The volume of information and level of sophistication of relationships among information could easily increase and go beyond the capability of human cognition.

Effectiveness of collaboration determines team performance, and thus the justice of decisions made or solutions proposed. Researchers have repeatedly found evidence of low performance of distributed groups compared to face-to-face groups. They have attributed the performance issues to the difficulty in maintaining awareness and extra efforts required to maintain *awareness* for distributed groups. To have smooth collaboration, people need to know a lot about their collaborators: where they are, what they are looking at, what

they are working on, what they recently did, what they are planning to do, what they know, what skills they have, who they know that might know something, and so forth. None of them, however, is intuitively available in distributed settings.

Indeed, many deficits exist that could result in disruptions in the flow of communication within distributed groups: field of view is diminished, facial expressions are limited, the possibility to use gestures is reduced, sharing of tools and artifacts is constrained, and exchanged information could be delayed. Further, in case of miscommunication, it is difficult to repair, or even to discover the resulted misunderstanding. Worse, the challenges become even more salient as the range and complexity of the task expands.

Collaborative information analysis tools, also known as Information analytical groupware Grudin (1994), are designed to facilitate analytic collaboration and collaborative sensemaking. Numerous tools have been developed in the research community. However, these works often focus on amplifying individual cognitive abilities (e.g. Stasko et al. (2008); Bier et al. (2008)), or lack complex serious empirical evaluation studies (Goyal & Fussell (2016); Convertino et al. (2011)), or do not include a full featured supporting tool Carroll et al. (2013); Borge et al. (2012). The state-of-the-art analytic tools that are currently wide used, such as IBM Analyst's Notebook IBM (2017), and PARC ACH PARC (2017), are designed for individual use only. The design space of groupware for complex collaborative information tasks remains largely unexplored.

1.1.2 Problem

This dissertation addresses the problem of *developing and evaluating groupware for supporting collaborative information analysis.*

A critical challenge for information analysts is building an structured preliminary data model, and insuring that the data model is employed effectively in hypothesis development and evaluation. This is an open challenge. Standard structured techniques often do not support it at all; for example, Analysis of Competing Hypotheses (ACH) assumes that data has been modeled, and that relevant evidence can be adduced appropriately to various hypotheses, but provides no structured support for either. My works aims to build an integrated environment which bootstraps a structured approach and analysts can build data model and develop hypotheses in one place.

Another challenge is to develop and maintain team awareness while engaging in a cognitive demanding task. The concept of awareness has been a major focus in the field of Computer-Supported Collaborative Work (CSCW). Research has investigated many aspects of awareness, its role and measurement, but its interaction with information analysis, more specifically, usage of structured techniques is less examined. Support for structured techniques are often designed for individuals, but it is actually best utilized for collaboration. This study focuses on awareness support in applying structured techniques.

Finally, the study aims to understand computer supported collaborative work in a close-to-reality environment. Many studies look into usage of low-fidelity tools and usage in a controlled lab experiment over a short period of time. It remains unclear whether insights gained from those low-fidelity tools can be applied to more advanced interactive tools. This is especially true for collaborative tools, because tools and teams often develop into a new system over time, and can behave quite differently as the system develops more awareness Stahl (2006). While new tools bring about new affordances, they inevitably introduce extra constraints. The new affordances and constraints of the

tool shape the way users approach a task, and even completely transform the task itself Carroll & Campbell (1989). This study examines the process when analysts employ high-fidelity groupware, and observe their behavior in a natural environment, particularly, in a classroom setting.

1.1.3 Research methods

1.1.4 Summary of research result

A collaborative visual analytic system, CAnalytics. Videos of CAnalytics features and use cases are available online: <https://www.youtube.com/playlist?list=PL-5njX53OUIITHWIki5na1CXxQWLpTC35>.

A use case of applying CAnalytics to an intelligence analysis project: <https://www.youtube.com/watch?v=2eE92My09Yo&t=2s&index=4&list=PL-5njX53OUIITHWIki5na1CXxQWLpTC35>

Demo of real time view sharing: <https://www.youtube.com/watch?v=Giz675rM1E&t=2s&index=7&list=PL-5njX53OUIITHWIki5na1CXxQWLpTC35>

Demo of collaborative hypothesis development: <https://www.youtube.com/watch?v=pHq0ukeg2KA&t=0s&index=3&list=PL-5njX53OUIITHWIki5na1CXxQWLpTC35>

1.1.5 Dissertation structure

Chapter 2 models the task of information analysis, and explores popular structured techniques. It then turns to review of research on awareness in collaboration. The chapter ends with a survey of available tools for supporting collaborative information analysis.

Chapter 3 introduces CAnalytics, a collaborative visualization tool I designed and developed for information analysis. I performed a task analysis in account for design

rationale behind CAnalytics. The chapter highlights major features of the system, and critical implementation details.

Chapter 4 and 5 present two classroom studies that investigate the design and evaluation of CAnalytics. Study One focuses on evaluating the design an integrated collaborative workspace and the way the structured technique is performed. Study Two examines collaborative hypothesis development. Each study yields design implications for the field.

Finally, Chapter 6 reflects on the research method and lessons learned from the result.

Chapter 2

Literature Review

2.1 Task of Collaborative Information Analysis

Collaborative information analysis is a form of sensemaking wherein a team analyzes a complex information space, assertions involving people, places, transactions and events with varying relevance to the analysts' concerns from sources with varying credibility and reliability. In general, analysts seek to identify relationships beyond those literally specified in the original data, requiring inference, at least, and typically assumptions plus inference.

We take a broad view of the task in that collaborative information analysis can occur in any group activities that are data intensive. Different forms of collaborative information analysis tasks are being actively explored in various research domains. For example, collaborative information seeking (CIS) Shah (2014) is capturing increasing attention as complementary to traditional individual information seeking. Beyond simply searching and collecting information, researchers note that CIS requires participants also to analyze and understand information Paul & Reddy (2010), which is similar to what we address here. Also, the visualization community investigates interactive visual techniques to assist collaborative analysis, known as collaborative visual analytics Isenberg et al. (2011), in which a team relies on the “use of computer-supported, interactive, visual representations of data with the common goal of contribution to joint information

processing activities” (p. 312). Further, Web 2.0 techniques make social data analysis possible, wherein any people can upload and share dataset, and gain collective insights Morton et al. (2014b).

Collaborative information analysis takes place in various scenarios, including intelligence analysis where professional analysts integrate evidence and expertise to support hypotheses Heuer (1999), scientific collaboration in which researchers piece together established theories and new findings to explain a phenomenon Farooq et al. (2009), crisis management where stakeholders share crisis context and resource distribution to make rescue plans ?Convertino et al. (2011), healthcare services where doctors share patient’s information and make diagnostic decision ?, daily trip plan where a family synthesizes information about tourist resorts and plans the route, and many other scenes.

To address the challenge of cognitive load and collaboration, we target at those information analysis tasks with complex data nature. Specifically we investigate in the domain of Intelligence Analysis.

2.1.1 Modeling the task of information analysis

Given the similarity of information analysis in different domains, researchers have proposed conceptual models to characterize the practice. These models abstract user behavior patterns and help researchers think of these tasks on a high level, thus guiding the design of supporting tools. Pirolli and Card’s Pirolli & Card (2005) Think Loop Model (Figure 2.1) is a well-cited model characterizing information analysis activities. The model describes the iterative process of information foraging and sensemaking in which raw evidence is successively aggregated, filtered, and synthesized into a best hypothesis.

In the model, analysts first identify and collate relevant information from external data sources. From this collection, they then organize an evidence file of propositions that can be used to make inferences. They next integrate evidence into problem-oriented schemata. These schemata are used to articulate hypotheses, which are evaluated to determine the best story of the information. The model is a bottom-up process of structure building, but also includes a local feedback loop at each stage. Thus, analysts can reconsider propositions in the evidence file, asking how they are related, or a given hypothesis, asking what schemata it rests upon. While the Think Loop Model identifies various leverage points for information analytic tools (especially in the area of visual analytics), Pirolli and Card acknowledged that the model was only a starting point to investigate the domain. Many problems remain unanswered by the model. For example, the model is targeted at individual information analysis and does not account for the social process in collaborative information analysis: How do analysts share their evidence files? How do analysts synthesize their information schema? How do analysts negotiate their hypotheses? Besides, the model stands from a data-centric view, describing how data transforms and flows, rather than the way analysts work and transition Kang & Kiesler (2012). The work processes, however, are important to develop a useful and usable system that can be integrated in real practice and ultimately transform the current practice.

While Pirolli and Card's model suggests a more linear-like model, Wheaton Wheaton (2011) addressed a parallel, or multi-phased model. He conceptualized the process as four functions: conceptual modeling, data collection, data analysis, and report production. The four functions are not in sequence, but all begin almost immediately.

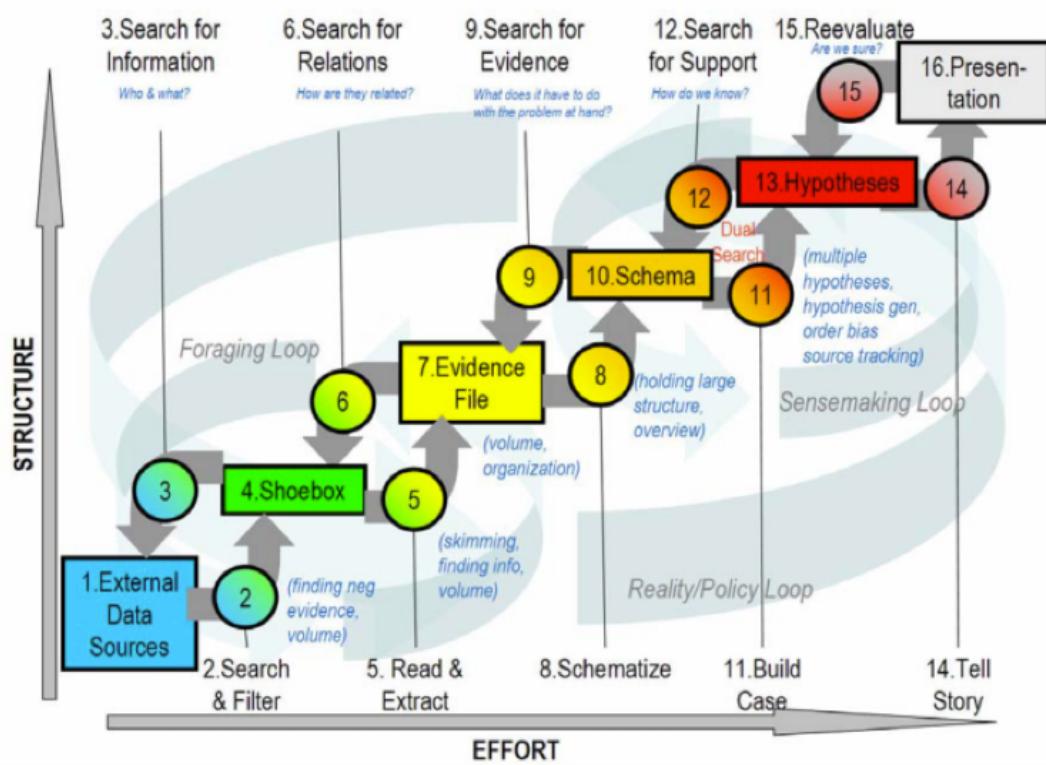


Fig. 2.1. Pirolli and Card's Think Loop Model)

Throughout the course of the analysis, the dominating function could shift at some point, and the amount of time spent on each function could change. Similar patterns were observed in Borge et al.'s study Borge et al. (2012). They observed behaviors of reading, sharing, synthesizing, interpreting, decision making, and receiving new information, yet no linear pattern or order of flow from one behavior to the other is observed. The workflow seems very flexible. Similar reports were conducted in Herrmann et al's study Herrmann et al. (2013).

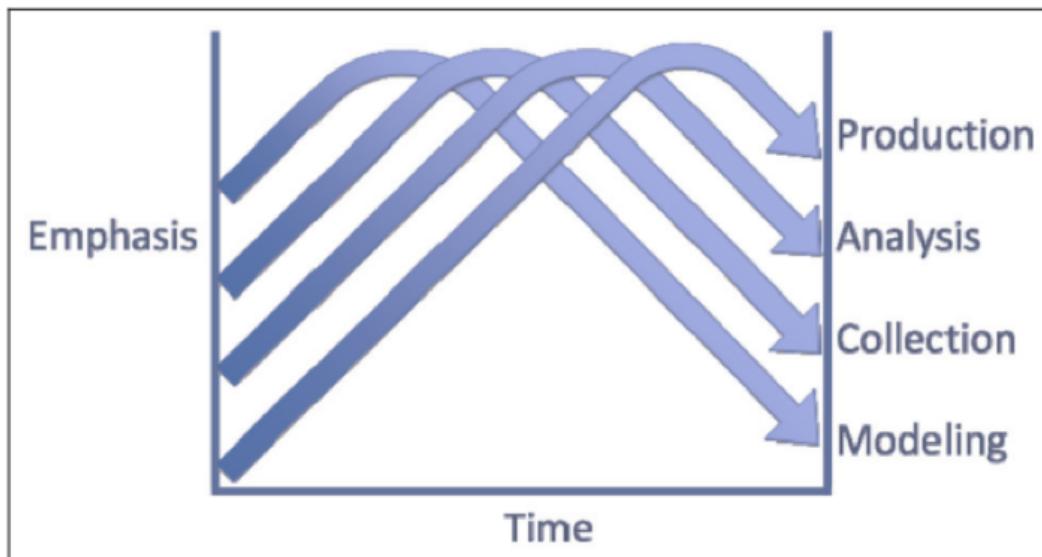


Fig. 2.2. Wheaton's multiphasic model of information analysis process)

Several empirical studies are conducted to contribute to a deeper understanding of the process of collaborative information analysis. For example, Chin, Kuchar and Wolf Chin Jr et al. (2009) observed practices of five professional intelligence analysts. Their report serves as an important source of truth because most professional analysts

are inaccessible due to the classification level of the content they address. They reported analysts first made annotations in their source materials to highlight purported facts about people, places and events, then they constructed various artifacts to hold and present facts, and finally they tried to identify patterns or connections among facts.

As a compromise, Kang and Stasko Kang & Kiesler (2012) chose intelligence analysts in training as research target. They conducted a longitudinal observational field study of their training course. They summarized that the analysis process covered four phases overall: conceptual model construction, information collection, information analysis, and production. Their observation highlighted a number of misconceptions designers might have about the intelligence analysis. For instance, intelligence analysis was more of an exploratory activity than an answer-finding process. Analysts might not have a specific hypothesis in mind; instead, they were like “cutting a path through the jungle that’s never been explored” (Kang & Kiesler 2012, p.145), and did not necessarily know where they were going. Also, analysts did not rely on a specific analytic tool; they tried out various techniques and strategies to solve a problem.

Another type of empirical study is laboratory experiment, in which ordinary people are usually recruited to perform a controlled task, either with a low-fidelity prototype (e.g. paper and pen) or existing commercial analytic tools. The premise is that observations of people’s interactions with non-digital artifacts can help reveal the basic work processes and thus inform functions digital tools should support. People’s interactions with familiar tools would reflect the way they understand and think about the problem at hand. For instance, Carroll et al. Borge et al. (2012); Carroll et al. (2013) developed a crime investigation scenario and recruited college students to investigate

into the case as groups. They identified six primary activities: 1) reading/searching intelligence reports, 2) sharing information, 3) synthesizing information, 4) interpreting information, 5) making final decisions, and 6) receiving new information. With paper and pen, participants spontaneously created graphic artifacts to aid reasoning and memory. Most frequently seen artifacts included tables, lists, calendars, maps, node-link graphs, and text annotations. When comparing performance across groups, researchers found that in better-performed teams, participants had a higher proportion of push acts, in which people voluntarily shared information with partners without being asked to. Also, groups tended to perform better when they created artifacts to synthesize and schematize findings, in addition to simply record facts. Besides, high performance teams were usually not dominated by an individual. Collaborators contributed their expertise without blocking and shifted authority as needed. Similarly, Isenberg et al. Isenberg et al. (2008) conducted a paper-based experiment and observed participants' temporal sequence of processes in analysis. They derived a framework characterizing the analysis activities, which consists of browse, parse, discuss collaboration style, establish task strategy, clarify, select, operate, and validate. Specifically, they noted that as opposed to many models, no typical temporal ordering of the sequence was evident. Analysts approach the problem with different strategies and workflows depending on the problem and the group dynamics. Findings from these laboratory experiments also contribute to understanding of group behavior in information analysis and inform the design of future systems.

2.1.2 Structured techniques

A distinct character of information analysis is the adoption of structured techniques in analysis. Compared to casual analysis, which often relies on experience and intuition, structured analysis provides a systematic, transparent method to approach the analytic process. The rationale behind structured techniques is effectively encapsulated by Heuer's claim (Heuer 1999, p.31):

Intelligence analysts should be self-conscious about their reasoning process.

They should think about how they make judgments and reach conclusions, not just about the judgments and conclusions themselves.

Structured techniques externalize internal thought in a systematic and transparent manner so that they can be shared, extended, and critiqued by other analysts. Thus understanding structured techniques and their application in the state-of-the-art is critical in designing and supporting collaborative information analysis.

However, due to complexity of these techniques and limited support of technology, practice of structured techniques is often simplified or even abandoned Chin Jr et al. (2009); ?. It is an open question to address the tension between benefit of structured technique and time cost in implementation. In this section we review three structured techniques that have been widely used in the intelligence community, and their challenges.

Analysis of competing hypotheses Analysis of competing hypotheses (ACH) Heuer (1999) emphasizes a rational, transparent process of analysis. The general steps of ACH are designed for systematic assessment of all hypotheses and evidence. These steps include

identifying a complete set of hypotheses, identifying all pieces of evidence, assessing the diagnostic value of each evidence against each hypothesis, and drawing conclusions of the likelihood of hypotheses. The principle of ACH is to refute rather than confirm alternatives. ACH guarantees an appropriate analytic process and increases the chance to avoid common analytical pitfalls. Studies by Lehner et al. Lehner et al. (2008) and Lord et al. Lord et al. (1979) demonstrate the function of ACH in reducing confirmation bias, the tendency to seek and interpret evidence in favor of a theory perceived most likely a priori. PARC ACH PARC (2017) is an implementation of ACH, facilitating hypothesis and evidence input and assessment.

Yet ACH has several pitfalls. For example, ACH does not provide a measure of uncertainty among hypotheses. Efforts have been made to combine ACH with probability reasoning systems, such as pairing with Bayesian Network Karvetski et al. (2013) or Bayesian statistical model Duncan & Wilson (2008). Our project differs from these efforts in that we employ a human-center approach and investigate how technology could assist teams of human analysts in applying structured techniques effectively.

Gelder van Gelder (2008) listed five problems of ACH from a practitioner's experience. For example, ACH treats evidence as an individual entity, forcing analysts to assess the value on its own with each hypothesis. In practice, however, evidence diagnosticity is often mediated by other propositions; judgment of one item of evidence is based on judgment of several other items. The rigid matrix structure of ACH does not allow for factoring of these mediating propositions. Also, the ACH matrix represents hypotheses in a "flat" structure: each hypothesis is entered individually across the top row. In many cases, a more flexible representation of hypotheses is needed. For example, a

general hypothesis can have sub-hypotheses, and two seemingly distinct hypotheses in the beginning of analysis might evolve into another alternative in later stage. Such hierarchical structure of hypotheses are unlikely to be represented in ACH. Besides, judgment of the consistency of evidence with each hypothesis is decontextualized. Analysts have to make a judgment on whether each item of evidence is consistent or inconsistent with hypotheses, but the supporting evidence for that judgment is not represented or documented anywhere. The issue, known as data provenance, is critical in information analysis Chin Jr et al. (2009). Lack of data provenance makes it difficult to ground a judgment, resolve group decision conflict, or correct mistakes.

In addition, we observed several deficiencies of ACH in our observation of student analysts performing ACH in class when we worked with instructors in the Security and Risk Analysis program. For example, manual efforts are required in generating and updating views of data. Analysts have to manually copy and paste evidence from document into cells of ACH matrix. This is not only a redundant effort, but also subject to errors in the process. This also results in missing connection between evidence in the matrix and source of evidence in the original document. It is difficult to review the credibility or reliability of a piece of evidence later. Besides, information analysis is a dynamic process: analysts tend to re-interpret and re-assess evidence iteratively. Once evidence is changed, analysts have to manually change the evidence in the ACH matrix as well, which again increases the chance of error.

Besides, ACH demands a high expertise bar for analysts. Users are required to identify a complete set of mutually exclusive hypotheses at the very beginning, which is difficult from people with little expertise and experience in the domain. Further,

hypotheses tend to evolve as analysis proceeds. A hypothesis valid in the beginning may no longer be of value later, or two seemingly separate hypotheses in an early stage of analysis could be combined in a way to better explain the situation later on. The assumption ACH makes that all hypotheses and evidence are identified and set in the very beginning limits the dynamic development of analysis.

We propose to address these problems by employing a more flexible approach and leveraging several other structured techniques. Our tool integrates information gathering with information analysis so that analysts can develop hypotheses based on their generated data models and that insights can always be traced back to the context where data models were created. Building on the data models, we leverage such structured techniques as link analysis and evidence marshaling to assist analysts in hypothesis discovery and development.

Link analysis Much of the data in intelligence analysis can be represented in link form, as a collection of connected nodes. Examples include contact reports (two people witnessed together at specific time and place), phone calls, and financial transactions. Link analysis enables analysts to see the bigger picture of connections, and also reveal patterns that might otherwise remain unnoticed. Link analysis helps analysts answer questions such as “who is central in this organization?”, “what role does an individual appear to be playing in an event?”, “how is this event related to other events?”, etc.

Link analysis is a popular structured technique employed in the intelligence community. In their empirical study of professional intelligence analysts, Chin et al. Chin Jr et al. (2009) emphasized the importance of link analysis in the reasoning process.

They observed that analysts created graphs by hand drawing facts and relationships, or used Microsoft PowerPoint to construct the network. A number of specialized tools are available, including i2 Analyst's Notebook IBM (2017). These tools provide node-link visualizations and facilitate storing, modifying, organizing, and querying links. But these tools are designed for individual use only. For collaboration, participants have to either be physically co-located and share the same screen, or screen capture the visualization and manually share it through other tools.

Another issue of link analysis tools, similar to PARC ACH, is that analysts have to duplicate data from the document space to the analytic space; they have to manually construct entities and relationships in a separate tool, which, again, is not only time-consuming, but also might introduce unnecessary errors and lose data provenance.

Moreover, criminal network is often characterized as dynamic rather than static Sparrow (1991). The relationships between entities often have a temporal distribution. For example, a person who is trivial in an organization might become critical later, or in a specific event period. This dynamic network feature often fails to be reflected in link analysis tools due to their lack of temporal data structure.

Our tool employs multiple coordinated views. We display temporal data in a timeline, spatial data in a map view, and relationship data in a node-link graph. Views are coordinated, meaning that change in any one view will cause change in another view. Thus analysts are able to investigate the criminal network during a specific time range when applying a temporal filter on the timeline, or network within a specific geographic area when applying a spatial filter on the map.

Evidence marshaling Evidence marshaling is a common technique to tie evidence to hypotheses and assertions. The technique connects bits of information together and provides a big picture of the story. It is an important step in Pirolli and Card’s sensemaking model Pirolli & Card (2005). Without tool support, analysts often do a simple, informal marshaling in their mind. With an elaborate computer-based method, analysts can coordinate events along a timeline, or organize evidence into stories about typical topics (e.g. what, when, where, who, and how).

Several tools are designed to support evidence marshaling. For example, Entity Workspace Bier et al. (2010) displays an entity-based view of documents. Users can organize their entity-based evidence into collapsible stories. Jigsaw Gorg et al. (2014) implemented a ShoeBox tool for evidence marshaling and note-taking. Users can record hypotheses and connect them with supporting or contradicting evidence. However, they eventually abandoned the design because it “was just too complex and did not allow analysts to take free-style notes in the way they would do on paper” (p. 342). As an alternative, they designed Tablet, a simplified tool for evidence marshaling, which essentially is a networking tool linking entities as well as interpretation of entities. An advantage of the Tablet is that it becomes more flexible and users can do a free-style note-taking much like on paper. Another recent tool is SAVANT Goyal & Fussell (2016), in which participants could make sticky notes anywhere in the dashboard, and connect notes as their sensemaking process proceeds. They also include a hypothesis window, allowing users to explicitly enter their hypotheses and supporting or refuting evidence. Similar to the result of Jigsaw, however, their user study indicates that the interface is too complex and discourages users from using. Sandbox Wright et al. (2006) takes the

metaphor of paper and analysts can arrange information within the workspace to their need. Rich interactions are designed to facilitate information manipulation. Limited user study was conducted, however, to prove its effectiveness in supporting analytic process.

2.2 Collaboration and Awareness

While several attempts, as discussed above, have been made to implement support for various structured techniques, most of them are designed for individual use only. However, prior work (Chin et al. 2009; Warner et al. 2008) and our own prior experience (Rosettex et al. 2004) emphasized that the work of information analysis at non-trivial scales is fundamentally collaborative, and that the single person operating the software is a bottleneck to the team process. A report from National Intelligence, Vision 2015, states that intelligence analysis increasingly becomes a collaborative enterprise, with the focus shifting “from traditional emphasis on self-reliance toward more collaborative operations”. This is a major change from the traditional concept of intelligence analysis as largely an individual activity.

Structured techniques are reported to be best utilized in a group effort so that collaborative opinions insights could be built Heuer (1999). The transparency of structured techniques ensures that divergent or conflicting perspectives among analysts are heard and seriously considered early in the analytic process. For example, ACH is often used in team collaboration because ACH leaves an audit trail showing what hypotheses were identified, what evidence was used, and how they were interpreted. If teammates disagree, the ACH matrix could highlight the source of disagreement, and be used to ground further team discussion. Te@mACH is a recent advance in ACH technology, developed

by Globalitica, LLC, in 2010. It is designed as a collaborative version of PARC ACH. It allows distributed analysts to work on the same ACH matrix simultaneously. Other tools for supporting structured techniques, however, mostly remain a single-user application.

With single-user applications, teams of analysts have to rely on external view sharing tools (Greenberg 1990) or manually share notebooks and screenshot graphs. This has the consequence that the tools are employed only at specific points in an analysis, often only in the early stages during which analysts are working on their own. Indeed, the use of these tools directly diminishes a team's activity awareness, requiring repeated manual resynchronizing to identify redundant or missing pieces of information, analyzes of information, and analytic hypotheses. This can be mitigated through maintaining and pooling individual analysis reports using Google's real-time editing tool set, Google Doc, Google Sheets, Google Calendar, etc. These tools enable collaborative document editing, and provide a revision history list of changes made to a document by each collaborator, supporting awareness of teammates' contributions. However, the Google tools themselves are not integrated; each tool manages its own data representation. Thus an event report in Google Doc cannot be exported to Google Calendar or Google Map. Non-existent or weak support for coordinating multiple visual representations could undermine activity awareness. We found this to be a specific source of problems in our study of spontaneously created physical visualizations in the information analysis task (Carroll et al., 2013).

Vision 2015 also notes a shift "away from coordination of draft products toward regular discussion of data and hypotheses early in the research phase". This emphasizes the importance of close coupling collaboration in the analytic process, as opposed to simple coordination as the final step. A key enabler of such effective and close collaboration is

activity awareness, a critical concept for research in computer-supported collaborative work (CSCW). In prior work Carroll and his team Carroll et al. (2003); ? developed a conceptual framework, conducted design research investigations of software tools Carroll et al. (2003); ?; ?, and carried out field studies ???, and controlled experimental studies ??Convertino et al. (2011). That body of work produced several results: Activity awareness can be measured, though this entails methods beyond those traditionally employed in awareness research, including conversation analysis ? and artifact analysis ?. Activity awareness grows with the development of reciprocal knowledge about, experience with, and trust of partners. It is facilitated when the sharing of collaborator-specific knowledge is implemented as an explicit, public and documented interaction within the collaborative activity; indeed, activity awareness can be increased beyond measured levels in face-to-face interactions Convertino et al. (2011), a rare result in computer-supported cooperative work (CSCW) sometimes referred to as “beyond being there” ?. The growth of activity awareness qualitatively changes coordination strategies that partners adopt (for example, early in a collaboration, members pull information from partners by voicing their needs and concerns to the team, but subsequently, as team members become more aware of the responsibilities and roles of other members, they selectively push information to fellow members). Thus, becoming a team does not merely speed up or improve coordination; it changes what coordination means and what members do to coordinate. Activity awareness engages active and meta-cognitive strategies for improving team cognition, for example, through negotiating and sharing ad hoc representational artifacts Carroll et al. (2013). Finally, activity awareness can

be effectively supported with interactive tools, and the benefits of supporting it can be measured.

In Chin et al's Chin Jr et al. (2009) study, the analysts had distinctive and strong beliefs about the artifacts they created themselves, yet claimed they would not trust another analyst's artifacts, although they believed they could achieve better results by collaborating with others. They would need to review the original source material.

Kang and Stasko Kang & Kiesler (2012) found that analysts worked as a team throughout the process and the degree of collaboration differs depending on the type of task.

2.3 Technology to Support Collaborative Information Analysis

A number of tools have been developed to support information analysis. For example, the Analyst's Notebook IBM (2017) and ACH PARC (2017) are widely used in intelligence analysis areas. However, these tools were developed for individual use only, yet prior work demonstrated that the work of information analysis at non-trivial scales is fundamentally collaborative. In practice analysts have to manually share notebooks or graphs to coordinate their work.

As the need of support for collaborative information analysis is capturing increasing attention, more technologies are being developed. Most of the tools take advantage of visualization techniques and display awareness information in a visual way. This is not hard to understand since the human visual channel has a high bandwidth and can take in a large amount of information in a short time. Visualization can efficiently capture user attention and represent awareness information in a concise and vivid way. Fussell and her

colleagues ??Goyal et al. (2013) have done a series of controlled, comparative laboratory experiments investigating the effect of visualization on collaboration. They found that groups using visualization tools had better performance and shared visualization increased group discussion. Various visualization techniques have been designed to facilitate different facets of awareness, and most of them fall into the categories below.

- Who are collaborators. Knowledge of the existence of potential collaborators and their skills and expertise is important. Forsén et al. ? proposed the concept of Pinpoint, an interactive visualization radially presenting colleagues that are most closely related and affords browsing, filtering, and further exploration of the networks of the organization. Marlow and Dabbish ? examined the effect of visualization on the attitude towards peer collaborators. They found that different details of visualization of work history have a different impact on behaviors and attitudes towards collaborators, which further influence group engagement.

- What are they doing. While it is easy to know what collaborators are doing in face-to-face settings, e.g. pointing at a figure, walking towards a whiteboard, etc., such signals are nonexistent in distributed collaboration and have to be explicitly supported by technology. Therefore Stefik et al., ? introduced WYSIWIS (What You See Is What I See), in which collaborators share the same view. While this design ensures everyone looks at the same thing, it blocks production as only one person has control of the view at one turn. A relaxed-WYSIWIS design Gutwin & Greenberg (1998) was then proposed, which enables team members to have control of their own view while having a miniature overview of partner's workspace, known as radar view ?. Radar view conveys information not only about what collaborators are doing, but also where they are working on. Greenberg et al.

(1996) also proposed the idea of telepointer, where participants can see the mouse cursor of their colleagues. They then further augmented telepointers with traces as means of visualization of the previous motion of the remote cursor. This provides a bit of context of the cursor and makes the mouse gestures easier to interpret. Similarly, Hajizadeh Hajizadeh et al. (2013) attempted to support awareness of collaborator's activities by visualizing their bushing actions. They found that persistent selection, which adds a "fading-away" effect to past selections, gets people more aware of collaborators' activities. Technologies as such are often applied in synchronous collaboration, attempting to help collaborators stay aware of what others are doing by embodying a remote gesture in visualization.

- What have they done. People may not be active in a team for all the time. When they come back it is important to know what collaborators have done and what changes they have made. In their approach to visualizing the change of data, Schumann et al. Schumann et al. (2013) used color to encode when a node was created. They also provided a replay function to show the evolvement. Similarly, Edwards and Mynatt ? visualized the history as a non-linear tree and argued that the design enabled transition from loosely-coupled work to tightly-coupled collaboration. In the context of collaborative Web search, Paul and Morris Paul & Reddy (2010) posited that displaying users' search history avoids duplicated efforts and increases team performance.

- Information sharing. Making sure everyone in the group has access to all information and is aware of the existence of the information is essential for collaborative information analysis. Participants can implicitly share information. For example, Goyal et al. Goyal et al. (2014) examined the consequence when user-generated notes are

automatically shared with other teammates. They found that participants remembered more clues about the task and participants perceived the tool more useful when implicit sharing is available. However, users may not want to share everything. They also need more control over what to share and what to keep private Mahyar & Tory (2013). Convertino et al. Convertino et al. (2011) designed an explicit but low-cost sharing mechanism through a “Copy-To” button. Analysts could create geo features on their private map and only share those features they wanted to onto a public map by manually clicking on the button. The function grants users explicit control over their acts but requires extra efforts. Nobarany et al. Nobarany et al. (2012) attempted to combine the advantages of explicit and implicit sharing. They introduced the concept of “obscurity of privacy” and posited that information in a public space can still maintain some level of privacy, depending on the availability of the information to others, awareness of the availability, and the amount of attention it receives. In their system, users can mark a piece of information as private, public, or published. Private information is only kept to the user who produced it. Public information is visible to all group members but does not show up on a highlighted position. Published pieces are not only visible to the group but also highlighted. The effect of this more sophisticated sharing mechanism, however, required further research, according to the authors.

- Insight sharing. Sharing insights enables people to build on the endeavors of teammates and get aware of what others are thinking or why they are doing that. In the well-known Many Eyes Viegas et al. (2007), a social visual analytic platform where people can collectively analyze data and share visualizations, the primary way to exchange ideas is through the comment system, where users can see other people’s comment on a specific

state of visualization. Another social platform, sense.us Heer et al. (2008), supports doubly-linked discussion, which means that comments and visualizations are bi-linked: a comment is attached with a visualization (the same as Many Eyes), and a specific visualization is linked to comments if anyone has already commented on that state.

Annotation is a widely used technology to externalize analysts' insights. For example, Xiao ? asked participants to annotate the rationale when they are performing an act, and found these annotations help other people understand why their colleagues did that. Hopfer and MacEachren ? designed a geospatial annotation mechanism, which attached annotation to a specific area of the visualization. This provides the annotation with clearer context. Dorn et al. Dorn et al. (2015) went further and constrained an annotation with a temporal dimension. Their system allowed students to annotate on a specific position of a video at a specific time point. The system helped instructors know better about students' puzzles and problems.

While various technologies have been developed to support facets of awareness, many of them lack rigorous empirical evaluations. Little is known how analysts in the real world will employ these technologies in their work processes, and how their work processes will be transformed by these technologies. This study is to investigate effects of technologies on different facets of activity awareness, and to examine the group processes that make the effects happen.

However, tools supporting information analysis are often targeted at a single phase of activity, and thus not supporting the whole workflow. For example, research efforts have been made to understand information collection and modeling Shah (2014); Jansen & Rieh (2010), but little support is provided to extend these models to analysis. Techniques

such as Information Extraction and Weight (IEW) helps structures data evidence but offers no structure to turn the evidence to hypothesis development. Similarly, tools supporting the activity of data analysis assume data has been modeled. For example, Analysis of Competing Hypotheses (ACH) assumes that data has been modeled, and that relevant evidence can be adduced appropriately to various hypotheses, but provides no structured support for either. Analytic tools such as interactive visualization emphasize presenting data in insightful means but provides no utility to data re-modeling. Ware Ware (2012) warned of the “*asymmetry in data rates*” [p.382], pointing out that visual analytic tools emphasized data flowing from systems to users far more than from users to systems. Functionalities are mostly designed to adjust data representation rather than modeling, which are in fact equally important. Similar calls were made in other data intensive task domains as well. For example, in interactive machine learning, researchers Chen et al. (2016); Amershi et al. (2015) call for an all-in-one environment in which machine learning practitioners can iteratively refine training data and evaluate model performance through visualization in one place. Our work aligns with these efforts, and contribute to the design and evaluation of an integrated workspace in supporting information analysis tasks.

Chapter 3

CAnalytics: Tool Supporting Collaborative Information Analysis

3.1 Task Analysis

We investigated into the design needs for collaborative information analysis in three ways. First, we examined existing works that already conducted empirical user studies of information analysts, and in particular details on the works by ?Carroll et al. (2013); Pirolli & Card (2005); ?. These works provide summarization of design needs in terms of support for collaboration and support for analytics, as well as empirical user feedback in support of the arguments. Among them, we used the works by Carroll and his colleagues Carroll et al. (2013); Borge et al. (2012); Borge & Carroll (2014) as the most the important guidelines because their paper prototype study context was most close to our study. In the study they observed 22 teams investigating a crime scenario using paper and pen. Teams of participants spent four hours analyzing 222 propositions embedded in a set of problem documents. This study sheds light on team process patterns, collaboration breakdowns, and possible technical support teams need. We reused their experiment materials in our second user study.

We worked closely with the Red Cell Analytics Lab at the Pennsylvania State University over one year. The lab is an instructor-directed student club that is designed to give students experience in analyzing real world problems to prepare students for

the professional world by helping them develop structured analysis skills and critical thinking capabilities. We worked with a fresh student analyst who had just begun to learn the skills, and two senior analysts who had more than two years experience. We met about once every two weeks, observing what and how they performed analytics and having informal interviews in terms of why and pain points. We also talked to the lab director, Colonel Jake Graham, who had rich in-person experience in professional intelligence analysis before retirement and now a professor teaching analytic skills at the Pennsylvania State University.

We also observed an intelligence analysis course instructed by Colonel Jake Graham. In the course, students learned theories of analysis (e.g. bottom-up analysis vs. top-down analysis, and concepts of hypothesis, assumption, and evidence), analytic techniques (e.g. Analysis of Competing Hypotheses (ACH) Heuer (1999) and Link Analysis Sparrow (1991)), as well as the state-of-the-art tools to support the implementation of those techniques (e.g. PARC ACH PARC (2017) and Analyst's Notebook IBM (2017)). Below we summarized our findings.

3.1.1 Major activities: data annotation, data analysis, and hypothesis development

On a high level, we observed three categories of activities: data modeling, data analysis, and hypothesis generation. A typical workflow is shown in Figure 3.1. Analysts received a collection of documents periodically. The documents included police reports, social media collections, news reports, video summaries, and other textual data. Analysts

must extract useful information from these documents, connect these pieces of information, make hypotheses of event indicators and warnings.

Upon receiving the documents, analysts would read through and mark critical information. When with pen and paper, they highlighted text and made annotations. They would extract these items and organize them in an “Information Extraction and Weighting” table. In the table they investigated each item’s face value and alternative value against analytic problems. When with computer support, they created “entities” in i2 Analyst’s Notebook, a popular software product by IBM for data analysis and investigation. These entities were data objects that described critical attributes and properties. Analysts connected these entities by marking relationships between them. Some frequently seen relationships included ownership, organizational belonging, and phone contact. Location was a special type of entity. To represent the spatial information, analysts replicated the location entities in Google Earth. One interesting detail was that analysts manually exported the location icons used in Analyst’s Notebook and imported them into Google Earth. The purpose was to keep consistent the representations in different tools.

Another special entity type was event, which included a temporal attribute. Event time was important for sequence analysis. Analysts added start and end time of an event, and repeated date and time if it was a recurring event.

With the modeled data, analysts tried to find connections and patterns through data visualization. They made use of Analyst’s Notebook network tool to map out the entities and their relationships. The timeline tool in Analyst’s Notebook displayed the sequential relationship and overlaps between events. They also used Google Earth and

identified spatial relationships among events. To share and keep record of supporting views, they took screenshots and saved in a Microsoft Word document as evidence files.

Based on these evidence files, analysts generated and recorded hypotheses. These hypotheses are in form of text, but often make reference to supporting evidence, either an entity, or a visualization of relationships between entities. To compare between hypotheses, they often employed the technique of Analysis of Competing Hypotheses (ACH). ACH lays out a full set of hypotheses and examines each item of evidence of its diagnostic value against each hypothesis. Analysts sought for evidence to refute hypotheses, and picked the least unlikely one. The final delivery was an intelligence report, in which analysts composed their hypothesis and all supporting evidence.

Note that we describe the three activities in a linear fashion for the sake of writing, but analysis could be performed in any order, and most commonly, in an iterative approach.

One problem in the workflow is a breakdown between activities. Systems that support intelligence analysis are aimed at a single activity and therefore only support part of the overall analysis workflow. This imposes a clear boundary between each of these activities on the analysts. For example, IEW helps structure evidence modeling, but does not extend utilization of evidence to hypothesis generation; ACH assumes that data has been modeled, and that relevant evidence can be adduced appropriately to various hypotheses, but provides no structured support for either. The unintended boundary between phases has the consequence that data modeled in one software cannot be effectively utilized in hypothesis development in another system. And analysts have to

handoff, often via replicating the data in the new system, information between software systems, making it difficult to revisit and revise the data model.

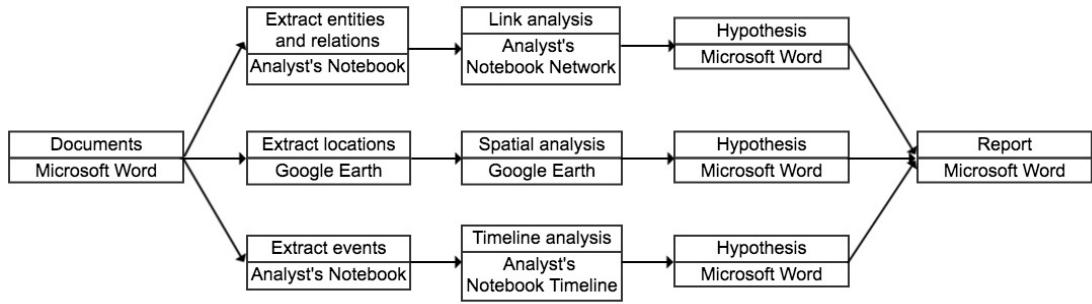


Fig. 3.1. Participant reported intelligence analysis workflow)

3.1.2 Collaboration

Intelligence analysis is fundamentally a collaborative activity because the task can easily become complicated enough that no single individual could handle. Collaborative intelligence analysis then becomes not only a mental process, but also a group process that involves management of team resources (tooling, teammate expertise, etc.), teammate goals and motivations, and partner's contemporary and past activities.

Most tools supporting intelligence analysis, however, are not designed for collaborative use. We observed several pain points in participants' collaboration. For example, participants were unable to contribute simultaneously. Working on the same file would cause conflict. Collaborators had to wait when one teammate was working on the tool. This was known as production blocking Diehl & Strpebe (1987), in which individual performing a task became a bottleneck of team process. To work around the

issue, participants often divided their work by tools: each person picked a tool and then created and analyzed an artifact with the tool on their own. This had the consequence that findings and hypotheses be made without integrating collective efforts and diverse knowledge. Participants shared their findings only after they already had a conclusion, which was obviously opposed to collaboration.

Analysts coordinated work outside the tool, by manually sharing documents or graphs through email or cloud storage service (e.g. Dropbox). As shown in Figure 3.1, analysts manually took screenshots of views and copied them into Microsoft Word. This has the consequence multiple views exist in distributed locations, adding burden to analyst's limited cognition. Analysts could easily overlook certain aspect when they were evaluating hypotheses. The interactive views become static images, making it impossible for collaborators to further explore. Worse, if data model changes, analysts must manually update the views, resulting multiple versions of views.

3.2 Design Objectives

Based on our task analysis, we come up with these design objectives for CAnalytics.

3.2.1 Objective 1. Provide an integrated, dynamic analytic workspace

While functions can be provided to support a single analytic activity (i.e. data annotation alone, or data visualization alone, and indeed a lot of tools exist to support either activity separately), a critical design objective is to integrate these functions in a single workspace. This is more than simply put together these functions; rather, a smooth transition should be enabled between activities. The goal should be achieved

in at least two aspects: (1) the same data pool and structure is utilized in all analytic activities. Data created in data annotation can be visualized in analysis, and utilized in hypothesis development; (2) The system provides a consistent user experience in different activities. Analysts should be confronted with a consistent user interface and interaction throughout the process.

3.2.2 Objective 2. Support structured data annotating and visualization techniques

Our design should allow for a structured approach towards modeling and analyzing data. Analysts spend a large amount of time and efforts annotating critical information from text documents. The essence of annotation is to turn a set of unstructured, noisy data into structured, useful information, while preserving its source. We want to provide a computational method that facilitate the annotation process so that analysts can focus on information itself rather than the logistics.

The annotated data objects should then allow for structured analysis of patterns from multiple aspects through visualization. For example, temporal visualization helps analysts keep track of temporal evolution of events. Spatial visualization helps identify patterns in the same area. Interactions should be provided so that analysts can coordinate and arrange these views into a bigger picture. These closely presented views form into a proximity-based projection Kang et al. (2014), where each view presents a related insight. Analysts can easily navigate these evidence views and marshal them together to support hypothesis development Pirolli & Card (2005).

3.2.3 Objective 3. Support information sharing and team awareness

Collaboration is a process of management of team resources and group process. And a pre-requirement for group management is to stay aware of them. We propose to enable real time data sharing and explicit awareness support. Analysts should be able to share source documents, annotated data objects, patterned views, and hypotheses and insights in real time. Partners can verify who identified and shared the information, and can validate against the source document to confirm sanity. One's activities should be transparent to teammates without extra effort, so that everyone is aware of other's activities.

3.2.4 Objective 4. Enabling collaborative hypotheses development

A more advanced objective is to enable hypothesis development and particularly collaborative development on a team basis. While simply sharing each teammate's hypotheses is beneficial, its positive impact is limited to an extent as it is not completely leveraging the various expertise of a team. Where instead, the combining and evolving of hypotheses from and by multiple members of the team (where the evidence that the hypothesis is based on is included in the hypothesis object) creates an environment where the team is more than just a sum of its parts. We will help analysts to share their insights/hypotheses with supporting visualization early in the process. To impart knowledge, visualizations conveniently improve the understanding of complex problems. When users are exploring the visualization and find an insight, they will be able to share the insight together with the grounding visualization. The visualization will be interactive rather than a static screenshot, keeping the ad hoc visualization state. Insofar as stating

unknowns is a difficult task, at least making what analysts do know more explicit and the assumptions that they made increases a large amount of transparency into their analysis

3.2.5 Objective 5. Capture team behavior

Other than assisting analysts in accomplishing the task more efficiently, we develop the tool as an instrument to gain a deeper understanding of team behavior. The computing tool is able to capture and log all interactions analysts have made. These interaction data provide a promising window into which researchers are able to identify insights in analysis patterns on both individual level and team level. Compared to video recording, computing logs have user interaction data already in structure and ready for data analysis without the hassal of human annotating. Log analysis is also easier to scale up to analysis of longer term interactions than video analysis (e.g. in Study One, we have 73 persons (25 teams), each person working for about 6 hours, adding up to 428 hours!)

3.3 CAnalytics Features

CAnalytics is a web application that aims to provide an integrated workspace environment wherein teams of analysts identify, visualize, integrate and assess facts from multiple sources. The interface is shown in Figure 3.2. The design is informed by our cognitive task analysis described beforehand. The system evolved over time. We tagged it as two versions, corresponding to the two classroom studies I have done. Most of the features are available in version one already, unless noted separately. Version two made several bug fixes, and added some a couple of new features, which we will discuss below.

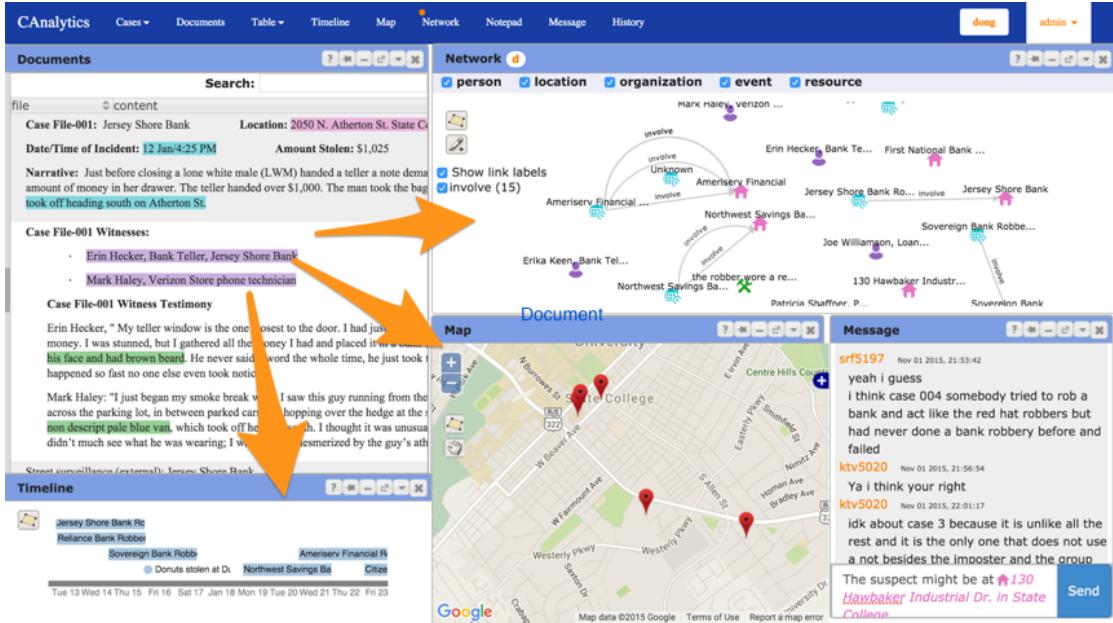


Fig. 3.2. CAnalytics user interface

3.3.1 An integrated workspace: annotation, visualization, and hypothesis

The CAnalytics workspace is conceptually composed of three functionalities: annotation for data modeling, visualization for data analysis, and hypothesis for hypothesis development. To make it an integrated environment, these functionalities share a consistent interface and data pool, and can communicate to each other in the same protocol. Each function resides in a self-contained floating window with a title in the top left, a set of functional buttons (help, fixed position, minimize window, maximize window, collapse window, and close window). Users can close a window when that functionality is not needed at the time of analysis to save screen space.

CAnalytics supports evidence annotation in the document view. The document view displays the text file each analyst receives. When a user selects and highlights snippets of information, a little icon pops up suggesting the text can be annotated.

Clicking the icon brings up the annotation editor (shown in Figure 3.3). The selected text is used by default as the name of the annotated entity, but can be modified by the user. The analyst can then mark the object as one of the following types: person, location, event, resource, organization, and a relationship. These entity types are reported to be the most frequently used by the Red Hat Lab and Colonel Jake Graham .

Annotations are “semantic”, meaning that users are not only highlighting plain text, they are also creating meaningful, structured data objects with attributes. For example, users can add such information as gender, job to a person entity. Users can also make reference to other entities; for example, users can name people who were involved in an event. Utilities are provided to assist in such input. Figure 3.4 shows the editor of location annotation. A place name-based search utility is provided to help find the place and geographic coordinates as users type. The location entity is then added to the right place in map automatically.

3.3.2 Multiple coordinated views of annotated evidence

CAnalytics supports multiple views for data visualization. Each view provides a means to schematize data, and thus an analytic perspective to understand data: Timeline organizes data by time, so analysts are able to do time-sensitive analysis, such as sequence of events and overlap between events. Map visualizes evidence by location; with that analysts can do spatial analysis, understanding the proximity of events. Node-link graph demonstrates the entities and relationships; connected entities are pulled closer by links and form into a cluster. This helps analysts easily decide the role of an entity: if an entity has a lot of connections in a big cluster, it is likely that it plays a leading role

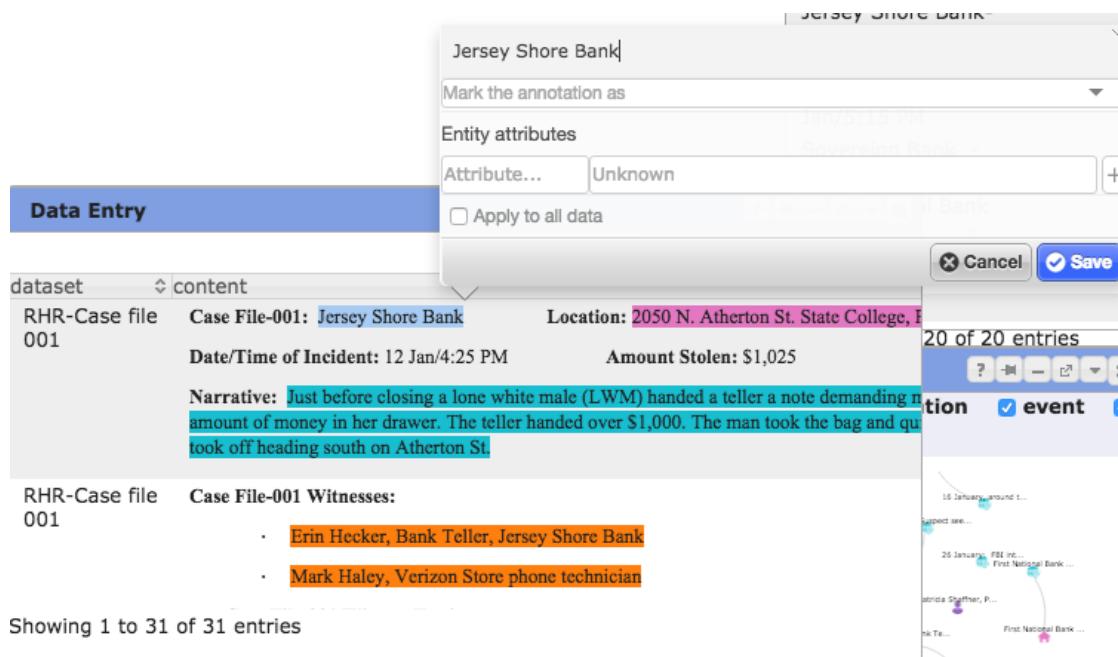


Fig. 3.3. Annotation in document view

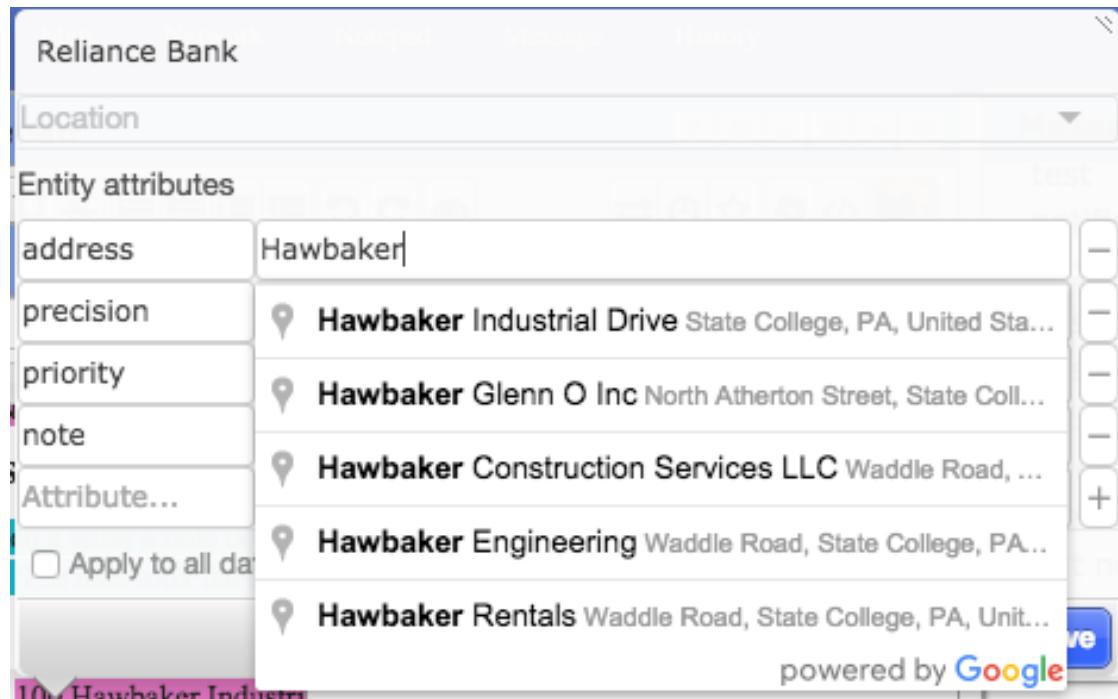


Fig. 3.4. Utility for location annotation

in the event. Finally, Table displays entity attributes and allows analysts to examine entities in detail.

These views are generated automatically from user created annotations. As shown in Figure 3.2, when an annotation is created in the document module, with attribute information about time, location, participants, and their relationships, a new event is created in the timeline module, a new location is created in the map module, and new people are created in the node-link graph module with a typed edges representing relationships among the people (or new edges are added to existing nodes). Different views are coordinated; that is, when users do a filter on a piece of information in one view, related information in other views will be highlighted. Filtering on different views/schematization provides different analytic strategies; e.g. timeline offers filtering by time, map offers filtering by location, and node-link graph offers filtering by related data objects. Based on default view of evidence, users can continue to schematize it. For example, users can drag the nodes in the node-link graph to make clusters. Users can also create a relationship in the graph by drawing a link between two nodes and input relationship attributes.

3.3.3 Information sharing and team awareness

Annotations are immediately shared with collaborators. CAnalytics supports real-time collaborative editing, similar to the Google Tools. Users can open several concurrent editors to collaboratively edit multiple annotations. Views are shared together with hypotheses. Team members will then be able to understand the hypothesis right in that visualization context. They can also leverage that visualization and continue exploring

by reorganizing the view, to challenge, confirm, or refine the shared hypothesis. In this way the team is leveraging each other's efforts and co-developing hypotheses beyond simply sharing. CAnalytics has a number of awareness features, including animation of data change, a notification system, a design we named “tool coordinator”, a message tool, and a history tool. As mentioned before, annotations and data objects created by collaborators are immediately shared, and updated in the views. An fade-in and fade-out animation in the views indicates the creation and deletion action by collaborators. A notification system sends individual's actions to the team, in the form of a text box in the top right corner of the workspace, keeping the team up to date with collaborator's actions. To reduce notification noise, instead of broadcasting whatever actions, we define a set of rules to determine which actions will be notified. For example, the action that creates an entity will be notified, but that re-layouts a view will not. Tool coordinator refers to a small indicator on the tool menu bar, suggesting who is working on the tool.

A message tool enables real time team communication. Chat history is persistent for traceability. We design a “mention” feature—users can refer to created entities and relationships in the workspace when they are composing a message. We believe this will improve communication efficiency because analysts are often observed to mention critical information entities in face-to-face discussion. When the message receiver hovers over the object, attributes of the object shows up, ensuring that the team is referring to the same thing.

The system maintains a persistent log of time-stamped individual activities. A history tool presents who did what to which object at when. Together with the notification system, users who work synchronously can be informed of others' activity continuously,

and be aware of the bigger picture of team's activity; users who work asynchronously will be able to use the history to reconstruct their work status and become aware of changes beyond the point of their last interaction. Different from awareness features developed in existing systems which are simply read-only text, the awareness information in our system is closely integrated with the analytic environment. For example, data objects that are changed and presented in the history view are clickable links. Users can hover over to read detailed attributes and click to do a filter on the object.

3.3.4 User activity log

We designed our log schema based upon Yi et al.'s Yi et al. (2007) and Guo et al.'s Guo et al. (2016) work. Our log focuses on high level activities that capture the task the user intends to accomplish rather than their mouse movement. Actions are categorized under the three top-level activities supported by the system: data modeling, visual analysis, and hypothesis development. More detailed actions, their definitions, as well as application-specific actions are shown in Table 3.1.

3.3.5 Collaborative hypothesis development

All the above features are included in CAnalytics for the first classroom study. In study two, we developed CAnalytics2. A major addition is support for a structured approach to support collaborative hypothesis development.

In study one, teams put down their hypotheses and notes in a shared note editor. The editor has similar features like Google Doc, in which partners can share text. In CAnalytics2, a hypothesis tool is developed as shown in Figure 3.5

Table 3.1. Specification of user activities

Activity	Action	Definition	Application-specific actions
Data modeling	Model	Create new data object	Create entity/relationship
	Remodel	Modify existing data object	Update entity/relationship
	Archive	Archive existing data object	Archive entity/relationship
Visual analysis	Elaborate	Show me more details	View details of entity/relationship
	Reconfigure	Show me a different arrangement	Adjust windows Rearrange views
	Filter	Show me something conditionally	Apply filter on views
Hypothesis development	Hypothesize	Create a hypothesis	Create a hypothesis and save the current view
	Validate	Validate a hypothesis	Validate a hypothesis and clone its view
Other			Write on notepad Send message Browse history

Entity attributes	
address	Hawbaker
precision	<input type="button" value="Hawbaker Industrial Drive State College, PA, United States"/> <input type="button" value="Hawbaker Glenn O Inc North Atherton Street, State Coll..."/> <input type="button" value="Hawbaker Construction Services LLC Waddle Road, ..."/> <input type="button" value="Hawbaker Engineering Waddle Road, State College, PA..."/>
priority	<input type="button" value="Hawbaker Glenn O Inc North Atherton Street, State Coll..."/>
note	<input type="button" value="Hawbaker Construction Services LLC Waddle Road, ..."/> <input type="button" value="Hawbaker Engineering Waddle Road, State College, PA..."/>
Attribute...	
<input type="checkbox"/> Apply to all data	

powered by

Fig. 3.5. Utility for location annotation

3.4 Implementation

3.4.1 Managing team analytic states

A particular challenge in implementing CAnalytics is an effective management of dynamic states that occur throughout the information analysis process. Rather than simply displaying a static set of data as in many visualization systems, data and views in CAnalytics keeps updating as analysts collect evidence, re-arrange views, and develop hypotheses. Each change generates a new “state” in the analytic system, which is like a “shot” of system dynamics that determines how a system looks like. Managing the dynamically changing states is particularly important in collaborative analysis as states of all analyst’s system should always keep synchronized, otherwise a coordination breakdown could occur.

Aligned with the three major activities in information analysis, we create state managers for each activity, as shown in Figure 3.6. A data manager is in charge of the raw documents, user created entities and relationships, and annotations that indicate in which part of a document is an entity created. A view manager controls the contemporary visualization state in the system. It includes a global view state and a local view state. The global state controls factors that affect all the views in system. For example, a filter applies to all views so that the same subset of data is shown across views. A highlight determines which entities are on focus and underscore them in their representation. A local view state manages how the data looks like in each individual visualization, including the position of the view in the workspace, the view extent, the zooming level, etc. Finally, the hypothesis state manager controls the current active hypothesis, as well as a path

of hypotheses it build upon. The path keeps track of development of hypotheses, and enables analysts to trace back.

Users interactions drive the change of system states. These interactions are modeled as events and controlled by a manager. Depending on the type of events, the system includes a log manager, which saves events as system logs and team activity history, and a sync manager, which broadcasts events to the other systems in the team.

After defining state managers and event managers, another challenge is to connect them to function as a dynamic system. A common implementation is event-driven: each system component manages its own state, and subscribes to events that change its state. This approach works for simple applications, but may be difficult to scale up in a complex system, especially a collaborative system in which there exist multiple sources that could change the system states. As shown in Figure 3.7a, a local component can be changed by both a local (the user in operation) and a remote interaction (teammate). In turn, the change of the local component could cause change to another local component, or even a remote component. For example, a brushing interaction on network changes the state of the network (highlight entity nodes). The change of network state in turn causes change to other views, for example, a reduced data view in the map showing related locations to those highlighted entities. The state could also be changed by a remote interaction when collaborators are sharing their view, and, similarly, cause change to other views. To some extent, the groupware developer no longer understands what changes a view, and the state of the groupware would become out of control.

To address the issue, we employed a state-centered approach to make state control more structured and clear. As Figure 3.7b shows, an event manager takes control of all

events, whether they are local or remote events. The event manager dispatches events to different event handlers and makes changes to a central state manager. Whenever the state changes, the state manager sends a signal to related interface components, notifying them to update the view. This approach centralizes the management of states, and the centered state becomes the “single source of truth”¹. All user interfaces (including the ad hoc user and teammates) subscribe to *the* state so that they will always be consistent.

Figure 3.8 demonstrates a typical event flow. An event is triggered by a user interaction. The event is sent to the Event Manager for processing. Depending on the nature of the event, the event manager may do three things. The event could change the local state directly (could be data, view, and hypothesis). The event is also sent to history manager. It is saved as system log for research purposes. There are also a set of rules that decide if the event is of interest to analysts. If it is, it is also saved as part of team history. Finally, the event is also passed to the sync manager if considered necessary. The sync manager then broadcasts the event to other systems in the team so that it can make exactly the same change on their state.

The system employs a standard three layer web application architecture. The frontend is the application running in browser that end users interact with directly. It is implemented in Javascript and HTML. We use PostGreSQL as the database. The backend server is implemented in Python/Django and Node.js. The Django server takes care of most REST API requests and reads and writes data into database. The Node.js is responsible for broadcasting data to systems in a team so that they can communicate with each other and keep their system state consistent. The tool takes advantage of

¹Single source of truth: https://en.wikipedia.org/wiki/Single_source_of_truth

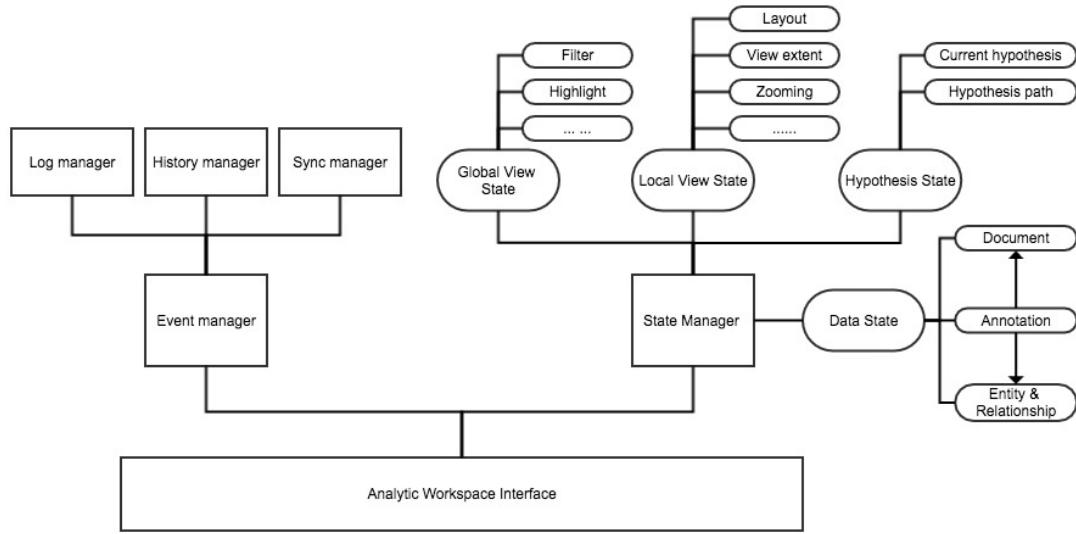


Fig. 3.6. State management in a dynamic system.

the state-of-the-art HTML5 WebSocket technique to realize real-time communication. WebSocket is a bidirectional, full-duplex communication channel that operates through a single socket over the web. It is part of HTML5 and provides a true standard for building scalable, real-time web applications. Since most modern browsers implement the protocol as a standard, no plug-in or other installation is required, which ensures convenient accessibility to users. We intentionally keep the API server and sync server separate so that they are loosely coupled. Each server works on their own (the Django server will be like a single-user application on its own, and the Node server can be potentially plugged into other applications to enable collaboration). This approach opens opportunity to develop the Node server into a *service* that provides a more general collaborative functionality to standalone applications in the future.

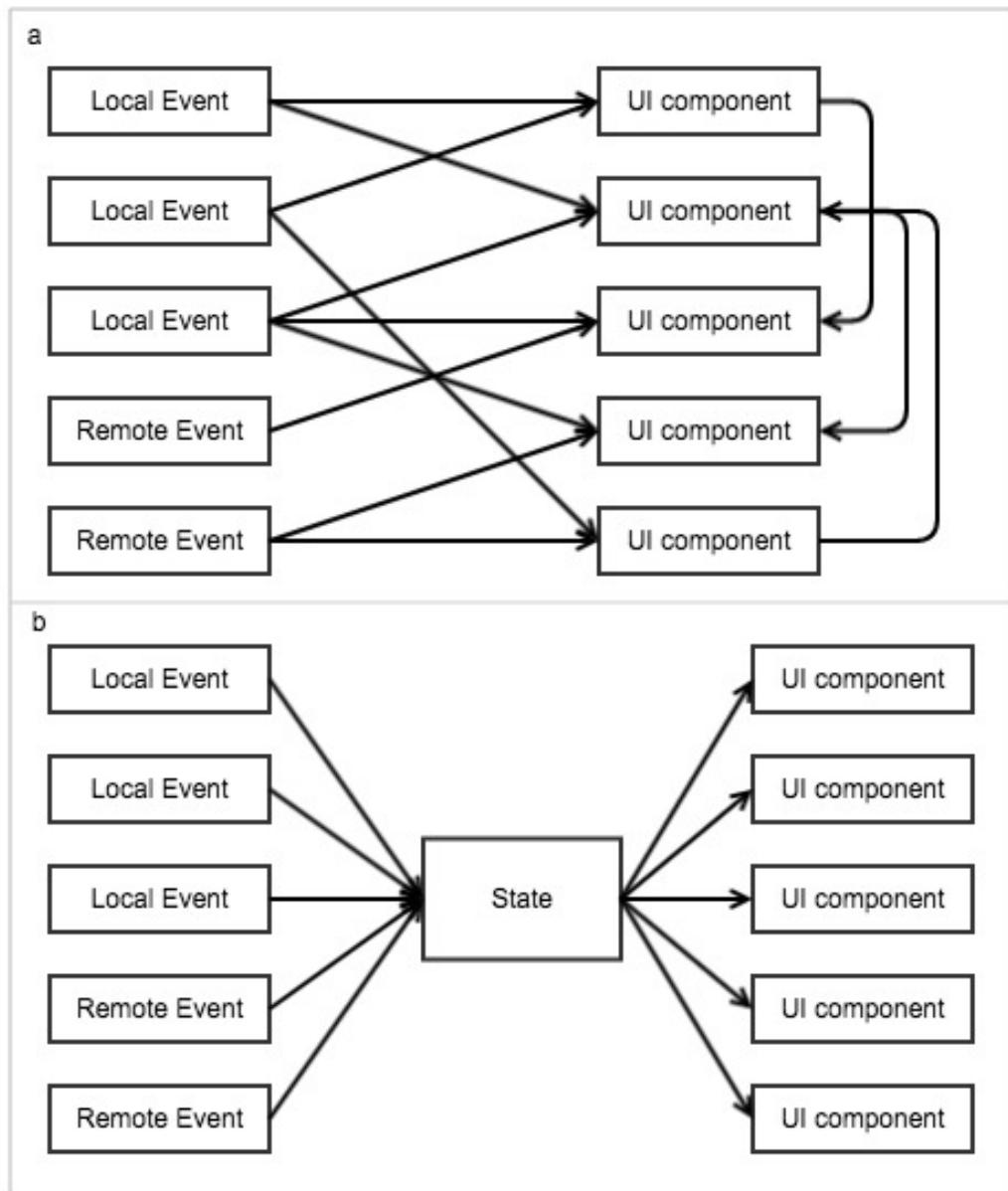


Fig. 3.7. Diagrams showing two state management approaches. a) Event-driven approach: each interface component manages its own state, and updates itself driven by events. b) Our approach: a central state manager takes care of all events and notifies interface components whenever there is a need to update.

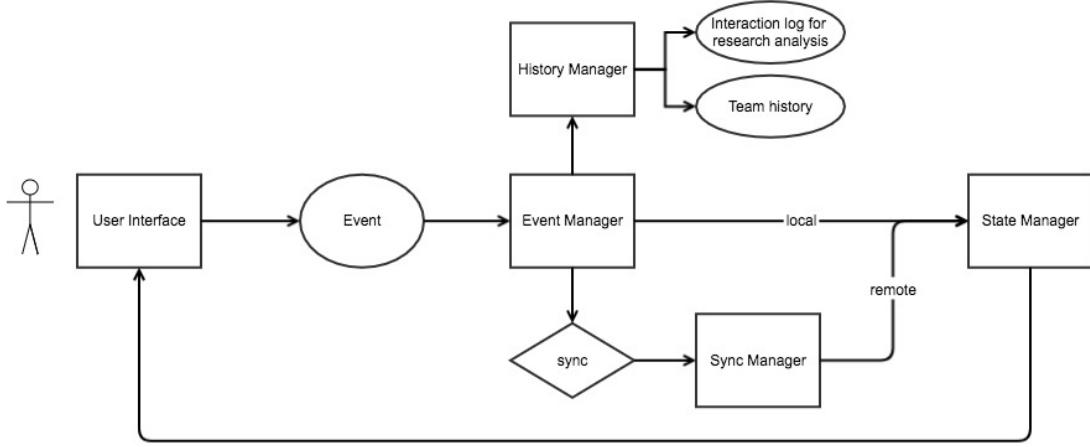


Fig. 3.8. Event flow in a collaborative system.

3.4.2 Conflict resolution

An important consideration in supporting collaboration is conflict resolution. A potential issue in CAnalytics is the addition of multiple annotations to the same text. CAnalytics supports multiple annotations by stacking them and marking the author and creation time of each annotation (Figure??). The same issue may exist when users add multiple relationships to two entities. The node-link graph in CAnalytics resolves the issue by displaying multiple labeled links between two nodes (as in the node-link graph in Figure 3.2). Therefore analysts can identify the potential conflicts and try to resolve them. It is also possible that it is not conflict at all. It may be two possible hypotheses due to uncertainty. In either case this approach helps analysts identify potential issues.

This conflict resolution approach is preferred in collaborative visualization systems [17, 18].

3.4.3 View restoration and sharing

Views in CAnalytics are handled as global views and local views. Global views refer to the window of each visualization, including which windows are open, the size of the window, and its position. Global views are saved and shared as JSON format.

Local views are more complicated as they include all the nodes and paths of visualization. To restore the view back to the same layout when the analyst reopens it, positions of all individual nodes are saved. This means a large amount of data will potentially be downloaded and parsed when individuals intend to restore an existing view. This is still fine (though it probably would take longer time), yet presents a challenge for real time sharing. If we share the whole view state node by node in real time, we need to pass a lot of data for frame, and several frames per second for smooth animation. Obviously this is not optimal. Instead of state sharing, we adopt event sharing. Only attributes of an event, for example, type of event (e.g. 'click', 'scroll'), target of the event (which node is the event applied to), are shared. When partners receive the event, the event is applied to the visualization. Since each event has a deterministic effect, the partner will see the same animation and result of the visualization as the event source. By sharing only event and ensuring event effects are deterministic, we make possible real time view sharing a smooth experience.

Chapter 4

The First Classroom Study

4.1 Problem Statement

Why we did this study?

1. To explore the feasibility and effectiveness of supporting information analysis with an integrated workspace.
2. To support activity awareness

This is the first classroom study to evaluate CAnalytics.

4.2 Setting and Method

The setting for this study was an undergraduate course in an intelligence training program in a US university. The program was designed to train students to become professional intelligence analysts. A key requirement of the course is to emphasize hands-on practice on team-based intelligence analysis.

During the first ten weeks of the course students learned several analytic techniques, including IEW (a technique to extract evidence and assess their values), ACH (a technique to evaluate multiple hypotheses against evidence), timeline analysis and network analysis, as well as state-of-the-art tools to implement these techniques, including Analyst's Notebook and PARC ACH. Students also practiced applying these techniques in two hands-on projects before our study. Students follow a typical “waterfall” workflow: they

start with IEW to extract evidence from documents and model them into a structured format. With that data available, they start building analytic artifacts such as an ACH Matrix in PARC ACH, and a timeline and a network graph in Analyst's Notebook. Sometimes the process is not necessarily sequential, and they want to jump back to edit data when building analysis. Yet since the tools are separate and data within the tools are not sharable, the annotation activity and analysis activity are largely isolated and streamlined.

Most tools in use lack serious collaboration support (except that some teams use Google Doc to construct an IEW table). Analysts were unable to contribute simultaneously (an issue known as production blocking Diehl & Strpebe (1987)). Students often divide their work by tools: each person picks a tool, creates and analyzes an artifact with that tool on their own, and puts individual analysis together towards the end. This had the consequence that findings and hypotheses be made without integrating collective efforts and diverse knowledge. Analysts coordinated work by manually sharing documents or graphs through email or cloud storage service (e.g. Dropbox), resulting in a scattered placement of results, requiring repeated manual resynchronizing to identify redundant or missing pieces of information, analysis of information, and analytic hypotheses. In summary, the students in our study had learned and practiced with tools lack of support for collaboration, and were all aware of the shortcomings.

Of the 98 students enrolled in the course (from two sections), 73 consented to participate in the study. All of the 73 students held major in the program of Security and Risk Analysis. Most (75%) of them were in the third academic year ($M = 3.06$ years, $SD = .44$), indicating that participants in our study had relatively advanced experience and knowledge

in intelligence analysis. Participants' age ranged from 19 to 28 ($M = 20.4, SD = 1.18$). 77% of the participants were male.

Students were given a tutorial on CAnalytics a week before the project began. One of the authors walked through the features of CAnalytics, without enforcing or implying any strategic use of the tool. Students then accomplished a small case analysis at their own pace. During the week of the actual study, one author was constantly available to help with any technical issues. Although students were encouraged to make full use of CAnalytics, to ensure a naturalist environment students were always free to employ any other tools that they believed useful.



Fig. 4.1. Classroom setting

Our study began on the 10th week of the course and lasted for one week. The analysis that students performed on our tool was the investigation of a series of bank robberies, all of which was fabricated by the course instructor. Teams were provided a set of documents pertaining to seven robberies, including police reports, witnesses reports, video records, and news media. The analysis was designed to be open-ended, meaning

that there was no single, definitive answer. The instructor explained that the task was to simulate real world scenarios, in which analysts always reasoned in the circumstances of uncertainty, ambiguity, and complexity. The instructor told the students that he expected the analysis to take around 6 hours, including in-class and outside-class work. In the end of the project students were required to submit a team report, describing their hypotheses, assumptions, conclusions and supporting evidence. When in class, students used a 27-in Macintosh desktop (as shown in Figure 4.1. Students could use any equipment when outside class.

Students were randomly assigned into 25 teams (23 three-person teams and 2 two-person teams). Research suggested that group size be an important factor in group collaboration, thus two-person teams might behave differently from three-person teams. We thus excluded data from the two-person teams in our analysis in this paper. Also, from the log (and confirmed by their questionnaire), we found that one team made little use of CAnalytics and opted for other tools (Google Doc). Hence their data was also excluded. Thus in this paper we reported the result from 22 teams.

We employed several data collection approaches. We administrated a post-study questionnaire. The questions used 7-point likert scale and included 9 items measuring individual's self-reported awareness, 7 items for communication quality, 6 items for collective efficacy, and 3 items for perceived performance (Convertino et al. (2011)). We also used NASA-TLX Hart & Staveland (1988) to measure cognitive load. The questionnaire also included open-ended questions asking how the tool helped or impeded their work. We captured user interactions with system logs. Instead of simply logging low-level events like mouse click and keyboard strokes, we recorded actions such as

creating an annotation and deleting an entity. Finally, we reviewed team reports and graded them as an indicator of team performance. Since the task was open-ended, there was no single right answer. We constructed an assessment rubric together with the course instructor by listing all possible hypotheses and evidence from the documents, with a full score of 16. The first author and a research assistant graded the reports independently. If the grades differ by less than 2, an average is set as the final grade (14 out of 22 reports). Otherwise (the rest 8 reports), the two graders review the reports together and make an agreement.

4.3 System

The system, CAnalytics, used in this study includes all features described in Section 3.3 except for the hypothesis tool. To sum up, CAnalytics includes an integrated workspace for data annotation and visualization, and enables real time sharing of user created data objects. It also includes a shared notepad for team note and hypothesis.

4.4 Results

Over the week, teams created 1805 entities and 1529 relationships in total. The number of entities teams created ranged from 24 to 223 ($M = 82, SD = 39.9$), and the number of relationships ranged from 7 to 237 ($M = 69.5, SD = 51.0$), showing a large variety. The big range was related to different team data modeling strategy, which will be discussed in detail later. While teams could work on the project any time, we found three intensive usage sessions from the interaction log: two were in class and one was before the team report deadline, outside of class.

CAnalytics was generally well received by the students. An overview of the related survey items (shown in Figure 4.2) shows that students positively rated all aspects of the tool except cognitive load, towards which they had a close to neutral feeling, which is understandable as the tool does not necessarily reduce the complexity and load of the task.

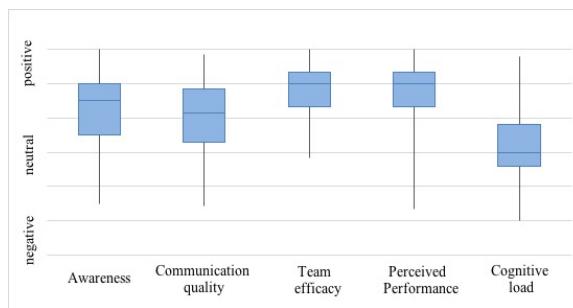


Fig. 4.2. Survey responses (box shows Q1-Q3 and median; ends of whiskers show maximum and minimum)

4.4.1 Filtering vs. Accretion

In examining participants' created data objects, we noted a distinction between filtering and accretion strategies, similar to what was reported in the paper prototype study Carroll et al. (2013). Filtering is selectively modeling of data and adding to an artifact. Users must decide what information is relevant, and thus what is to be excluded, as well as what granularity of information is to model. Filtering requires more team coordination, because teammates must reach a common ground of the current problem as well as information needed to answer the problem. Figure 4.3a is an example of network built with filtering strategy. It only represented key information of robberies.

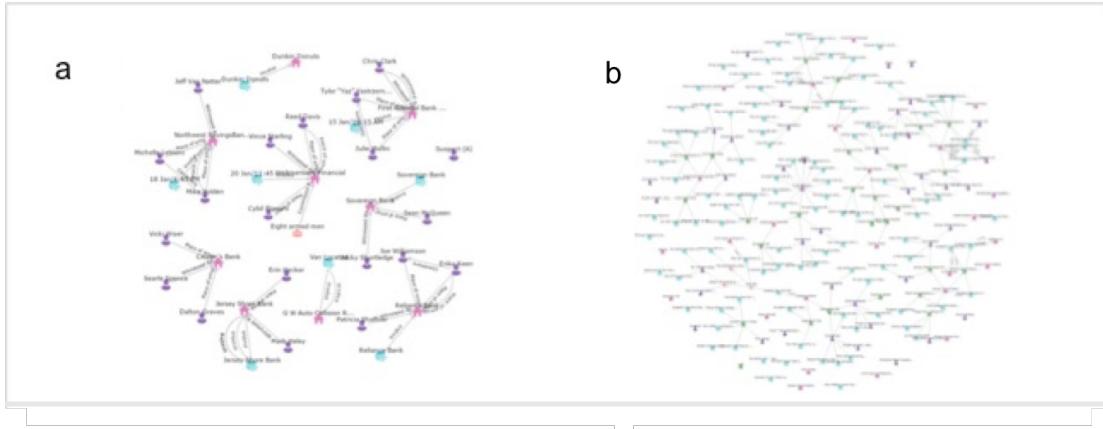


Fig. 4.3. Network artifact comparison: filtering (a) vs. accretion (b)

Accretion is an attempt to comprehensively represent the problem by adding all information to an artifact. Users extract every fact from the document, regardless of its immediate relevance to the problem. Accretion requires less coordination as it is relatively mechanical note taking. A disadvantage of accretion is that it can be time consuming to model all details and the produced artifact could be fairly complex. An example is Team 108, who modeled every step the suspects took, which resulted in far more entities (223) than the average (82) and more cluttered network view (Figure 4.3b). This accounted for the large range of entities created. Users also realized the problem. They reflected that they spent too much time in detail events, and many did not help their analysis at all:

I felt that after we were done annotating, we hadn't really accomplished anything and that we were no closer to solving the case than when we had started. In the end it didn't really help that we had annotated the data. (P86)

4.4.2 Situational logic vs. Comparison

We examined the patterns of the annotations teams created. We saw clear distinction in terms of how teams divided their work in creating annotations. Seven teams created annotations with a document-based pattern: each teammate focused on their own set of documents and made concrete detailed annotations. (as shown in Figure 4.4a). Individuals consider the known facts of the current document and seeks to identify the cause-effect relationships of this document. We take the term *situational logic* from Heuer Heuer (1999) to describe this pattern, in which teams start hypotheses “with concrete elements of the current situation, rather than with broad generalization that encompass many similar cases” (p.32). The team relied on partners to identify and share useful information within their assigned documents. The failing of this strategy is thus in case an individual fails to identify or convey evidence in the document, the team may overlook or misunderstand the information altogether.

Four teams followed a different strategy. Instead of dividing by documents, they divided work by entity types: each individual went through all documents but only focused on and annotated entities of certain types, e.g. teammate A only annotated persons and teammate B annotated locations (as shown in Figure 4.4b). This is called a *comparison* strategy Heuer (1999). When analysts recognize two entities as comparable, they use the known entity to fill gaps in their understanding of present entity. They build connections between two entities, make analysis of similarities and distinctions. Different from situational logic, comparison interprets the present evidence in the context of precedent elements in other documents, and they could examine entity patterns across

multiple documents simultaneously. The challenge is of course to decide whether two entities are truly comparable. There is a tendency to conclude two entities are equivalent simply based on the fact that they are equivalent in some aspects. When individuals fail to convey the assumptions made in the comparison, it could mislead the whole team reasoning.

The rest (eleven) of the teams did not show specific labor division patterns. Teammates read and annotate the same document. This could be challenging traditionally but is possible because the tool enables them to see new annotations by others in real time and build on top of other's annotation. Figure 4.5 shows an example where one team worked on the same document simultaneously. The three team members exhibited high synchronicity in which document to analyze, which we believe was not by accident. This has the advantage that teammates are always on the same page and can discuss hypotheses throughout the analysis process. A problem with this approach is scaling. Since each teammate is required to analyze all documents and entities, and constantly switch between individual work and collaboration work, it demands a lot of individual efforts. As data size increases and analysis time tightens, the team may not afford this approach.

4.4.3 Inferential connection vs. Factual discrete

When examining the network artifacts participants created, a clear distinction exists in the shape of clusters. Networks from eight teams consist of discrete clusters (Figure 4.6a): Nodes within a cluster are connected. They are factual entities and relationships extracted from one single robbery case. Each cluster is self contained,

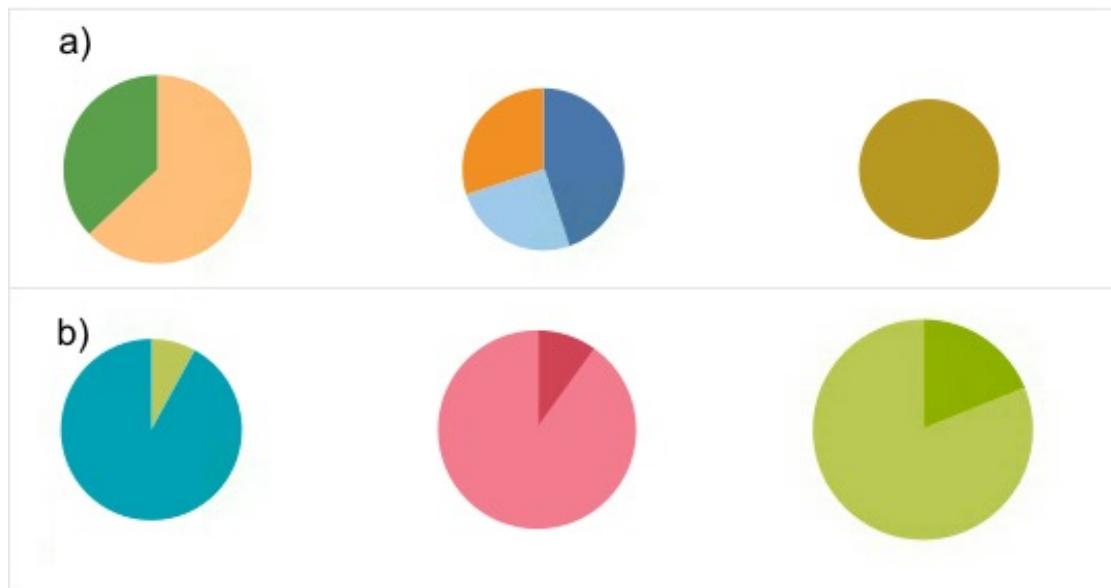


Fig. 4.4. Pie charts showing different labor division strategies. Each pie chart corresponds to one team member. (a) Document-based collaboration. The pie chart shows the documents one team member annotated, color coded by document ID. (b) Entity-based collaboration. The pie chart shows the entity types one team member created, color coded by entity types.

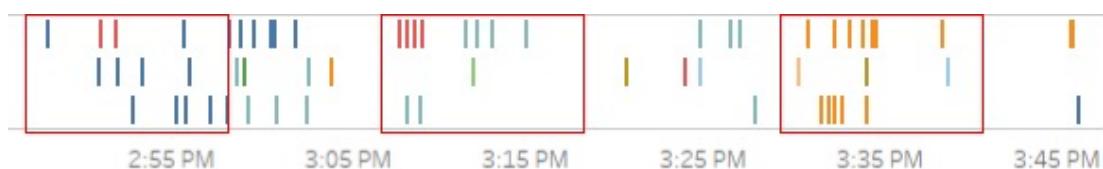


Fig. 4.5. Graph showing the timeline of one team creating annotations. Each row corresponds to one team member. Each bar represents an annotation, color coded by document ID. The red blocks highlights the periods when all teammates worked on the same documents simultaneously.

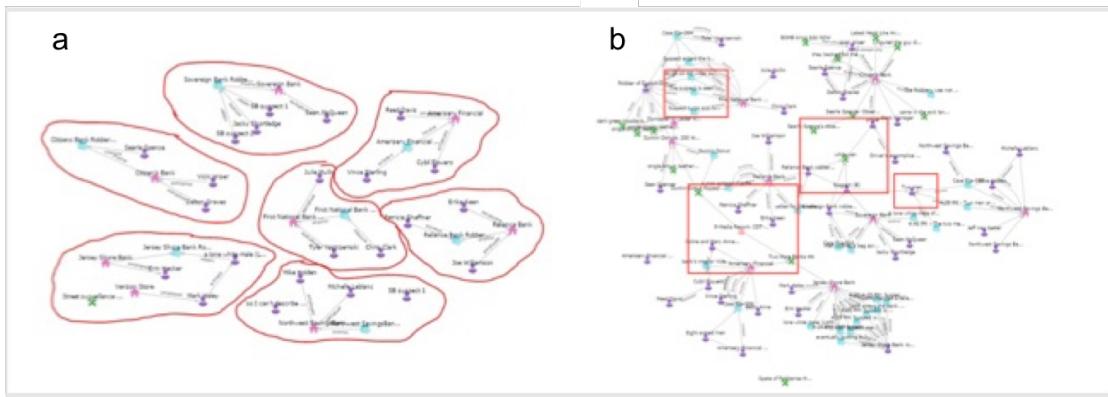


Fig. 4.6. Network artifact comparison: separate clusters (a) vs. connected clusters (b). The parts highlighted in red squares in (b) are key evidence that connects clusters

with no links in between. In contrast, six other teams created networks that consist of connected clusters. Nodes and links within a cluster are similar to ones in the discrete cluster, but these clusters are connected through an inferential link (Figure 4.6b, marked in red in the graph).

The differences in network artifacts relate to how teams annotated and represented entities in documents, especially in circumstances of incomplete, ambiguous information. For example, we have the following document snippets:

Case 1: Suspect is shown running down N Atherton and jumping the passenger side of a white van

Case 2: [...] we all watched as the guy ran around the parking lot. I went to the door and saw him jump into a white van of some type

In both cases suspects are associated with “a white van”. Yet with only that information we are not able to conclude the white vans are the same one. That’s the *uncertainty* analysts must consider, and must properly represent. We see three types of

annotation in team's network artifacts, as shown in Figure 4.7. In figure a, the vans are treated as two separate entities with the same name. It keeps their relationship open to question, yet analysts might have difficulty in identifying this hidden link in later analysis, especially when the network scales. In figure b, analysts merge the two vans in a single entity. This representation makes the connection apparent, yet misses the uncertainty information, which could mislead teammates. We found some teams using the notepad tool in CAnalytics to complement that the link is hypothetical. In figure c, vans are perceived as two entities but connected with a link, which is often labeled with text like "hypothetical".

The three representations reflect different strategies in handling uncertainty. In figure a, analysts are more "negative", or conservative, toward uncertainty. They avoid bringing in any hypotheses during the stage of annotation, so as to avoid any bias confirmation in analysis. In contrast, analysts in figure b are most "positive" with uncertainty. If they believe the inferential connection is a high possibility, such representation helps them identify important patterns and make quick decisions. In general, figure c appears to be more reasonable. It represents the connection while preserving the uncertainty, yet requires more annotation efforts.

For example, However, it does not mean these teams believe these cases are irrelevant. In fact, these teams still discussed connections between robberies in their report. It turned out that these teams documented possible relationships in the notepad tool in the form of free text, separate from the network graph. that is, these teams distinguished information content and synthesized them in different artifacts.

In contrast,

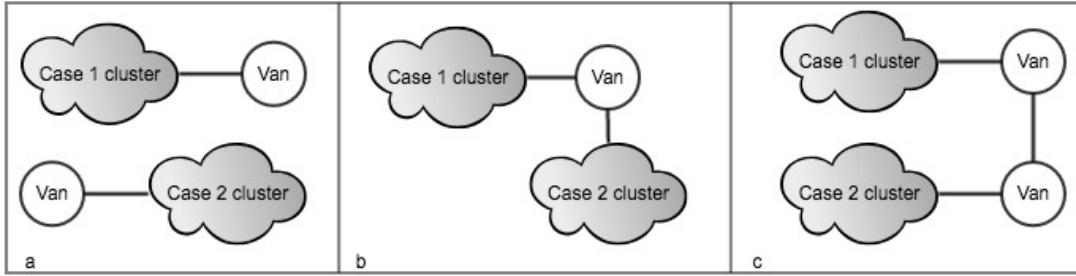


Fig. 4.7. Three different representations of uncertainty for entity relationships

By comparing these two types of networks, we found that links within a cluster were typically *factual* relationships modeled from raw documents (e.g. a white van was witnessed at a location), and links between clusters were often *inferences* beyond literally documented (e.g. a white van at location A is the same van witnessed at location B). Teams creating separate clusters represented only facts in the network and held evidence of uncertainty in a separate artifact. One advantage of distinguishing facts and inferences is that teams can always be aware of assumptions made when making a conclusion. And since all inferences are held in one place, teams are forced to confront them and review their uncertainty iteratively in the process. However, the strategy also adds difficulty to analysis as analysts may overlook or fail to combine evidence scattered in different artifacts.

On the contrary, in connected-cluster networks, facts and inferences overlaid in one artifact together drive the layout of the network, are better synthesized, and give analysts a clearer picture in one place. Teams may discuss and evaluate the level of uncertainty of inferences to decide whether to add them to the network. This strategy makes analysis more interactive among teammates: they need to negotiate, evaluate, and

reach consensus on the value and validity of inferences. However, a problem with mixing facts and inferences is, to some extent teams might forget whether a link is factual or inferred, and ask whether conclusion derived from the visualization can be trusted under uncertainty.

4.4.4 Interleaving data annotation and analysis

We examined the pattern of data modeling and analysis first by qualitatively looking at a visualization of the entire interaction log (e.g. Figure 4.8a shows one team's interaction). All teams worked intensively on data modeling as they started off the project. This was the phase when teams were getting themselves familiar with the documents and made initial annotation input into CAnalytics. Starting from certain time point, all teams started analysis on visualizations, followed by frequent transitions between data modeling and analysis. 11 teams started data analysis in the first usage session, while the other 11 teams had this transition in the second usage session. In average, the transitions occurred in 47.6 minutes after the project began. The earliest transition occurred in 14 minutes after the team started the project, and the last team had the transition around 104 minutes, later in the second session. We also found performance difference among teams that started analysis early and those late. Teams that started analysis in session one had higher performance ($M = 8.6$) than teams that started from session two ($M = 6.7$), although the difference was not statistically significant.

The fact that participants returned to making annotations after analysis indicated that they did not wait to start analysis till they had finished modeling. Indeed, the activity of data modeling and data analysis were highly interleaved throughout the project

(as shown by the interleaving color bar in Figure 4.8a). Participants switched from one activity to the other activity frequently. The state transition diagram (Figure 4.8b) demonstrates the interleaving in an aggregated way, in which we encode the number of transitions as width of the link. This result confirms our design expectation that data modeling and analysis should not be supported as separate staged activities, and that an integrated environment should streamline the workflow.

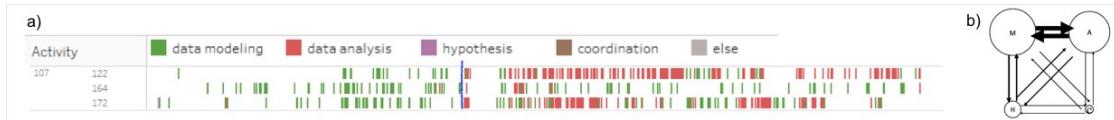


Fig. 4.8. (a) Visualization of interaction logs of Team 107. Each row of colored marks indicates the sequence of top level activities a participant performed. (b) State transition diagram of interaction logs of Team 107. Each node is an activity, whose size represents the time spent on the it (M: data modeling; A: data analysis; H: hypothesis development; C: coordination); a link represents a switch from one activity to another, whose width encodes the number of switches. We see highly frequent transitions between data modeling and data analysis

4.4.5 Collaboration and awareness

One recurring theme in the subject feedback we collected was that the collaboration features were helpful for solving the problem. In the survey 88% of the students positively rated their group awareness. Participants appreciated that the tool complemented traditional analytic tools, describing CAnalytics as Analyst's Notebook with real time synchronization features similarly to Google Docs, or described it as Google Doc with added visual analytics. To quote one participant, "*CAnalytics is like an analysts notebook*

that multiple people could work on at once [...] an analysts version of a Google Doc” (P65).

Participants reflected that they could now contribute simultaneously without concerns of interference and could have everything in one place instead of manually sharing documents via a cloud service.

It was much easier to coordinate as a team with CAnalytics because we could all work on the same system at the same time. Without CAnalytics, we were forced to do the work separately and compile all the work onto one system after we had finished. (P156)

Students also reported that being able to see teammate's status made the task more motivating and engaging:

During class I wasn't sure if my teammates were doing work for that class or another thing but then seeing their dot [tool indicator] switch between applications on the software and updates pop up on my screen I knew they were doing work for 231. (P141)

The fact that you can see what other teammates are doing and they can see what you are doing creates a sense of accountability in terms of separating the work load. (P51)

The motivating effect of awareness might account for, at least partially, the fact that teams were participating equally. We measured the equality of participation in terms of number of created entities and time spent on CAnalytics. We refer to Olson Olson

et al. (2017) in calculating equality: one minus the variance of proportions times 100 (for better readability). Thus the score ranges from .67 (when only one person contributes in a three-member team) to 1.00 (when everyone participated exactly equally), and higher score indicates higher balanced participation. The resulted equality of created entities and time was .96 and .99 in average respectively. This indicates participants contributed fairly evenly.

Another repeated theme was the awareness features helped assure all teammates were executing the team plan. Participants reflected on their experience that a common team breakdown was misunderstanding of the team plan, and that they did not realize the misunderstanding until everyone had spent significant efforts finishing their “perceived” job. CAnalytics made plan execution assured because they could always see where teammates were working and what they were generating; and if anything unexpected happened, they could communicate immediately rather than in the end of the project.

Participants reported many other instances of awareness they realized using CAnalytics. We categorized them based on the element of awareness, or the essential problem of awareness of *what* Schmidt (2002), into social awareness, information awareness, action awareness, history awareness, and intention awareness, as shown in Table 1.

When asked what features helped them stay aware of team activities, 28 participants mentioned the tool coordinator, 24 mentioned the notification system, 19 mentioned the history tool, 14 mentioned the real-time update of user-generated data, 12 mentioned the collaborative editor, and 7 mentioned the message tool. Although the number of mentions does not simply indicate tool usefulness, it suggests users were explicitly aware of these awareness features and appreciated their support.

Students' positive feedback on awareness was further corroborated by interaction logs. For example, we measured the number of entities accessed by other collaborators. While data generated by users is automatically shared, it is up to collaborators whether to read/edit the shared information or ignore information altogether. The log showed that on average, 77.6% of the created entities were *read* by at least one other teammate. Further, We measured how many entities were *edited* by collaborators, a phenomenon we argue requires higher awareness, because the collaborator must not only realize the creation of the entity, but also understand its content. We defined *peer editing*, manipulated as the ratio of editing other's entities over editing those created by oneself. We found that all teams edited collaborator's entity objects, with a peer editing value equal to .83 ($SD = .45$). The result suggests that teams had little difficulty accessing and modifying partner's created data objects.

One major critique is the lack of support for sharing intermediate analytic insights. An insight is revealed and contextualized by a specific arrangement of views, e.g. a reduced data view of interest through filter, a highlighted entity representing the analytic focus, and a clustering layout of network to demonstrate a specific relationship. While teammates share the same data pool, they are likely to have different views of data, and thus different *interpretations* toward the data. A dynamic view together with its interpretation represents user's intermediate analytic status. Sharing these insights could inspire team analysis Gotz & Zhou (2009). With CAnalytics participants complaint that they could not easily make such communications. The team could "be looking at the same information but arranged in completely different ways" (P131).

4.4.6 Interaction with team performance

To systematically evaluate factors that influenced team performance, we conducted a multiple linear regression between team performance as the dependent variable and team collaboration characteristics (i.e. equality of entities and time respectively, peer editing, number of entities created, shared entities, switch time from data modeling to analysis) as independent variables. We treated a team as a system Henman (2003) and modeled team-level relationships. The relationship of individual level variables (e.g. survey items) could not be simulated with regression directly because data within a team was interdependent. The analysis was performed using *R* R Core Team (2016).

The result is shown in Figure 4.9. The regression model was found to be statistically significant ($F(5, 16) = 4.58; p < .01$), with an R squared equal to .59, meaning 59% of the variability in team performance was explained by the model. Five team-level variables contributed significantly to prediction of team performance. The model suggested that balanced contribution of entities predicted higher team performance scores ($\beta = 45.79, p < .05$), but balanced participation time did not show such effect. More peer editing led to better performance ($\beta = 3.89, p < .01$), implying that analysts should be encouraged to model data as a team, and to remodel other's entity objects as needed. Longer elapsed time before a team started analysis (*switch time* in the figure) predicted lower performance ($\beta = -.05, p < .05$), which suggested that teams should shorten the time of pure data modeling and start analysis earlier. Larger number of entities teams created also predicted higher performance ($\beta = .03, p < .05$). This can be interpreted that teams providing sufficient model input did benefit from system support. However, we are not ready to

claim more entities are always better because we should also be cautious that entities irrelevant to the team problem bear no value to team analysis, as discussed beforehand. Somewhat surprisingly, the number of shared entities negatively predicted performance ($\beta = -8.99, p < .05$). We considered this might also relate to the issue of entity value: sharing entities without relevance to the team problem leads to reduced team efficiency and creates distraction, which in turn would lead to worse team performance. Another possibility is that teammates may not necessarily access all entities partners created. The theory of Transactive Memory Wegner (1987) suggests that teammates do not have to know everything in the team but should know “who knows what”. Individuals specialize in their own field (e.g. one specializes in event analysis while another in social relationship examination) and reach for collaborators when other knowledge is needed. In such case, common knowledge in shared entities would actually be redundant. Individuals do not need to access content of entities created by others, but only keep aware of the creator.

4.5 Discussion

Participants did have concern for possible duplication. As one participant complaint, “*we could not actively see the changes our teammates were making until well after they had made them*” (P46). This was because an annotation was shared only *after* it was created, yet another teammate might already be drafting an annotation on the same text snippet in the meantime.

The goal of the study is to understand and support an interleaved, collaborative workflow in information analysis by evaluating tool usage in a natural environment over multiple usage sessions. Our work responds to calls for an integrated environment

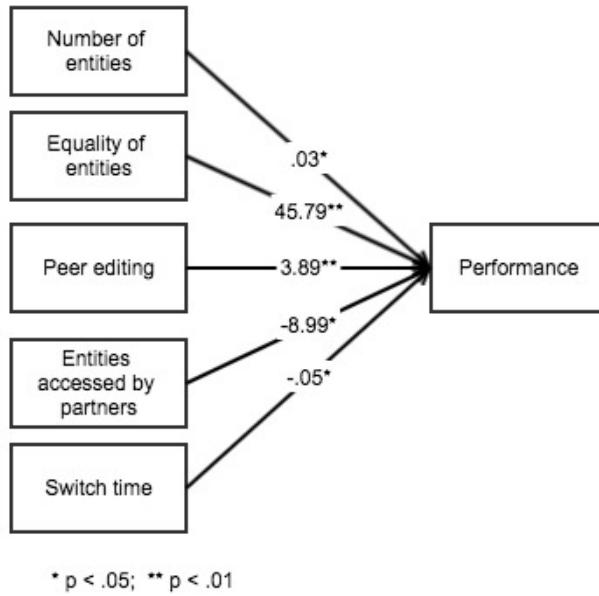


Fig. 4.9. The relationship between collaboration characteristics and team performance.

made in the intelligence community as well as other data intensive domains Shah (2014); Chen et al. (2016); Director of National Intelligence (2008); Amershi et al. (2015); Ware (2012). The system builds upon prior empirical studies (e.g. Carroll et al. (2013); Borge et al. (2012); Kang & Stasko (2011); Chin Jr et al. (2009)) and embodies their design implications in our tool. The study also complements research that only tests tools in short term lab studies (e.g. Convertino et al. (2011); Goyal & Fussell (2016); Hajizadeh et al. (2013)) by investigating tool usage over multiple usage sessions.

This study adopts an *analyst-centered design* approach. A critical requirement of developing tools that meet user needs is to understand their needs and practices. When these needs and practices are specialized (as is the case in intelligence analysis), it is particularly important to include the target user population in the design process Scholtz

& Endert (2014). Our classroom study provided an opportunity with deep access to analysts in training. These analysts have been trained with knowledge and skills of intelligence analysis, and have experience with state-of-the-art analytic tools such as Analyst's Notebook and PARC ACH. In their reflections, participants often compared CAnalytics to those tools. Their multi-session usage of CAnalytics also allows them to adapt to the tool and learn to appropriate it to the best of their team purpose Stahl (2006). Therefore, while their feedback is admittedly not the same as an experienced professional, their feedback does provide a deeper insight into strengths and weaknesses of CAnalytics.

The study provides encouraging results. Participants appreciated an integrated environment where they could share raw documents, evidence snippets, views of evidence and hypotheses in one place. They liked the fact that they could contribute simultaneously without blocking or interfering each other. Another llaborative tool is to keep teammates aware of each other's activities. Staying aware of teammates not only helps establish a common ground for planning team strategy, but also ensure everyone is following the plan as expected. Moreover, results suggest the awareness features provide positive *social facilitation* Zajonc (1965): individuals found the task motivating and engaging with awareness of each other's activity. We also measured collaboration characteristics that impacted team performance, and found that balanced contribution, peer editing, and earlier switching from modeling to analysis predicted higher team performance. Most importantly, we documented the interleaved workflow enabled by the integrated environment, and explored momentum in modeling and analysis behaviors that drove the

activity switching. Below we discuss design implications that could potentially enable a better collaborative experience in information analysis tasks.

4.5.1 Scaffold a structured interleaved workflow

An often misconception about information analysis is that data modeling and data analysis are two staged activities. This is akin to the *waterfall* software development model, which features a sequential process that flows downwards through the phases of requirement conception, software design and implementation, and testing and maintenance. Critics have argued against this approach and instead called for an iterative design process that leads to reframe user requirements, redesign, redevelopment, and retesting.

Our result demonstrates a similar iterative and dynamic process in information analysis. The result is striking especially because our participants have been trained with tools that impose a waterfall model. They could have followed their old waterfall practice with our tool, yet instead all teams spontaneously switched to an iterative manner. We projected that modeling on multiple granularities drives analysis on different levels, and uncertainty in analysis pushes analysts back to collecting additional data.

Realizing that, we probably could shape analysis into a more interleaved workflow with a more structured approach. Annotation is a process in which analysts are constructing their own version of information space on the basis of information provided by the document. The resulted representation of annotation will strongly influence their analysis. Meanwhile, analysis generates hypotheses which will frame how analysts perceive their data. The gap between hypotheses and facts then drives analysts to annotate a new version of information space. Structured techniques such as IEW and ACH help users

perform analysis in a systematic and transparent manner in each well defined activity, yet fall short in guiding analysts in switching between the two activities Kang & Stasko (2011). Our system implies a structured modeling through annotation and a structured analysis by visualizing data in multiple views. By sharing the same data structure and consistent user interface, we enable a smooth switching between the two stages. Our result implies the role of multilevel modeling and analysis uncertainty in driving the switch. Based on that, there exist design opportunities to enable a more interleaved flow. For example, we can build a more structured scaffolding process to bridge modeling and analysis. When user adds a new data object, the system could suggest possible connections to existing evidence in the context of an appropriate level of views, which is likely to help analysts discover new patterns. When a user creates a new hypothesis with uncertainty, the system could highlight associated evidence, which would prompt the analyst to re-examine the data and look for more data. Such scaffolding provides a basic structure to link stages of analytic activities that analysts can take on without imposing a specific fixed workflow.

A smoother interleaved workflow could also be made by increasing team awareness of partner’s modeling and analysis. That is, uncertainty in one’s analysis not only motivates oneself to data modeling, but also drives partners (who is aware of the what and why of the uncertainty) to collect more data. Similarly, one’s modeling could influence and drive partner’s analysis, given the partner is fully aware of what is modeled and how the new data connects to the level of existing data. Such “team-level” interleaving could make teamwork more interactive and close coupling, but also requires support of higher awareness, especially of hypothesis uncertainty and data model context. Improved design

for multi-granularity modeling, uncertainty representation, as well as team awareness of these features opens up possibilities for coordinating interleaving at the team level. We discuss these design implications in further detail in the following paragraphs.

We noticed iterative data annotating activities, which indicates the need for analyst in control of data modeling process. Depending on the questions and hypotheses in analyst's mind, an analyst may want to model different types of evidence. Therefore the analyst should always be in control of evidence modeling, while the computational method only *facilitates* data modeling, e.g. ease data input, hide complexity of format structuring.

4.5.2 Represent uncertainty in collaborative analysis

Our finding suggests that uncertainty in collaborative analysis requires deeper understanding and brings more design challenges and opportunities than in traditional data visualization. Uncertainty is prevalent in data. For example, evidence is uncertain depending on its source (witness reports vs. police report) and collection means (video vs. narrative). Most research efforts focus on representing uncertainty from data itself Pang et al. (1997); Zuk (2008), but teammates become a source of uncertainty as well in case of collaborative data analysis. Collaborators share annotations and hypotheses based on the data and knowledge of their own, and such shared information is derived with uncertainty. Yet it is often easier (and people are more inclined) to represent information of certain, but those of uncertain tends to be lost or vague. These propose new research questions: how

to allow and encourage analysts to represent uncertainty? How to represent collaborator-generated uncertainty together with data-bearing uncertainty? Shall we distinguish them or treat them equally?

Further, there may exist nuance in understanding of uncertainty among teammates. Collaborators make their own evaluation of uncertainty, based upon their unique knowledge, experience, role, as well as trust in the source teammate Chin Jr et al. (2009). Thus it is important that teammates are aware of existence of uncertainty, and reach a team-level agreement of the information gap. Graves Graves (2000) characterized intelligence analysis as a progression where the analyst's certainty grows over time. And we can probably describe collaborative information analysis as a progression where a team of analysts establish awareness of certainty, and achieve a growing team agreement of certainty.

While uncertainty poses a big challenge to analysts, we also perceive it as a design opportunity, especially an opportunity for a new round of team collaboration. Uncertainty emerges and that means a new need for teams to communicate and establish awareness. We should represent uncertainty in a way that is more discoverable and, and more referable in their communication. Teams in our study spontaneously employed two different approaches: to represent uncertainty on top of data, and to separate uncertainty from data. In fact the research community does not have an agreement whether it is better if uncertainty is visualized or suppressed, or under what conditions it is better Maceachren et al. (2005). We should probably provide users with control in deciding how to display uncertainty. For example, users can *filter* by uncertainty so that they can choose to consider only evidence of certainty, or to review all inferences and decide what

extra data is needed to consolidate them. We should design a richer graphic language to encode uncertainty into the analytic view Bertin (1983). Teams need a channel to refer to, discuss, and consolidate uncertainty.

4.5.3 Build collapsible views for multi-level modeling

We observed that analysts built data models in multiple granularities and coordinated among multiple levels of details. For example, analysts jumped from digging into details of a single event, to comparing between two events, and to overviewing all robberies as a complete story. When sharing these data, a critical requirement is to represent them in an appropriate context in order to ensure teammates understand them correctly. A collapsible data view could be a solution to accommodate such multi-level modeling. This can help draw teammate's attention to a specific item while keeping the global context available. Analysts can focus on a certain level of detail at a time while conveniently switching between levels. A collapsible view also reduces the problem of cluttered view when data volume increases. and when analysts dig into greater details (e.g. representing suspect's all actions to identify patterns of common actions in two robberies). An analyst can overview all robberies and only unfold detailed actions when investigating into a specific robbery.

4.5.4 Share views as team resource

View sharing is important for sharing and understanding analytic result Morton et al. (2014a), and provide an anchor for team communication. A common approach to view sharing is to enable a public view which always keeps synchronized for all teammates

Convertino et al. (2011); Greenberg (1990). However, a single public view does not meet the need in dynamic information analysis. Analysts often share multiple views in an iterative workflow. Most views are for evidence keeping, and some views are more important that may be repeatedly used in a report. Other views may be reused by teammates and further developed.

We propose that views should be treated as a *team resource*. Just like data, views can be sharable, extensible, and reusable. For example, an analyst can save their current view as a shared resource when they feel it useful to collaborators. Other people can reuse the view to their need. To facilitate the reuse of views, Nobarany et al. (2012) designed a recommendation system to help discover possible useful views from teammates. Shared views should be interactive rather than static images, so that analysts can perform all interactions including filtering and highlighting, and are able to evolve the view with collective team efforts, a critical requirement emphasized in Carroll et al. (2013).

A public workspace to which partners can refer sets up a common ground for discussion. Synchronous groupware used to impose “what you see is what I see” (WYSIWIS) (e.g. Stefik et al., 1987), in which all team members share the same view. However, this design blocks production as only one member is able to control the workspace in one turn. Relaxed-WYSIWIS is thus proposed in contrast to the previous strict-WYSIWIS, allowing people to navigate the workspace independently Gutwin & Greenberg (1998). Indeed, in observation of empirical teamwork, participants were found to keep private notes while contributing to shared note-taking Borge et al. (2012).

4.5.5 Distinguish visible vs. valuable contributions

Finally, we noted cases where teams created far more entities than needed with an accretion strategy. Strikingly, while similar data modeling strategy was reported in the paper prototype study Carroll et al. (2013), users with CAnalytics seemed far more tempted to accretively add information, with far more entities and cluttered views. For example, the extreme team created as many as three times entities than the rest teams in our study, much more than the difference in the paper prototype study. Why did this happen? We guess both the context of classroom study and the system design contributed. Unlike in the lab study where teams are temporarily assembled, teams in a class evaluate peers either consciously or unconsciously and value how themselves are being evaluated. Such social pressure motivate individuals to make contributions, and indeed to make *visible* contributions, more than *valuable* contributions. That is, participants noticed that their work activity was visible to their partners, and accordingly prioritized doing more visible work over doing less visible work. In some cases, this led to a new problem of easy and less valuable contributions that were highly visible - such as creating and therefore sharing data entities that were not particularly important, and subsequently made data models seem cluttered. For example, creating and therefore sharing an entity gets immediately notified to the team whereas weighing the importance and relevance of an entity goes silent in the system. We need to investigate approaches to making significant contributions more visible, or perhaps making it more immediately visible that less important contributions are indeed less important.

Chapter 5

The second Classroom Study

5.0.1 Problem Statement

Study One presents a positive impact on team behavior of an integrated analytic workspace and a structured approach of data annotation and visualization. On top of that, in Study Two, we want to explore impact of a more structured approach to collaborative hypothesis development.

Analysts generate multiple hypotheses and find evidence to confirm or disconfirm them. A common technique to evaluate hypotheses is Analysis of competing hypotheses (ACH) Heuer (1999). ACH emphasizes a rational, transparent process of hypothesis analysis. The steps include identifying a complete set of hypotheses, identifying all pieces of evidence, assessing the diagnostic value of each evidence against each hypothesis, and drawing conclusions of the likelihood of hypotheses. ACH guarantees an appropriate analytic process and increases the chance to avoid common analytical pitfalls such as confirmation bias. In practice, however, we found students rarely came up with a complete set of hypotheses in the very beginning. Instead, they began with reading documents and got familiar with the suspects. They would propose a hypothesis in the process of extracting and marshaling evidence. They would share their hypothesis with teammates or document it in a note. With evidence accumulating, the team collaboratively refine the existing hypotheses, and propose alternative hypotheses. The process was iterative, with

hypotheses constantly being refined and evolved. ACH demands a high expertise bar for analysts. Users are required to identify a complete set of mutually exclusive hypotheses at the very beginning, which is difficult for people with little expertise and experience in the domain. Further, hypotheses tend to evolve as analysis proceeds. A hypothesis valid in the beginning may no longer be of value later, or two seemingly separate hypotheses in an early stage of analysis could be combined in a way to better explain the situation later on. The assumption ACH makes that all hypotheses and evidence are identified and set in the very beginning limits the dynamic evolvement and development of analysis.

We believe that analysts should not only share their hypotheses, but should also collaboratively develop and refine their hypotheses. Structured techniques such as ACH tend to treat hypotheses as complete, well-defined objects. Analysts decide a set of alternatives and evaluate evidence against them. In reality, however, hypotheses tend to change and evolve as analysts collect more information and gain deeper understanding of the situation. Hypotheses that might be valid at the beginning of a process, may come to be inappropriate, or incorrect as the analysis progresses. While simply sharing hypotheses is beneficial to analysis teams, its positive impact is limited to an extent as it is not completely leveraging the various expertise of a team. Where instead, the combining and evolving of hypotheses from and by multiple members of the team (where the evidence that the hypothesis is based on is included in the hypothesis object) creates an environment where the team is more than just a sum of its parts.

To enable collaborative hypotheses development, we propose four functions: 1) Connect and ground hypotheses to evidence.: In our current system we have focused on making the entry of data user friendly and seamless to the analysis of the data itself.

The process of entering data has been tested and verified by users who were training as data analysts. We will build on the structures that emerge from this process to lay bare the provenance of each developed hypothesis. In this way, others can trace the decisions and assumptions that are usually invisible and implicit to hypothesis generation.

- 2) Sharing the evidence marshaling map: In our system, the evidence will be marshalled into a map for each hypothesis. This enables teammates to keep track of the analysis as it is progressing and they will have immediate access to the hypothesis team artifact.
- 3) Ability to reuse and build on teammate's evidence marshaling map: This enables analysts to be able to further develop the hypotheses without duplicating efforts.

5.0.2 CAnalytics@v2: supporting collaborative hypothesis development

We developed a hypothesis development panel. When the user creates a hypothesis, the current state of views are automatically captured and saved, and bound to the hypothesis. Thus each hypothesis includes a message and a link to the view. The view includes the data, filters applied to the data, as well as state of visualizations, so that the view could be restored to what it was like when the hypothesis was created. The hypothesis is shared with the team immediately. Teammates could restore the views and investigate into the hypothesis in context. Since the view is interactive rather than a static image, partners can continue exploring the view and develop a new hypothesis based on the last one, whether supporting or refuting that. The system builds a thread structure based on how the view of hypothesis is “inherited”. For example, if the view used in Hypothesis A is re-arranged on the base of another view of Hypothesis B, A and

B are in the same hypothesis thread. However, if A and B are using different views, or analysts believe they are different stories, A and B will be in two threads.

Based on feedback from Study One, we made several improvements on CAnalytics. A major change is to enable a structured approach to hypothesis development, as described in Section 3.3.5. We also add animation on change of visualization. For example, when a partner creates or deletes an entity, the entity fades in or fades away in the visualization, prompting user's awareness of change.

facilitate follow-up investigation. As information is analyzed it often yields new questions and hypotheses requiring additional filters and searches. Enabling hypothesis sharing and restoration allows analysts to build upon teammate's effort, either disconfirming their hypothesis or gaining additional insights.

We used CAnalytics2 for this study. One major improvement is the feature of collaborative hypothesis development. Instead of a general shared notebook, CAnalytics2 includes a more structured approach for hypothesis composition. Analysts can also bind the ad hoc visualization with a hypothesis and share with the team, which can be restored and interactive. We have covered the features in detail in Section 3.3.

Other than the sharing hypothesis feature, CAnalytics2 includes several other improvements, including:

Timeline schematization Timeline is an important tool for displaying events. Yet the timeline in study one contains too much information to be useful. Based on user's feedback, we further schematize events by people (Figure 5.1), so that analysts can easily examine availability of each suspect during critical event period. Given that the scenario

spans a long period of time, we adopt a overview and detail visualization technique so that analysts can zoom into event details while having the big picture in view.



Fig. 5.1. The new timeline displays events schematized by people

Capability to restore deleted data 3. Making the archive concept more salient
(change label delete to archive; items archived are greyed out in table)

Animated change 4. Animation: change by teammates (adding and deletion of entities and relationships) are faded in or out

When one user is deleting an object, collaborators can easily miss that person's intention to delete the object and even the deletion itself. One design solution is to make small actions larger, in this case, by making the delete action more prominent. This could be done by adding a visual atop the deleted object to emphasise that it has been selected, and then adding an animation to the actual delete event to make it more salient, where it perhaps grows in size for a few seconds before shrinking to nothing [@Greensberg2006]

5.1 Use case demonstration

5.1.1 Classroom study setting

We deployed CAnalytics in SRA 468, a course in the major of Security and Risk Analysis at Pennsylvania State University. The course was designed to teach senior undergraduate students about application of visual analysis in the domain of intelligence analysis.

The study began in the later phase of the course, when students had learned about common techniques of visual analytics, such as time series analysis, tree-based data analysis, and network analysis, as well as popular visualization tools like ArcGIS and Tableau. With such equipment of basic visualization knowledge, from Nov 7 to Nov 18, we introduced CAnalytics and asked students to use the tool to solve a complex task.

Students were first given a tutorial, in which one of the authors guided students step by step to use CAnalytics features to solve a simplified task. The students then had two full classes to do the project, each class lasting for 50 minutes. Due to the complexity of the task, students were asked to work for extra hours outside class.

The task used in the study was a fabricated laptop theft scenario. The task was applied and validated in earlier user studies Carroll et al. (2013); Borge et al. (2012). The task consists of 222 pieces of information, involving social media data, bank transactions, suspect interviews, and other records. Teammates were randomly assigned one of the three roles and were responsible for a unique set of documents: a Web Analyst had social media data and would contribute mostly on social relationships; a Record Analyst had archival information about suspects' daily schedules and could contribute by providing

locations of suspects when theft occurred; and an Interview Analyst had interview data from people of interest and could contribute by identifying the motivation of suspects. To reach the right conclusion, teams had to share their unique information sufficiently. Students were told not to share documents directly but they could share their information by creating data annotations in CAnalytics.

Of the 42 students enrolled in the course, 30 consented to our study and 21 (7 teams) finished the study (submitted reports for both phase 1 and phase 2 task). In our analysis, we focused on the 7 teams that completed both tasks. Of the 7 teams, one team (Team 163) in their post-task questionnaire reported that they ended up using Google Sheet to schematize data instead of CAnalytics. Therefore we excluded their interaction log in the analysis, but kept their performance as a baseline reference for other teams.

5.2 Result

5.2.1 Confirming result from study one

Many feedbacks from students in study two echo and confirm results from study one. For example, users like the feature that maps out annotated data objects to different visualizations. This allows them to put various perspectives into the otherwise unstructured text documents.

We feel that this service was helpful in that we could visually see the relationships that we mapped out as well as add our own parts to it without needing to email them to each other or having multiple networks. (P178)

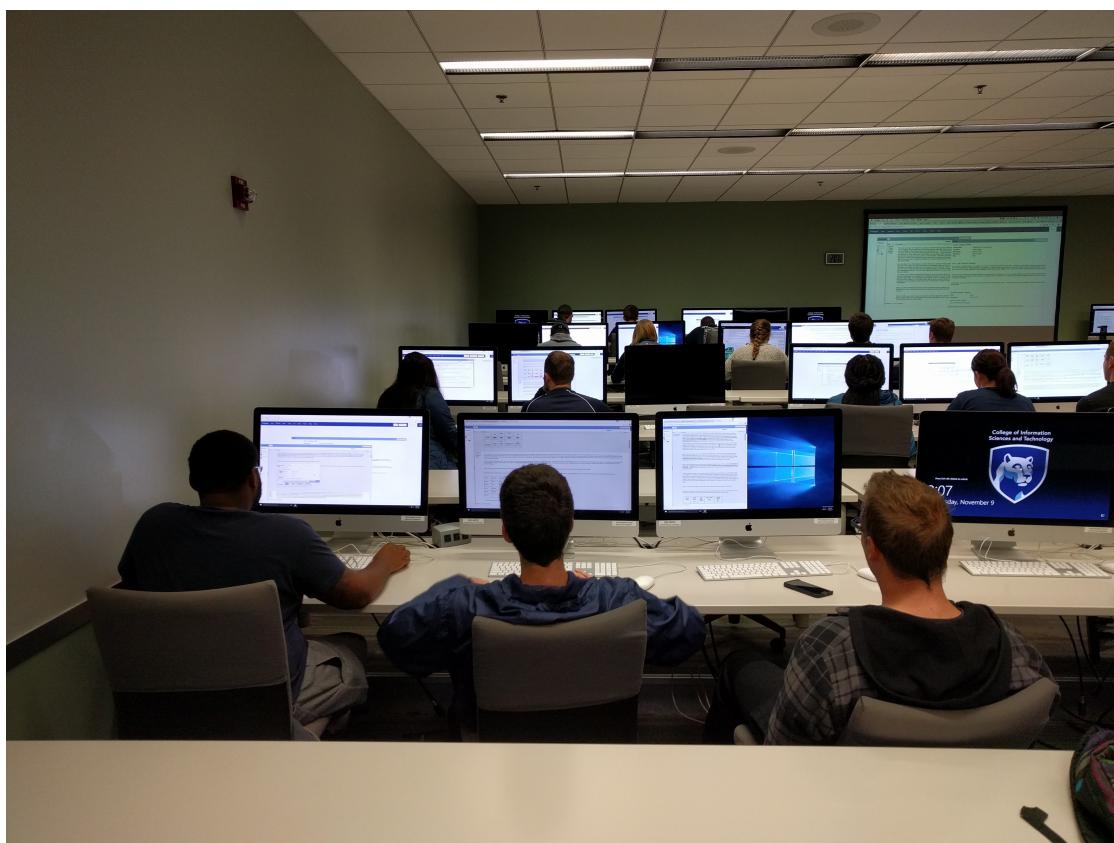


Fig. 5.2. The second classroom study setting

Users also mentioned usage of the new timeline. They appreciate the capability to match individual events against critical events; thus they can quickly target at data of interest. The multiple tracks in timeline also encourage them to use the filter feature.

It helped because we were able to see who was where at what times so when the crimes were taking place we knew who was near the crime or who was busy etc. (P150)

The main tool used to help eliminate subjects was by using the filter tool to see what suspects were not at the location of the thefts at the times for each incident (P188)

Users rate highly of the collaboration features. They feel the idea of sharing one common analytic artifact change the way they collaborate.

This differs because I can actually see what my team is doing as they are doing it, but we can also alter the information and that would change the entire network. Changing the network and altering information also updated and altered my own information, which was something new compared to the other tools I have used

One user also comments that he would first check his teammates' update before he sets out for his own analysis.

Whenever I would work, I would go to the network first to see the connections that had been made. (P186)

Similar to result in study one, students frequently associate CAnalytics with Google Docs, and think of it as the Google Doc for analysts.

CAnalytics tool is quite helpful when compared with other information analysis platforms such as google docs because it helps in real time annotating of various persons, relationships, events etc. It also is a good tool when multiple people need to work on the same document and since it provides updates it helps reduce redundant work. Also, it provides additional features such as Timeline, Map, Network to help analyze information. (P193)

It gave us the ability to create and see the visuals that we would have only been writing about in programs like Google Docs. (P178)

5.2.2 Feedback on hypothesis development tool

Feedback on the structured approach for hypothesis development is mixed. We observe two major ways teams use this features. First, they use it as a space for insight curation. As individuals go through the annotation and visualization, they create notes about their hypotheses, findings, or ideas.

The hypothesis was nice in the sense that it was a clean space to put final thoughts (p100)

As we went throughout our work we noted any important information and relationships that we found in the raw data to create hypotheses (P183)

These ideas then become a resource to which they refer when they compose their final report. This can be a useful feature as it helps analysts to focus on high level hypotheses instead of relooking at underlying data and visualizations.

I read through the hypothesis and used that to form my answers to the questions.

(p183)

Second, participants use it to narrow down analysis, and direct the team to focus on analyzing a particular aspect. When one team member create a hypothesis, partners look into that hypothesis, retrieve the context where that hypothesis was generated, and help validate it connecting with their own knowledge.

It helped us focus and narrow in on an event and suspect. (P191)

I confirmed the results by filtering out the maps, networks, and tables to confirm the conclusion. (P188)

When my teammate shared a hypothesis about where some of the suspects work, I responded by using that hypothesis and applying it to my own analysis to see if I can make a connection between the hypothesis and the theft. (P198)

On the other hand, some participants comment that the way a hypothesis is generated does not follow their workflow. They reflect that when they find something interesting during their analysis, they would share it immediately with teammates (verbally).

We all agreed with each other when we posted hypotheses because we worked together the whole time (p100)

We wrote it together so we didn't really respond to it. (p150)

The team discuss about it and come up with a hypothesis together. In other words, the team work closely in a synchronous manner to generate a hypothesis that everyone would agree on. This is in contrast with the model our tool design was based on, which encourages frequent turn-around in developing a hypothesis; each turn-around would be more *asynchronous* between which teammate would spend time assessing validity of the last hypothesis. This closer approach of generating hypotheses seems prevalent as several teams mention the mismatch between the tool design and their workflow. This is in contrast with their feedback about data annotation and visualization, for which the tool supports a similar structured technique. Participants mostly like those features as “they reduced the efforts to frequently keep updated with the team”. We will extend exploration into this contrast in the discussion section.

One student mentions scalability issue. When the team creates too many hypotheses, it becomes less efficient to continue developing these hypotheses.

*we enjoyed making hypothesis together, and the tool quickly got out of hand
when we all just tried tossing in info (P204)*

It is true that the tool currency does not include any feature to search or manage hypotheses yet. The display is also not optimized for large number of hypotheses. The goal of this study is to explore feasibility of such a structured hypothesis development approach. But it will be an interesting future research topic to investigate how to enable large scale, long term hypothesis development and management.

A few participants also mention the discoverability and usability issue of the feature. The feature is only accessible under a secondary menu in the system, thus it is not straightforward to start using the feature. The success of the tool also depends on how the whole team is using it. If one made a hypothesis but never got a response from the team, they gave up.

My team did not share a hypothesis, I created a couple but I did not get a response from my team. (P186)

It was useful to see what they were indicating, but I still had to call them over to fully explain their rationale and logic. It was a good start point, but without them explaining in person to me what patterns they were seeing I was having difficulty making the same conclusions. (P204)

Interestingly, one participant did not use the feature but reflected that they probably should.

We never completed a hypothesis within CAnalytics because it did not occur to us to do so. However, looking back, it might have been helpful to solve the case. (P197)

We do not know what makes him/her think that way, but research has observed such behavior before that might account for the reason. A structured techniques demands more efforts as users must explicitly externalize their thought, which takes more time than verbally communicating with teammates. Externalization, however, in the long run, benefits team as they have an artifact for team reference.

5.3 Discussion

The design goal of CAnalytics v2 is to support higher level collaborative sense-making while confirming the positive effect of integrating low level data annotation and visualization efforts established in study one. Information analysis can be roughly divided into three key levels Pirolli & Card (2005); Kang et al. (2014): (1) identifying critical evidence; (2) organizing evidence into a schema; and (3) generating hypotheses based on evidence schema. Accordingly, CAnalytics provides data annotation, data visualization, and hypothesis threading tool for each level. Study one has suggested positive impact with integration of data annotation and visualization. Study two aims to provide a linkage between hypothesis development and data visualization, and investigates feasibility in a collaborative setting.

While collaboration occurred throughout the project, we observed that the degree of closeness in which they collaborated varied depending on the phase of analysis. When they set out the project, they usually had a close discussion on how they were going to tackle the problem strategically and how they would divide their work. This is often known as *strategic coordination*, or functional collaboration Kang & Stasko (2011). CAnalytics does not have explicit supporting feature, but we do observe participants discussing over the messaging tool.

The team usually then divided up the work and worked on their part individually. They still collaborated by sharing evidence they annotated, but the collaboration was more loose because it was mostly *content* sharing without a need for deep discussion. The real time sharing feature provided by CAnalytics helped facilitate collaboration.

We observed a stronger demand for close collaboration in hypothesis development than in data annotation or visualization. We speculate that collaboration in higher level sensemaking requires higher volume of information to be exchanged. While data annotation activities share only a piece of annotation, hypothesis development requires sharing of a rich amount of evidence and schema at a high frequency. And based on the evidence, there can more than one direction analysts can draw from. The team needs a more synchronous, closer channel to exchange their ideas. This is probably why they would prefer to discuss over hypotheses together and compose hypothesis collectively.

This nuance in collaboration closeness suggests us to consider accounting for the different needs in our design for groupware. Teammates in loosely coupled collaboration are weakly dependent on one another and can function autonomously, often without the need for immediate clarification or negotiation with others. Yet closely coupled analysts have patterns of collaboration that distinguish them from other types of teamwork, and groupware systems that are designed to support both couplings must address these differences.

Compared with ACH

Comprehensive hypothesis validation vs. opportunistic hypothesis discovery ACH requires all hypotheses to be developed in the beginning, which could be challenging to beginners.

With computer supported tools, teams can keep track of hypotheses, and build upon each other. Analytics is treated as a course throughout which hypothesis is created, rejected, and recreated, and refined.

ACH requires analysis of diagnostic value of each piece of evidence. Tools always respects uncertainty, and considers analysis as a process of reducing uncertainty. Tool helps capture uncertainty and identify it as opportunity for collaboration.

Managing meta data generated during hypothesis development

Chapter 6

Reflection on Evolution of CAnaytics: A Design Research

6.1 Reflection on method

6.1.1 Classroom study

A critical requirement of developing tools that meet user needs is to understand their needs and practices. When these needs and practices are specialized (as is the case in Intelligence Analysis), it is particularly important to include the target user population in the design process. Intelligence analysis is a specialized, domain-specific task and it is important to understand analysts with domain knowledge, learn their practice, and observe their react to tool features Scholtz & Endert (2014). However, professional analysts are very hard to include in a long-term design cycle due to confidentiality and security issues. Our classroom study provided an opportunity with deep access to analysts in training. These analysts have been trained with knowledge and skills of intelligence analysis, and have experience with state-of-the-art analytic tools such as Analyst's Notebook and PARC ACH. In their reflections, participants often compared CAnalytics to those tools, as well as the different teamwork experiences. Therefore, while their feedback is admittedly not the same as an experienced professional, their feedback does provide a deeper insight into strengths and weaknesses of CAnalytics. In addition, the students are young learners that are willing to employ new work practices supported

by features in tools. They are important parts of the future intelligence community. In some sense, their practice can be treated as a view into the future of practice of the community Martin (2014).

Our study spanned multiple usage sessions. This allowed participants time to learn to adapt to team functions and to appropriate the tool to best serve their team purpose. Teams were able to explore different team strategies, and to make changes if they got stuck Stahl (2006). For example, we noticed that two teams decided to change the use of the tool halfway in their analysis. One team started with dividing work by case documents, but later decided members should annotate different entity types. Another team started with an accretion strategy by annotating all entities. Later they discovered that this strategy brought too much noise, and decided to clean out irrelevant entities (filtering strategy). Such change occurs as a consequence of increased awareness of team functions and tool capabilities, which takes time to develop.

With deep access to these analysts, we are able to interpret and triangulate our data from multiple sources. We qualitatively analyzed participants' feedback to understand their first person experience, and corroborated them with surveys and interaction logs. The interaction logs provided us a detailed view into their analysis process, and allowed us to quantitatively characterize their team behavior. Finally, their analytic products, including the visual artifacts and team reports, helped us distinguish team outcomes.

Yet classroom study also has limits. For example, many factors and variables could exist that affect team performance. The fact that these factors are often impossible to model or control adds to the difficulty in data analysis (e.g. teammate absence). Also, data collection is challenging because team interactions are not always accessible. Teams

can choose to work synchronously or asynchronously, and it is difficult to predict when or where the interaction of most interest is to occur. Verbal communication is not accessible, which could be useful to infer team awareness as a complement to interaction logs. Our work identifies both positive evidence and problematic situations, and propose potential solutions and possible hypotheses, yet rigorously evaluating these solutions and validating hypotheses is beyond this study. Lab studies and case studies can be conducted in the future to address these problems with greater control and deeper data access.

6.1.2 A design research of groupware

Groupware research has a lot of different aspects from research on individual tools, and bears its unique challenges. Groupware research involves dynamics between multiple individuals, and such dynamics is heavily shaped and mediated by the tool. The tool and the team function like a single system in accomplishing a task. Therefore, unlike single-user tool research in which we can mock up some features to explore user's reaction, groupware must be fully functional, with full capability to support team communication and coordination. The team dynamics cannot be simulated.

Team dynamics also take time to evolve. This includes development of social awareness, situation awareness, workspace awareness and activity awareness. Such awareness needs time to be established and maintained to a level where team can start work effectively. This is also different from single-user research, in which user reaction to a tool can be mostly measured immediately. A longer term research is preferred if we want to investigate the real impact of a groupware. And again, to enable a longer term investigation, the groupware must be fully featured.

Teams behave totally differently with tools of different levels of collaboration support. Teams with paper prototypes naturally do more synchronous, co-located collaboration than teams with computer prototypes supporting distributed work. To some extent, teams with paper prototype are accomplishing a different *task* than teams with computer support. Tasks people engage in or wish to engage in define requirements for artifacts. Those artifacts, in turn, open up new possibilities to accomplish the task, while, inevitably, introduce new errors and complexities in task performance. Again the new tasks devolve into requirements for further technology evolution, provoking another round of transaction. This cycle, known as task-artifact evolution (Figure 6.1) ?, drives the underlying rationale of my research on groupware. I aim to develop a tool to drive the task evolution in possible directions, in an attempt to explore the design space of groupware.

I take a *design research* approach ? to explore how interactive visualization could support collaboration. The goal of design research in HCI is to address the “wicked problems” ? in real world and the increasing complexity of situations that needs software support. The outcome of a design research is not only artifacts of more effectiveness but also new knowledge that can inform future design. Therefore, the developed groupware is not only a team supporting *tool* that facilitates their intelligence analysis projects, but, and arguably more important, also a research *instrument* that reveals team behaviors and patterns in that specific setting, and ultimately contribute to the knowledge pool of design rationale for groupware.

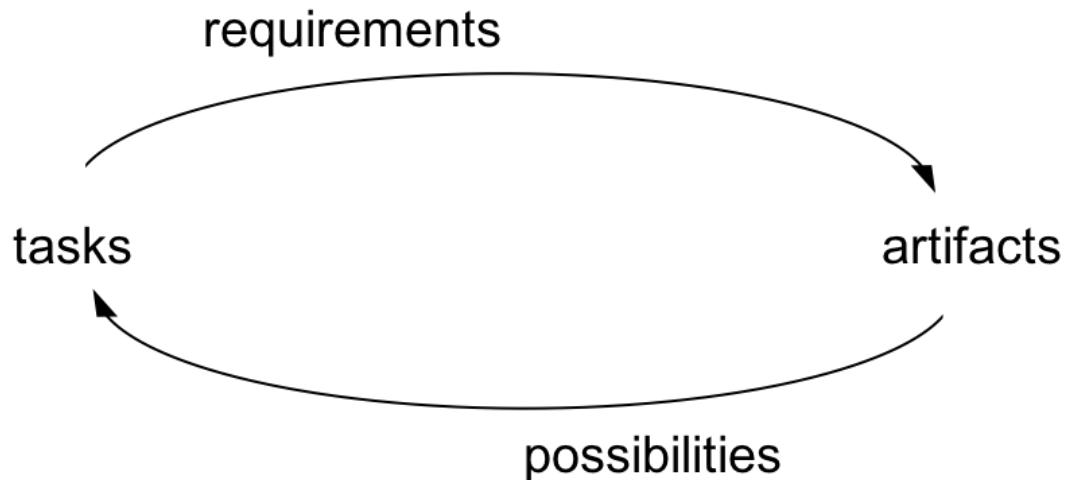


Fig. 6.1. Task-artifact co-evolution structured with design methodology from ?

6.1.3 Design research

6.2 Reflection on collaborative information analysis support

6.2.1 Analyst in control

Analysts should always be in control. This includes manual annotation, deciding uncertainty.

Unlike in other entity-based systems such as Bier et al. (2010); Stasko et al. (2008), in which natural language processing (NLP) techniques are employed to extract and model entities automatically, we use annotation to allow analysts to manually create evidence objects of interest. While algorithms for named entity recognition have improved significantly, developers of the entity-based systems (e.g. Gorg et al. (2014)) admitted that the accuracy of automatic entity extraction is not sufficient to support human analysis. Besides, relationships identified by algorithms are limited to co-occurrence of

entities in the same document, whereas identifying meaningful relationships *implicit* in documents that are of more importance to analysts is not possible. And perhaps the most problematic is the fact that the NLP treats all pieces of information equally with no regard to the problem analysts have at hand. We posit that information analysis is a progressive process, wherein analysts place varying priority on evidence depending on how they frame the problem Heuer (1999). Manual annotation allows for greater user control, allows more integrated source data objects to be identified, and avoids the user problems associated with automatic identification of disaggregated people, locations and times Bier et al. (2008). Users can decide their own information of interest and granularity that best suits their ad-hoc analytic needs.

Learning collaborative information analysis

hands on experience shape behavior

6.2.2 The cost vs. benefits of structured techniques

The cost

The benefits

role of visualization in reducing cost while maintaining benefits

Human mind is blamed for its stateless memory—people keep update with the current thinking, but lose track of the last idea even ten seconds ago. Information analysis typically takes a long time, and the final conclusion may differ completely from the original idea. Analysts need a way to record their analysis process to gain support and evidence for their final result.

Apart from the cognitive reason, information analysis in a collaborative setting suggests another reason. People "think" in a different way in group than individually—the so called collaborative thinking. Recording and sharing people's reasoning keep collaborators aware of their activities, their knowledge, and their rationale.

Visual analysis is a reasoning process. Analysts keep on interacting with visualizations—changing layout, filtering, zooming, brushing, linking, etc.—to explore useful information. A specific visualization may reveal a key pattern otherwise unidentified (analytic touch point). It is important to save that visualization as evidence—actually people do that in real, typically by saving screenshot. Pinning visualization enables two functions: saving that visualization, and attach textual explanation for the graph. What's more, the visualization can be retrieved and relayed for further analysis.

6.2.3 The cost vs. benefits of collaboration

In general, we find teams do better when they employ a strategy that demands closer collaboration. The strategy probably pushes teams to collaborate more closely, which in turn makes them better aware of partner's activities. In contrast, other teams well scoped their individual work, to such an extent that they almost had no need to talk to each other. It eventually harms their teamwork, as some teams reflected themselves as described in section 4.4.

? distinguishes between content common ground and process common ground.

Process common ground

Reaching common ground vs. maintaining common ground

6.2.4 Education in collaborative information analysis

' behavior towards more collaborative learning. With traditional single-user tools, students often employ a divide-and-conquer strategy; they divide their job responsibility, work individually on their own part, and put the results together in the end for report submission. In our study, we observed that students spontaneously conducted closer collaboration and enjoyed being able to contribute simultaneously.

An important direction for this research is to explore how such collaborative instrument could be employed to support collaborative learning in other course contexts. We studied an introductory intelligence analysis course, but many other courses involve heavy information analysis tasks, in which snippets of information must be annotated and analyzed with respect to concepts, connections, weights of evidence, and pros and cons of opinions. This is similar to generating and evaluating hypotheses in our case. Thus one could imagine courses in business, new media, and those involving heavy literature review adopting this tool similar to the case described here.

In addition to shaping the *learning* behavior of students, a collaborative supporting tool can also change how an instructor *teaches* analytic skills. Anecdotal reflections from the instructor in our classroom study emphasized the role of system support in *instructor intervention*. During our classroom study, the instructor frequently went over to student's desk and checked how they were doing. The instructor commented that he valued students' analytic process as much as their final report. His emphasis on analytic process is consistent with the value of intelligence community, who claimed that analysts "should think about how they make judgements and reach conclusions, not just about the

judgments and conclusions themselves" Heuer (1999). However, process is hard or costly to capture for assessment in reality. In the context of education, for example, all students are conducting analysis at the same time. Without explicit support, the instructor has limited supervision over the process.

The collaborative learning instrument is likely to provide an opportunity to get student's analytic traces preserved and assessed because the system has already captured and saved user interactions. These logs are currently saved for research purposes in our study, but potentially they can be displayed to the instructor for assessment purposes. Taking advantage of the the system's synchronization capability, student's behavior can be streamed to a "monitoring dashboard" in real time, from which the instructor could check student's progress and take early intervention if any team deviates the expected path.

!– paper supporting that early intervention –i

Chapter 7

Conclusion

Bibliography

- Amershi, S., Chickering, M., Drucker, S. M., Lee, B., Simard, P., & Suh, J. 2015, in Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, 337–346
- Bertin, J. 1983
- Bier, E. a., Card, S. K., & Bodnar, J. W. 2008, in 2008 IEEE Symposium on Visual Analytics Science and Technology (Ieee), 99–106
- Bier, E. a., Card, S. K., & Bodnar, J. W. 2010, IEEE transactions on visualization and computer graphics, 16, 178
- Borge, M. & Carroll, J. M. 2014, in GROUP '14 Proceedings of the 18th International Conference on Supporting Group Work, 215–225
- Borge, M., Ganoe, C. H., Shih, S.-I., & Carroll, J. M. 2012, in Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work - CSCW '12 (New York, New York, USA: ACM Press), 1105–1114
- Carroll, J. M., Borge, M., & Shih, S. 2013, Journal of Visual Languages & Computing, 24, 248
- Carroll, J. M. & Campbell, R. L. 1989, Behaviour & Information Technology, 8, 247
- Carroll, J. M., Neale, D. C., Isenhour, P. L., Rosson, M. B., & McCrickard, D. 2003, International Journal of Human-Computer Studies, 58, 605

- Chen, D., Bellamy, R. K. E., Malkin, P. K., & Erickson, T. 2016, in Visual Languages and Human-Centric Computing (VL/HCC), 2016 IEEE Symposium on. IEEE, 87–95
- Chin Jr, G., Kuchar, O. A., & Wolf, K. E. 2009, in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 11–20
- Convertino, G., Mentis, H. M., Slavkovic, A., Rosson, M. B., & Carroll, J. M. 2011, ACM Transactions on Computer-Human Interaction, 18, 1
- Diehl, M. & Strypebe, W. 1987, Journal of personality and social psychology, 53, 497
- Director of National Intelligence. 2008, Vision 2015: A Globally Networked and Integrated Intelligence Enterprise, Tech. rep.
- Dorn, B., Schroeder, L. B., & Stankiewicz, A. 2015, in CSCW '15 Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing, 393–403
- Duncan, K. A. & Wilson, J. L. 2008, Risk analysis, 28, 1699
- Farooq, U., Ganoe, C. H., Carroll, J. M., & Giles, C. L. 2009, International Journal of Human Computer Studies, 67, 297
- Gorg, C., Liu, Z., & Stasko, J. 2014, Information Visualization, 13, 336
- Gotz, D. & Zhou, M. X. 2009, Information Visualization, 8, 42
- Goyal, N. & Fussell, S. R. 2016, in CSCW '16 Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing

- Goyal, N., Leshed, G., Cosley, D., & Fussell, S. R. 2014, in Proceedings of the Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 129–138
- Goyal, N., Leshed, G., & Fussell, S. 2013, in CHI '13 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2721–2724
- Graves, E. 2000, Depicting strategic warning: A new model for the age of information operations., Tech. rep.
- Greenberg, S. 1990, COCS '90 Proceedings of the ACM SIGOIS and IEEE CS TC-OA conference on Office information systems, 11, 227
- Grudin, J. 1994, Computer, 27, 19
- Guo, H., Gomez, S. R., Ziemkiewicz, C., & Laidlaw, D. H. 2016, IEEE Transactions on Visualization and Computer Graphics, 22, 51
- Gutwin, C. & Greenberg, S. 1998, in CSCW '98 Proceedings of the 1998 ACM conference on Computer supported cooperative work, New York, NY, USA, 207–216
- Hajizadeh, A. H., Tory, M., & Leung, R. 2013, IEEE transactions on visualization and computer graphics, 19, 2189
- Hart, S. G. & Staveland, L. E. 1988, Advances in Psychology, 52, 139
- Heer, J., Mackinlay, J. D., Stolte, C., & Agrawala, M. 2008, IEEE Transactions on Visualization and Computer Graphics, 14, 1189

- Henman, L. D. 2003, in Small group communication theory & practice: An anthology, 8th edn., ed. R. Y. Hirokawa, S. Robert, L. A. Samovar, & L. A. Henman (New York: Oxford University Press), 3–7
- Herrmann, T., Nolte, A., & Prilla, M. 2013, Computer Supported Cooperative Work, 22, 241
- Heuer, R. J. 1999, Psychology of intelligence analysis (Lulu.com)
- IBM. 2017, i2 Analyst's Notebook
- Isenberg, P., Elmquist, N., Scholtz, J., Cernea, D., & Hagen, H. 2011, Information Visualization, 10, 310
- Isenberg, P., Tang, A., & Carpendale, S. 2008, Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI '08, 1217
- Jansen, B. J. & Rieh, S. Y. 2010, Journal of the American Society for Information Science and Technology, 61, 1517
- Kang, R., Kane, a., & Kiesler, S. 2014, Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing, 797
- Kang, R. & Kiesler, S. 2012, in Proceedings of the ACM Conference on Computer Supported Cooperative Work, CSCW, 123–126
- Kang, Y.-a. & Stasko, J. 2011, in 2011 IEEE Conference on Visual Analytics Science and Technology (VAST) (Ieee), 21–30

- Karvetski, C. W., Olson, K. C., Gantz, D. T., & Cross, G. A. 2013, *EURO Journal on Decision Processes*, 1, 205
- Lehner, P. E., Adelman, L., Cheikes, B. A., & Brown, M. J. 2008, *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 38, 584
- Lord, C. G., Ross, L., & Lepper, M. R. 1979, *Journal of Personality and Social Psychology*, 37, 2098
- Maceachren, A. M., Robinson, A., Hopper, S., Gardner, S., Murray, R., Gahegan, M., & Hetzler, E. 2005, *Cartography and Geographic Information Science*, 32, 139
- Mahyar, N. & Tory, M. 2013, *Visualization and Computer Graphics, IEEE Transactions on*, 20, 1633
- Martin, I. 2014, in *The International Annual Scientific Session Strategies XXI 2*
- Morton, K., Balazinska, M., Grossman, D., Kosara, R., & Mackinlay, J. 2014a, in *ACM SIGMOD*, 17–22
- Morton, K., Balazinska, M., Grossman, D., & Mackinlay, J. 2014b, in *Proceedings of the VLDB Endowment*, Vol. 7, 453–456
- Nobarany, S., Haraty, M., & Fisher, B. 2012, in *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, 1223–1232
- Olson, J. S., Wang, D., Olson, G. M., & Zhang, J. 2017, *ACM Transactions on Computer-Human Interaction (TOCHI)*, 24, 1
- Pang, A. T., Wittenbrink, C. M., & Lodha, S. K. 1997, *The Visual Computer*, 13, 370

PARC. 2017, ACH

Paul, S. & Reddy, M. 2010, in Proceedings of the 2010 ACM conference on Computer supported cooperative work, 321–330

Pirolli, P. & Card, S. 2005, in Proceedings of International Conference on Intelligence Analysis., 2–4

R Core Team. 2016, R: A Language and Environment for Statistical Computing

Schmidt, K. 2002, in Computer Supported Cooperative Work (CSCW), 285–298

Scholtz, J. & Endert, A. 2014, 2014 International Conference on Collaboration Technologies and Systems (CTS), 478

Schumann, J., Buttler, T., & Lukosch, S. 2013, Computer Supported Cooperative Work, 22, 271

Shah, C. 2014, Journal of the Association for Information Science and Technology, 65, 215

Sparrow, M. K. 1991, Social Networks, 13, 251

Stahl, G. 2006, Group cognition: Computer support for building collaborative knowledge (Mit Press Cambridge, MA)

Stasko, J., Görg, C., & Spence, R. 2008, Information Visualization, 7, 118

van Gelder, T. 2008, Australian Institute of Professional Intelligence Officers News, 55

Viegas, F. B., Wattenberg, M., Van Ham, F., Kriss, J., & McKeon, M. 2007, IEEE Transactions on Visualization and Computer Graphics, 13, 1121

Ware, C. 2012, Information visualization: perception for design (Elsevier)

Wegner, D. M. 1987, in Theories of group behavior (Springer), 185–208

Wheaton, K. 2011, Unpublished manuscript

Wright, W., Schroh, D., Proulx, P., Skaburskis, A., & Cort, B. 2006, Proc. CHI '06, 801

Yi, J. S., Kang, Y. A., Stasko, J., & Jacko, J. 2007, IEEE transactions on visualization and computer graphics, 13, 1224

Zajonc, R. B. 1965, Science, New Series, 149, 269

Zuk, T. D. 2008, PhD thesis, University of Calgary

Vita

Your Name Here

Education

Ph.D. in Astronomy & Astrophysics, expected in (Month) (Year)

Area of Specialization: Something

(*Undergrad Institute Name*) (City), (State) 199?–199?

B.S. in Physics (or whatever), *cum laude* with distinction in the major (etc.)

Awards and Honors

Award Name	year
Another Award	200?–200?
One more for good measure	200?, 200?

Research Experience

Doctoral Research The Pennsylvania State University 199?–Present

Thesis Advisor: Prof. Someone R. Other

This research involved lots of cool stuff.

Graduate Research The Pennsylvania State University 199?–199?

Research Advisor: Prof. Someoneelse R. Other

This research also involved lots of cool stuff.

Undergraduate Research Undergrad University 199?–199?

Research Advisor: Prof. Name Goes Here

Really cool stuff is what I did for this research.

Teaching Experience

Guest Lecturer The Pennsylvania State University 199?–Present

I taught lectures which involved doing cool stuff.

Teaching Assistant The Pennsylvania State University 199?

I taught labs and did cool stuff.

Another Experience Some School, Some City, Some State 199?–199?

I taught cool stuff to cool people.