DSOC: Decision Support for Organizational Change

Rob Barwell and Eric Spero

Fig. 1. Version 3 of DSOC. Columns: Demand (left) and Supply (right). Rows: Organizational (top), Process (middle) and Redundancy (bottom).

Abstract— DSOC is a novel way to visualize the complex problem of organizational change by allowing the user to quickly see where gaps exist in an organization, understand the gaps that exist, and help the user determine how they might fill the gaps. This is supported by using email within an organization to model the flow from one employee to another. This representation provides the most current and concrete representation of the organization and allows the user to make better decisions when faced with organizational change.

Index Terms—Organizational change, decision-making, information visualization

---◆---

1 INTRODUCTION

The rise of the 'gig economy' [1,2] has forced society through a massive transition in the past two decades with people transitioning between jobs with greater frequency. Historically, a person would obtain a job from high school or university and stay with a specific company until they retire. This resulted in organizations that had minimal change, which could be easily managed. Modern organizations are forced to adapt with the shift to the gig economy and other non-traditional employment models.

An organization is a social network [3] of individuals engaged in

---

• *Rob Barwell and Eric Spero are with Carleton University. E-mail: rob.barwell/eric.spero@carleton.ca.*

complex *interlocking contingencies* [4]: individuals in an organization are tightly interconnected, where the behaviour of one both depends on, and has subsequent consequences for the behaviour of others [4]. This feature of organizations means that the results of changes to its social structure—say, by adding or removing an individual—can be difficult to understand. Change is disruptive, and care must be taken to minimize any negative side-effects. The challenge of minimizing organizational disruption as a result of personnel change depends critically on understanding the effects of that change. In the gig economy, this challenge, and the need for tools that support it, is even greater.

Say a bit about emails here?

During the course of development, as a result of our conversations with others and our own observations, we noticed that the problem of organizational change boils down to a problem of *demand and supply*. When an individual leaves or is removed from an organization, this

creates a need (i.e. a gap, a *demand*) which must be addressed by the manager. To do so, the manager will have to inspect their internal pool of resources (i.e. their *supply*) and assign new duties/responsibilities to one or more people.

An individual's function within an organization can be viewed using different lenses. For example, employees can be viewed in terms of their position in the hierarchical structure of the organization, the processes they participate in, and their process- and organizational-agnostic social ties. When dealing with an organizational change issue, the manager be considering each of these views (and perhaps others not mentioned) on both sides of the demand and supply issue mentioned above.

We present DSOC (Decision Support for Organizational Change), a visual analytics dashboard to help managers respond to organizational change in a way that minimizes disruption to the organization. A screenshot of our functional prototype is shown in Figure 1. When using DSOC, managers are shows a number of visualizations that each help managers grapple with a part of the complex demand and supply problem. Visualizations are organized into two columns: demand and supply, and as many rows corresponding to the views of an employee that a manager is interested in. We implement demand and supply visualizations for three views: *organizational*, *process*, and *redundancy*, which correspond to the three example views mentioned in the example above.

In this paper, we discuss . . . in Section 2, . . . in Section 3 . . . in Section 4 . . . in Section 5 . . . in Section 6 . . . in Section 7

2 Literature Review

Need for a review of the literature in this area. Relevant concepts, similar works, etc.

2.1 Information visualization and cognition

We aim to address this problem through *information visualization*. Visualizations support thought by reducing the gap between the data, and the users' *mental model* of the data [5]. A mental model is an internal representation of how something in the world works [6, 7]. Wherever there is distance between the presentation of the data and our understanding of the data, mental work must be done so that understanding is possible. This type of mental work does not bring us closer to solving domain goals, but rather is a sort of unfortunate precursor for the really important work, and therefore should be avoided wherever possible [8]. Fortunately, the physical environment can be used to store information, which allows us to 'off-load' mental work onto the environment [9]. Visualizations are essentially one way of effectively leveraging this property of the environment to aid thought.

The development of our visualization started with defining a methodology to help guide the design process. We reviewed a number of papers in the area of evaluation of information visualization. We found this area of research to be less concrete then other topics we had studied before. The Challenge of Information Visualization Evaluation [10] paper we reviewed helped us to clarify our evaluation criteria and approach by validating our desire to present our design and findings to various user groups throughout the design process. The Empirical Studies in Information Visualization: Seven Scenarios paper [11] paper inspired us to take an iterative approach, similar to agile software development. The approach presented in the paper was Pre-design, design, prototype, deployment, and re-design. This approach would also line up with using different user groups through the process.

3 Design

Our design is built on top of two key ideas:

1. The problem of dealing with an individual leaving an organization is in essence a problem of *supply* and *demand*.

2. An individual's role in an organization is *multifaceted*: it is different depending on the *view* one takes.

When an individual leaves an organization, a gap has been left (or a demand has been created), which needs to be filled. The manager responsible for addressing this issue will first need to take note of what and where the gaps are. Then, they will look at the remaining members of their organization to find ways of filling those gaps.

The problem is that there are a number of different kinds of gaps, pertaining to each of the facets, or views, of the individual's role. These views are often not reflected in the person's job description, and they might be unknown, or difficult to find out. The supply and demand problem spreads over all of these views. This problem is duplicated when looking through one's supply for replacements, as their roles are equally difficult to know.

Our interface is meant to visualize the *demand* created by an individual's departure from an organization for each facet of their role. Alongside the visualizations of the demand created, we want to explore the space of potential replacements, visualizing their roles as well.

In the remainder of this section, we discuss the various elements of our design in more depth, tracing its evolutionary history through several versions, and providing rationale for our design decisions.

3.1 Background

3.1.1 Methodology

Our design was informed by consultation with three user groups. Those groups are:

A. **Government Department**. Members of a large department at the Government of Canada who address organizational change issues in their work.

B. **Academia**. Students and a professor at Carleton University.

C. **Private Company**. Users at a private multi-national company who discovered our research during the design process.

We made the real-world viability of our design a top priority. As such, we felt it was important to show our design to people from a number of different backgrounds to help ensure that our design was comprehensible to a wide variety of potential users, and that our visualizations effectively communicated the underlying data and helped support the task at hand. Our initial design underwent two major revisions in response to feedback received from members of these groups.

After implementing Version 1 (V1) of our prototype we presented it to Group A. Group A provided the key insight that the problem we were addressing was problem of demand and supply across multiple views. The design concept behind V2—a major redesign—came directly from our interactions with Group A. V2 was then shown to V3, who pointed out a number of issues with specific visualizations. This group provided creative ways of helping support the needs identified by Group A. V3 was then shown to all groups, where no major issues with our implementation were identified.

3.1.2 Problem Space

After implementing our original design and going through several subsequent iterations, the way in which we framed the core problem we were trying to address also changed. We loosely followed Pirolli and Card's sensemaking loop [12]: we started with a question, eventually leading to a hypothesis, and then through a validation process, leading us back to a new formulation of the question.

The problem as we originally conceived it was simply how to help managers fill empty positions in the gig economy era, which has high employee overturn. Managers rely on organizational models like organizational charts to help with this task, which are representations of the structure and function of an organization. Organizational charts are not ideal tools for this task as they are quickly out of date in the gig economy, and they provide a limited view of a person's role in an organization. We set out to offer a new solution leveraging the power of graphical user interfaces and information visualization.

As we developed our design, we found it useful to break our original definition of the problem into sub-problems. Each sub-problem imposed their own demands on the interface, which were lost in the original definition. We eventually recast the original problem into three research questions:

1. "How do we support decision makers to minimize negative side-effects from organizational change?"

2. "How do we value a person in the organization?"

3. "How do we provide options to users to address vacancies within an organization?"

From the point of view of the manager, these questions take the following form:

"Where is the gap?" Gaps left when an employee leaves are not restricted to the employees organizational group, but could effect other groups as well. Job descriptions are typically a poor reflect of an employees job as they are only updated when an employee leaves. This means Eric in our example needs to look at the organizational side of the problem and the process side of the problem.

"How big is the gap?" When gaps are found the manager needs to know are they small such as a person just needs to sign an approval form, or are they big such as the person who is responsible for producing a yearly financial report.

"What can I do about the gap?" This problem is the most complex as it require the manager to evaluate first if they are filling the gap and then what resources do they have available to them to fill the gap.

3.1.3 Option Space

Using a refined problem description we started to focus on how we would address these statements. Using the work done by Ware on Visual Thinking Algorithms [13, Chapter 11] we decided to enumerate the steps a user would do manually to solve the problem statements and determine where visualization could help offload cognitive processes.

During our walkthrough of V1 with Group A, we presented V1 of the interface and asked users to imagine using the software to solve the original problem. Group A immediately sub-divided the problem into the three sub-problems mentioned in Section 3.1.2, and explored it from different perspectives or views. Group A expressed that it would be helpful if the interface could model all of the different views. The different views are summarized below:

Organizational This places an employee in the traditional hierarchical organizational structure. Commonly this is represented by a tree where the employee is a node and an edge is the link between nodes.

Process This places an employee as part of a process where they receive information, process information, and output information to the next stage of the process. Users typically used flow charts and other process diagrams to represent this view.

Redundancy During the walkthrough we noticed it was most common to address the vacancy by re-distributing the workload within the group, however the users occasionally tried to find people that did similar tasks as replacement.

3.1.4 Data

During the exploration of the problem space we needed to identify what data we were going to present to the user with the visualization. Building upon previous work from our colleague on information flow within an organization **Can we say a sentence or two about this previous work?**, we decided to use emails. The majority of communication within a modern organization is through email or instant messaging. We thought that analyzing email data would allow us to build an accurate and up to date model of how information flows through an organization.

Using emails as the basis of our data model was well-received by Group A because it addressed some shortcomings with the traditional
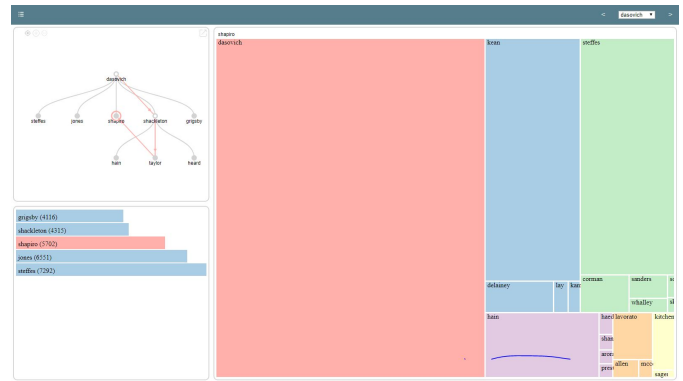

Fig. 2. Version 1.

organizational chart model, which is typically out of date and does not capture all of the contributions an employee makes to an organization.

The data model requires enrichment through the addition of value models. One value model would be required to help the manager find out "How big is the gap?" and another value model would be required to help the manager find out "What can I do about the gap?". We use rudimentary value models as the primary focus of this paper is the problem of how to visualize the data, and not the data itself.

Gap Size By modeling each employee as a node, the emails between nodes would represent an edge with a weight of 1.

Gap Options One option to address the gap could be redistributing the work to a 'similar' employee. The value model we used to compare the similarity of two employees is 'overlap' in terms (a) number of emails sent, and (b) who the emails are sent to.

3.2 Versions

Our made incremental refinements to our design in response to feedback obtained from the users in the groups mentioned in Section 3.1.1. Our design went through three major versions:

V1 The initial design by the authors that focused on the organization and communication problem.

V2 Overhauled layout and introduced grid layout. Added the process and redundancy views.

V3 Refined the process and redundancy views to better align with the users' mental model and how they interact with the visualization.

Our greatest challenge was creating a functional and comprehensible layout that also conveyed the complex intricacies of organizational change.

3.3 Layout

Here we discuss the structure of our layout and how it is navigated.

3.3.1 Structure

The original design, shown in Figure 2, had viewing panes to try and convey the organizational perspective of information flow and the process perspective. It attempted to do this using the standard organizational tree for organizational structure and a tree map to show the volume of communication between a selected individual and their communication links.

This view presented many challenges for users as they found it difficult to understand what the visualization was trying to communicate. There seemed to be a severe incompatibility between this design and Group A's mental model of the problem of organizational change. For Group A, the problem at hand boiled down to a problem of *supply* and *demand* (these were the words they used most frequently during
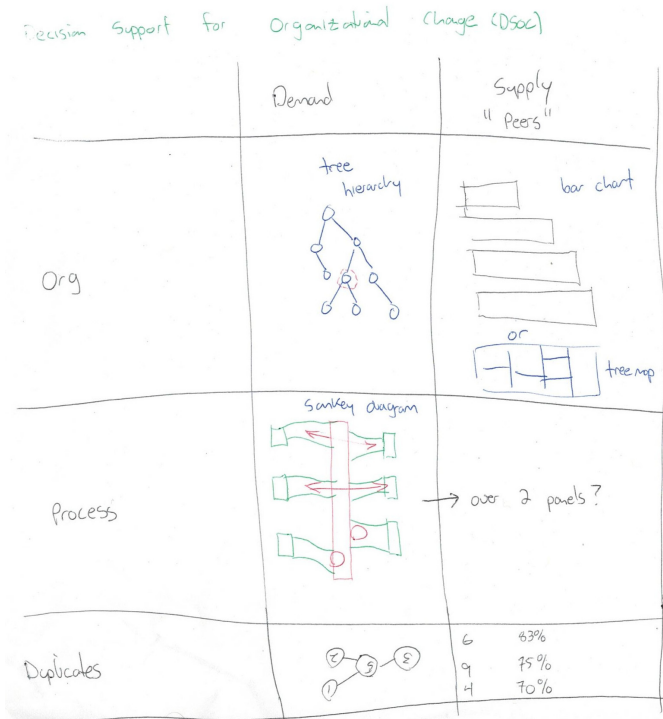
Fig. 3. UI design sketch drawn after initial meeting with Group 1. Shows the grid pattern and several of the visualizations used in the final version.

Fig. 4. Version 2

this conversation), yet these concepts were not straightforwardly represented in our design. This observation called for a major redesign of the structure, where starting in V2 (Figure 4) we would build the notion of supply and demand into our structure. The first stage in this redesign is captured in the sketch in Figure 3.

The new design featured a grid layout of 2 columns × N rows. Visualizations that focus on the *demand* aspect of the problem are placed in the left column, and visualizations focusing on the *supply* aspect are placed in the right column. By visualizing both the supply and demand aspects of the problem next to each other, managers can simultaneously see the gaps left by an individual's absence, and also ways to fill those gaps.

Visualizations for the different views of the problem space were placed inside their own row. We say there are N rows because there is no theoretical upper limit to the number of views that a manager may want to examine. The views that a manager may want to examine can vary based on the demands of the specific situation they are in. We provide three example views, but more can be created and added to the dashboard as needed.

We found that visualizing each view of the problem space separately had a similarly positive effect as visualizing supply and demand.

3.3.2 Navigation

Our goal was to provide the user with the freedom to explore the problem space however they choose. This required the support of a brushing scatter plot concept where the interactions within an individual sub-view are reflected in the other sub-views. The visualizations are all driven by the *selected* node: users view a single data point through a number of lenses. Users can select a node by clicking on nodes in either of the "Organizational" views.

As the user explored the problem it was also important to have a way for them to keep track of the problem space they have already explored. This was a salient point that we found during the literature review with the quote "Information exploration is inherently a process with many steps, so keeping the history of actions and allowing users to retrace their steps is important" [14]. This was included by providing the user the option to display history as direction arrows or simply colouring
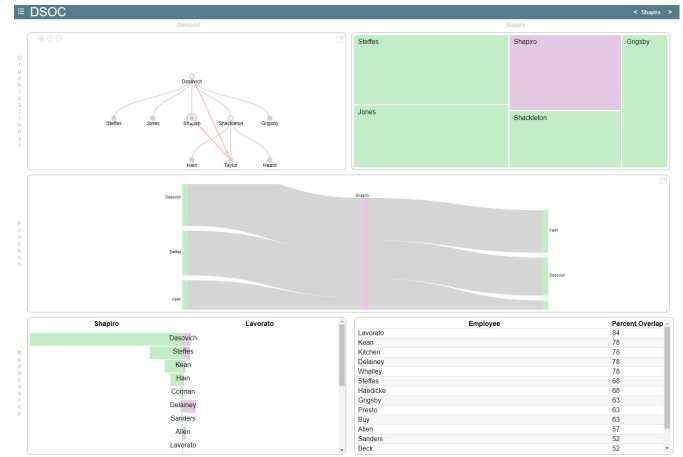
the nodes in the organizational chart. The user can then navigate the history using the navigation bar at the top of the screen.

3.4 Views

3.4.1 Organizational

The organizational demand and supply view changed relatively little between V1 and V3 compared to the other visualizations. The org-demand view was modelled after a classical organizational tree structure, as this paradigm was most aligned with the users current mental model of the problem. During the design phase we uncovered a number of issues that we needed to address, which we discuss next.

Actions Organizational trees can be quite large, so we need to provide the user with the ability to zoom in and out of the tree, expand and collapse a node's children, and focus on a specific node. In V1, a single click both expanded a node's children and selected the node; double-click would zoom in and out of the tree. Users in Group A complained about this, and wished to expand a node without selecting it and vice versa. These functions were separated more cleanly in V2, where a single click selected a node, the double click would zoom in and out, and a right mouse click would open a node with children.

Highlight During the initial design of V1 we needed to come up with a creative solution to visualize the various states a node could have. This includes: being selected, having children, being visited, and being filtered. We originally experimented with glyphs and other icons, however that the resulting visualization was too busy. We settled on using concentric circles of different colours to denote various states of a node. The concentric circles can be seen in Figure 5 A node with children has a filled in circle; nodes the user has visited have a faint orange outline; nodes that are filtered have a darker orange ring around the outline of the main circle; and the selected node has an even larger red ring around the previous ring.

Navigation Navigating large trees can present challenges for users. One way we help users navigate large trees is with a real-time filter. Users can adjust the filter parameters, and the nodes that meet the filter requirements are highlighted with an orange circle, distinguishing them from non-filtered nodes. Users can also expand the organization demand panel to fill the whole window so users can see more of the tree at once. Icons at the top of the frame provide navigational short-cuts allowing the user to center on the current selected node, expand all nodes, or collapse all nodes.

Comparison When users hover over a node, a blue line is drawn from that node to all the nodes they communicate with. This allowed users to see at a glance everyone a given individual talked to. Originally the strength of the connection between two nodes was not communicated to the user. In later revisions, after discussions with Group B, this feature was changed to communicate the strength of the connection
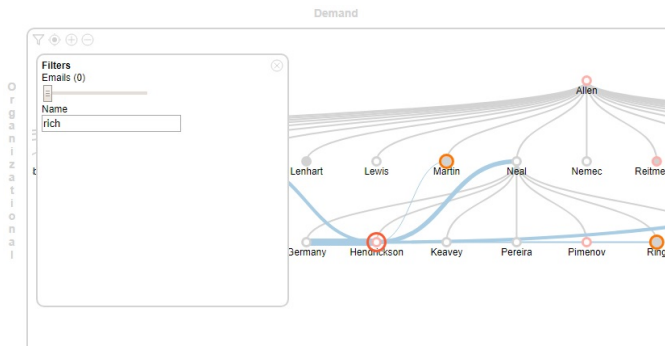
Fig. 5. Node Selection

Fig. 6. Palm Tree

through the thickness of the blue line connecting the two nodes, where a thicker line meant a stronger connection. This feature can be seen in Figure 5.

The organizational supply view was originally a horizontal bar chart, as shown on in the bottom-left pane in Figure 2. In V2 we experimented with changing this to a tree map, as we wanted to show the relative size of each employee in a group compared to the others. The space filling method [15] as discussed in Ben Shneiderman original treemap paper was appealing as we did not know how large a group might be. After discussion with Group B we decided treemaps were not appropriate as there were no levels to drill down to or group by. It was also noted by Group A that a bar chart might be better for a comparison view point. When the user clicks on any of the bars it will re-orient all views to select the employee. The final vertical bar chart is shown in Figure 1.

3.4.2 Process

The process demand and supply visualizations underwent major changes between versions. In V1 a treemap was used to model the communication flow between the selected employee and those they communicate with. This approach provided an overview of who the employee talked to and how much. However, Group A noted that this approach failed to give an impression of how information flowed through an individual, nor what processes started or ended with an individual—two things a manager must consider when looking for gaps and ways of filling gaps. Group A's suggestion was to investigate using a Sankey diagram, which we added in V2.

The Sankey diagram used in V2 and V3 show the selected employee in the center, the employees who send emails to the selected employee on the left, and the employees who the selected employee sends emails to on the right. The relative width of the lines communicates the number of communications sent. Dark lines on the left indicate that the employee talks to the selected employee but not vice versa, and dark lines on the left indicates the reverse. These suggest instances where the selected employee is the start (right) or end (left) of a process.

The original version of the Sankey diagram in V2, shown in Figure 4, was problematic for Group B. The flow lines on the left side of the selected employee (the line in the middle of the diagram) appeared to carry through to the lines on the right, which made it seem as though the information from an employee on the left was transferred directly to an employee on the right by the selected employee. This was an accident of the layout algorithm and not something that we intended to communicate (since we have no knowledge of the content of the emails), and therefore it was misleading. Additionally, the lines between employees were typically quite large which made the visualization hard to read. Also, employees on the left and right were presented in different orders, which made direct input-output comparisons for a given conversational partner difficult. Based on this feedback, we decided to completely redesign this diagram for V3.

For V3, we scaled the links down to make them easier to differentiate on the screen; we added colour denote links that either started or ended with the selected employee; and we lined up employees that appeared on both the left and right, to allow for easy comparison.
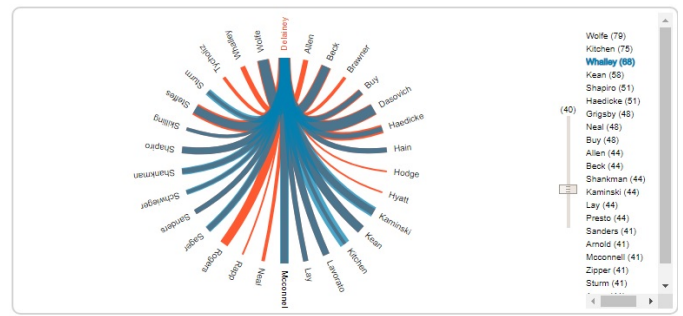
The Sankey diagram is unique among the visualizations we use in its ability to communicate both supply and demand simultaneously. This feature will be further discussed in the evaluation and discussion section.

3.4.3 Redundancy

The redundancy view was introduced in V2 after our initial discussions with Group A. They thought it would be a good idea to show similar people within an organization as a potential supply source to fill the identified gap. V2 included a simple implementation of this concept. In the demand pane, a back to back bar chart was used to help the user determine if the supply lined up with the demand. In the supply pane, we provided a table showing the employees who had the most 'overlap' with the selected employee, in descending order of percentage of overlap.

Group B expressed dissatisfaction with our decision to show overlap supply with a table. They felt that overlap was an inherently visual concept, and thus demanded a visual representation. This led us to investigate a number of options including hierarchical edge bundling and chord diagrams to try and show the overlap between two selections. The chord diagram was an unsatisfactory solution as the selected employee filled the entire circle, making it hard to differentiate between the different paths. The hierarchical edge bundling had the major drawback of not showing the relative size of each connection. We decided to merge these two concepts together, creating a single visualization: the 'palm tree' diagram shown in Figure 6.

The palm tree diagram quickly shows how much an employee overlaps with a selected employee in a visual fashion. The diagram shows the selected employee, whose name is written in orange, and all of the employees they communicate with. Their connection is signalled with an orange line, and the width of the line shows the volume of communication between the two parties. A list on the left shows a list of other employees in the organization in descending order of the amount they overlap with the selected employee. A slider to the left of this list allows the user to filter this list by overlap percent. When the user hovers over an employee on this list, their pattern of communication with the employees the selected employees speaks to is drawn with blue lines from the same origin as the selected employee. By rolling the mouse up and down the list, the user can make quick comparisons between employees. We chose the orange and blue colour scheme because we felt it provided the optimal balance between high contrast, allowing the user to easily distinguish between the two lines when performing overlap comparisons, and aesthetics.

This approach seemed to resonate well with all user groups in V3. Users appreciated its flexibility in gracefully showing a large number of connections, and its ability to convey overlap information quickly and effectively.

4 IMPLEMENTATION

Each version of the design was implemented using MySQL, PHP, HTML, CSS, and Javascript using the D3 library. A number of visualization systems were contemplated, however D3 was chosen due to its maturity level and flexibility.

4.1 Data

Email (SMTP) header information was used to implement the value model discussed in the previous sections. Alternative value models are discussed later in the paper. **Where?**

To demonstrate our value model we chose the Enron email dataset [16, 17]. This dataset was originally made public during the Federal Energy Regulatory Commission investigation. It contains the emails of approximately 150 users in the Enron organization, most of whom are senior managers. This particular dataset has been frequently used in research, typically with a focus on social networking and natural language processing (e.g. [18]). We use this dataset to illustrate our visualization to support the flow of information through an organization.

In addition to employee's emails, we also needed an organizational chart, which provides one of the views of an employee's role. Unfortunately, we were unable to find an official Enron organizational chart, which is a known issue with the dataset. Since our research is primarily focused on the visualization and not the underlying data, we decided it would be acceptable to use an organizational chart that was reverse-engineered from this dataset by a group from the University of Amsterdam [19]. Out of a number of different approaches to extracting organizational chart information from a dataset, the approach used by this group seemed to us to be the most sophisticated.

4.1.1 Data Preparation

The Enron dataset was obtained from Carniege Mellon University [16]. We used a Python script to parse SMTP header information for each email. We stored this data in a MySQL database, which would be further processed by the application. We chose SMTP information because it contained all the basic information required to construct the nodes and edges within the graph.

4.2 Organizational

The organizational view was straightforward to implement. We used the hierarchical tree and bar chart components of D3. Custom path generation was required to connect nodes upon a mouse hover event, shown in Figure 5. This was accomplished by binding the database node id to the D3 object id. This facilitated the selection of individual nodes in the tree to determine their x and y position to draw a path between. This process was also used for the history link paths, which can be seen in Figures 2 and 4.

The only design choice that arose during implementation was what to do when two nodes are connected, but one node isn't visible. In this case we decided to connect the visible node to the closest parent of the hidden node.

4.3 Process

The process view was the most complex to implement. It required a custom written layout generator. The generator takes a list of all nodes and edges for a current view and then splits the edges into incoming and outgoing. After iterating through the edges to determine if the incoming or outgoing edge is greater, it conducts a layout which utilizes the full space allocation. Nodes are also logarithmically scaled to reduce overlap and make the visualization easier to use.

4.4 Redundancy

The redundancy view was the most challenging to implement since it had a number of custom visualization components. A custom bar chart was required to display the back-to-back bars in the redundancy demand view, as D3 only had a standard bar chart. However, a number of D3 features that create bar charts were utilized to create the back-to-back bar chart.

The most challenging feature to create was the palm tree in the redundancy supply panel. This required a custom layout that would place the selected employee in a circle with all the people they communicate with. Then additional layers were placed on top to position the compared employee in place of the selected employee. This also required a custom data structure to communicate this information from the server to the D3 engine.

4.5 Other Implementation Choices

Colour A single colour scheme was implemented to ensure consistency across all views. This provides consistency for the user where all the nodes are the same colour regardless of view, and the selected node is easily identifiable. We chose orange and blue because they are highly contrasting while still looking aesthetically pleasing.

Layout The layout was implemented using CSS grids that allow for the screen to be divided into frames. This allowed us to show two full rows no matter the size of the window..

Labels The new supply and demand layout gave meaning to each row and column. We added labels to explicitly communicate this meaning to the user. Users found this very helpful to understand the problem space.

5 EVALUATION AND DISCUSSION

Evaluation of the design was done at a number of points during the development. We made many significant changes as a result of feedback from users. In later stages of evaluation, we received direction on future work, and gained some insight into how organizations might use the final product. During the development we explained our design using a use case to help participants put themselves in the shoes of the intended user, thereby increasing the ecological validity of the results. We describe the use case below.

5.1 Use Case Walkthrough

An employee, Rob, has left a company, and the manager, Eric, has to fill the need this has created. The employee could have left because they switched jobs, they could be on a temporary leave of absence, or they could be seconded to other positions. The manager needs to assess the gap that has been left, and how to fill it. To do this, Eric must seek answers to the three questions we described earlier in Section 3.1.2:

1. "Where is the gap?"

2. "How big is the gap?"

3. "What can I do about the gap?"

5.1.1 Finding the Gap

Our visualization helps Eric with each of these questions. First he can identify the gap by using the organizational demand tree shown in Figure 1. By navigating and filtering the tree Eric can start to understand if Rob managed any employees under him. The higher he is on the organizational chart, the more employees in the tree will be without leadership. The tree can also be used to quickly determine which parts of the organization might be effected most, by hovering over Rob and looking where the largest blue lines connect to.

Using the process view, Eric can determine which email connections start or stop with Rob. This typically denotes starts and ends of processes, which are high-priority gaps to fill. Additionally, the process diagram can provide Eric an overview of the stronger communication links Rob had, which signal processes which Rob was an integral part of.

5.1.2 Size of the Gap

During the exploration process in finding the gap, Eric has already started to understand the size of the problem by looking at weight of the lines in the organization and process demand chart.

5.1.3 Filling the Gap

After Eric has a feel for the locations and the size of the gaps that have been left by Rob's departure, Eric can now use the supply side of the application to see how he might fill these gaps. Managers commonly begin their search through the supply by looking at other members of the departed's group, which is shown in the organizational supply view. The organizational supply bar chart can quickly show Eric the relative

contribution of Rob to the group. Employees who are under-utilized (i.e. who have low bars) might be good candidates to take on at least some of Rob's workload. Eric may wish to take a closer look at a potential replacement's role in the organization, in which case he can click on that employee's bar to make them the new selected node, and the focus of the other visualizations. Afterwards, Eric can click the left arrow in the top right of the screen to go 'back' to making Rob the employee of focus.

The process chart can be used to determine if Rob is even required in some processes. This could result in system efficiencies by removing Rob from the process and simply having people on the left side connect directly with people on the right side. **Can you go into a bit more detail here about how Eric might use the process view? Maybe mention a few more examples. And a general note: I think this section is really important, so if you can think of more details to add throughout showing how someone might actually use the system in a real-world scenario that would really help the reader I think.**

Lastly, Eric can try and find other employees in the organization who provide a similar role to Rob (i.e. they communicate with the same people with a similar frequency), and see if they have spare capacity to take on some of the workload. Eric begins by hovering over the employee with the highest percentage overlap with Rob, and 'roll' down the list, looking for the right pattern. Eric may find a few employees that appear to have a similar pattern. He can click on their names to make them appear in the redundancy demand view opposite Rob. This view allows a finer-grain look into who they are talking to, and how much.

5.1.4 Result

The result of this process is Eric having a better understanding of the gap created when Rob left the position and possible sources of workload redistribution to people (a) within the organizational group, (b) others in the process, or (c) other employees in the organization that provide a similar role. In a real world scenario Eric would most likely fill the gap with one or more of the identified supply sources. This is why our tool was designed as decision *support* tool and not a decision-*making* tool, as only a human is able to weigh the pros and cons of these respective options and ultimately make the final decision.

5.2 Organizational Models

We showed the final design to Gropu C, which suggested the real-world applicability of this tool. When Group C saw the final design they were able to immediately envision how it could help their organization solve an organizational change problem they had been working on, which was to flatten their organizational chart. Having the different views that our application presented allowed Group C to revise their concept of using a hierarchy to model their organization and start thinking about using processes instead.

The new model that was discussed involves removing the hierarchical nature of an organization and replace it with the concept of employees that have skills and contribute to processes. This would result in the organization only needing two or three levels to manage the allocation of resources and deal with conflicts when they arise.

The concept of a traditional project manager would be extended and modified to create a process manager. The process manager would be responsible for identifying the skills required to accomplish their process. A resource manager would then provide an employee with those skills for the time identified by the process manager. This would have the effect of providing the organization with the flexibility to recruit specific skills vice generic positions that may or may not require that skill depending on the job.

The concept is further extended to employee remuneration, where it could be adapted to pay based on skill or proficiency of a skill. Employees with more skills or deeper knowledge of skills would be remunerated at a higher rate.

6 FUTURE WORK

During the design and development of the visualization we discovered many areas that required further thought and research. Some of these areas were identified by us and others came from the groups that supported our research.

6.1 Value Model

All groups were quick to understand the visualization we were demonstrating and wanted to focus their time and discussions around the value model aspect of the problem vice the visualization.

The weight of each edge in the value model needs to be adjusted. For example this could include a new value based on whether the user is in the `from`, `to`, `bcc`, or `cc` field. It could also explore the content within the email through natural language processing to reduce the weight of trivial emails that do not contribute value to the business.

The value model around redundancy could also be adjusted to account for volume or frequency of emails to help the user find another employee in the organization with appropriate overlap.

6.2 Performance

The current data is housed in a relational database which is not optimized for graph sets. To increase the performance of the application it might be better aligned with a noSQL database or other non-traditional database engine.

6.3 Data

The Enron dataset was used to demonstrate the current visualization, however it contains gaps within the organizational tree and only a limited set of email data. Further work would be required to validate our approach using a more fulsome and realistic dataset.

6.4 Views

The current layout was developed to allow extension to the application with new views into the problem. An interesting topic was generated during one of our discussions to refocus the problem from an internal redistribution of resources and also include the ability to obtain external resources to fill the gap. This would include a new view to identify the skill sets that are most common in the emails of an employee and help the user understand what types of skills would be required from an external hire.

6.5 Documentation

As the user explores the dataset it would be helpful for the application to capture the notes or thoughts of the user as the look at the problem space. This might involve adding a sticky note type feature to each node that could then be used in a collaborative environment to allow multiple users to generate a solution [13].

7 CONCLUSIONS

In this paper we presented DSOC: Design Support for Organizational Change, a visualization dashboard to help managers fill gaps resulting from an employee leaving an organization.

We found during our research there was a lot of excitement about this topic and there are a number of different pieces of follow on work that we can use in the future.

8 ACKNOWLEDGMENTS

REFERENCES

[1] V. De Stefano, "The rise of the just-in-time workforce: On-demand work, crowdwork, and labor protection in the gig-economy," *Comp. Lab. L. & Pol'y J.*, vol. 37, p. 471, 2015.

[2] G. Friedman, "Workers without employers: shadow corporations and the rise of the gig economy," *Review of Keynesian Economics*, vol. 2, no. 2, pp. 171–188, 2014.

[3] J. Scott, "Social network analysis," *Sociology*, vol. 22, no. 1, pp. 109–127, 1988.

[4] S. S. Glenn and M. E. Malott, "Complexity and selection: Implications for organizational change," *Behavior and Social Issues*, vol. 13, no. 2, pp. 89–106, 2006.

[5] J. S. Yi, Y. ah Kang, J. T. Stasko, J. A. Jacko *et al.*, "Toward a deeper understanding of the role of interaction in information visualization," *IEEE Transactions on Visualization & Computer Graphics*, no. 6, 2007.

[6] N. Staggers and A. F. Norcio, "Mental models: concepts for human-computer interaction research," *International Journal of Man-Machine Studies*, vol. 38, no. 4, pp. 587–605, 1993.

[7] D. A. Norman, "Some observations on mental models," in *Mental models*. Psychology Press, 2014, pp. 15–22.

[8] F. Paas, A. Renkl, and J. Sweller, "Cognitive load theory and instructional design: Recent developments," *Educational psychologist*, vol. 38, no. 1, pp. 1–4, 2003.

[9] M. Wilson, "Six views of embodied cognition," *Psychonomic bulletin & review*, vol. 9, no. 4, pp. 625–636, 2002.

[10] C. Plaisant, "The challenge of information visualization evaluation," *IEEE Proc. of AVI*, 2004.

[11] H. Lam, E. Bertini, P. Isenberg, C. Plaisant, and S. Carpendale, "Empirical studies in information visualization: Seven scenarios," *IEEE transactions on visualization and computer graphics*, vol. 18, no. 9, pp. 1520–1536, 2012.

[12] P. Pirolli and S. Card, "The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis," in *Proceedings of international conference on intelligence analysis*, vol. 5. McLean, VA, USA, 2005, pp. 2–4.

[13] C. Ware, *Information visualization: perception for design*. Elsevier, 2012.

[14] A. Figueiras, "Towards the understanding of interaction in information visualization," *2015 19th International Conference on Information Visualization*, pp. 140–147, 2015.

[15] B. Shneiderman, "Tree visualization with tree-maps: 2-d space-filling approach," *ACM Transactions on graphics (TOG)*, vol. 11, no. 1, pp. 92–99, 1992.

[16] (2015) Enron email dataset. [Online]. Available: https://www.cs.cmu.edu/ ~./enron/

[17] B. Klimt and Y. Yang, "Introducing the Enron corpus." in *CEAS*, 2004.

[18] J. Diesner and K. M. Carley, "Exploration of communication networks from the enron email corpus," in *SIAM International Conference on Data Mining: Workshop on Link Analysis, Counterterrorism and Security, Newport Beach, CA*. Citeseer, 2005, pp. 3–14.

[19] R. Rowe, G. Creamer, S. Hershkop, and S. J. Stolfo, "Automated social hierarchy detection through email network analysis," in *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*. ACM, 2007, pp. 109–117.