EG UK Computer Graphics & Visual Computing (2018)
G. Tam and F. Vidal (Editors)

# RiverState: A Visual Metaphor Representing Millions of Time-Oriented State Transitions

Richard C. Roberts[1], Dylan Rees[1], Robert S. Laramee[1], Paul Brookes [2], and Gary A. Smith [2]

[1]Computer Science, Swansea University, Swansea, Wales
[2]QPC Labs, QPC Ltd, The Harlech Building, Theatre Clwyd Complex, Mold, Flintshire, www.qpc.com

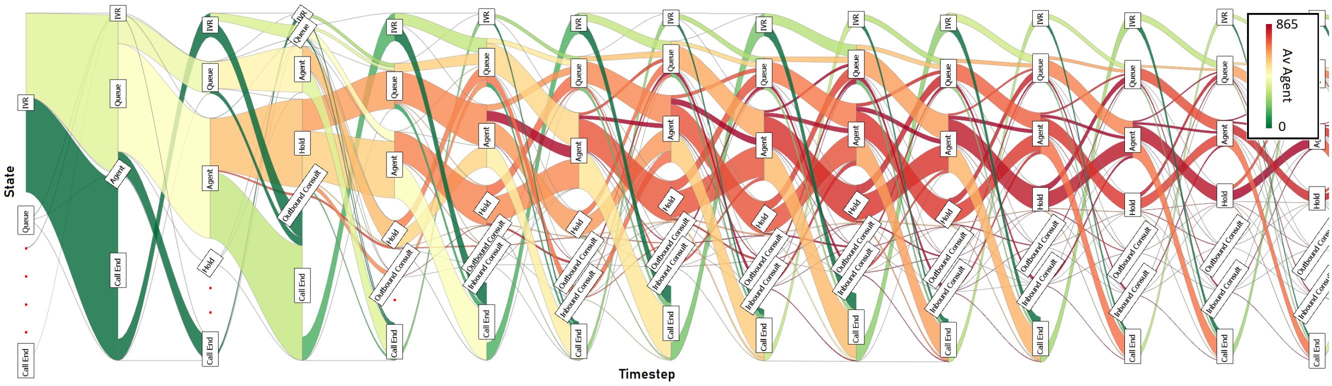

**Figure 1:** *We present the RiverState application. State transitions flow from left to right. This data is taken from a call centre, showing the flow of the collective customer journey. Colour is mapped to the average call length at each transition.*

**Abstract**

*Developing a positive relationship between a business and its customers is vital to success. The outcome of any customer interaction can determine future patronage of the business. Many industry's only point of interaction with their customers is through a contact centre where everything from sales to complaints are handled. This places tremendous importance on the operational efficiency of the contact centre and the level of care provided to the customers. These customer interactions are recorded and archived in large databases, but undertaking insightful analysis is challenging due to both the size and complexity of the data. We present a visual solution to the tracking of customer interactions at a large scale. RiverState visualises the collective flow of callers through the process of interacting with a contact centre using a river metaphor. We use finite state transition machines with customised edges to depict millions of events and the states callers go through to complete their journey. We implement a range of novel features to enhance the analytical qualities of the application, and collect feedback from domain experts to analyse and evaluate the use of the software.*

**CCS Concepts**

***•Human-centered computing*** → ***Visual analytics;*** *Visualization techniques; Information visualization;* ***•Mathematics of computing*** → *Exploratory data analysis;*

## 1. Introduction and Motivation

> "Your most unhappy customers are your greatest source of learning" - Bill Gates [Gat99]

The operational determinants of call centre satisfaction have long been researched [FKH*00]. However, as industries move away from bricks-and-mortar retailers towards online outlets the customer experience is changing in nature [KMHT17, LYMB05]. Call centres strive to understand and evaluate the service that they provide but due to the complexity and scale of the data are often unable to see the 'big picture'.

We utilise a large dataset of around 180,000 phone calls from call centre interactions and visualise the collective flow of calls through a call centre. Each caller typically enters the call through the IVR (Interactive Voice Response) and is then routed into a queue. From there they are transferred on to speak to an agent. Many call centre interactions are more complicated than that. Callers are often placed on hold, agents may need to call other agents to consult, callers may have been routed to the wrong department, or the caller might have multiple reasons for calling and therefore is transferred multiple times. Examining the 'average' customer journey in a contact centre is a non-trivial task amongst hundreds of thousands of calls.

In order to overcome these challenges we represent the call centre data with a finite state transition machine. Within the dataset we have repeat callers. When displayed, previously unseen trends can be identified and explored. We adapt the design of a Sankey diagram to show the volume of callers traversing different states along a horizontal axis mapped to each time step.

We implement brushing interactions that enable the user to select only the states they wish to view at whichever time interval they choose. The user can also break down transition types into subcategories (Generic agent interaction –> department interaction).

The main challenges associated with this project are that large state transition data does not fit directly into traditional Finite State Transition (FST) visualisation models. In general, state transition diagrams do not represent the frequency of transition. A typical Sankey diagram does not show connections of states beyond one transition. Transition paths cannot be inferred from any state. Our dataset is also unevenly distributed towards the earlier states, therefore longer sequences of transitions are more difficult to explore. The contributions this paper provides are:

- A novel adaptation of a finite state machine using a large call centre data set.
- A novel interactive scaling feature to examine the lower frequency transitions whilst maintaining context for the rest of the transitions.
- Interactive filtering operations that enable the user to focus on subsets of interesting calls.
- A journey tracing feature that links connections throughout the entire RiverState flow as well as a details-on-demand interaction that displays the complete path of all calls passing through a state.
- Domain expert evaluation of the software from our data providers.

In the next section we discuss the related work surrounding this topic. In Section 3 we discuss the application background and describe the data we are working with. Section 4 we discuss the finite state transition visual layout and the interactive features implemented in the software. Section 5 discusses the use of this software in a case study of call centre data, revealing observations that were previously undiscovered. In Section 6 we evaluate the application using feedback from domain experts and then we provide conclusions and future work in the final section.

## 2. Related Work

McNabb and Laramee offer a useful starting point for reviewing related literature [ML17]. The finite state machine is an abstract model of computation in which machines can only exist in one of a finite number of states at a time [Cho78]. State transitions refer to the movement between these states. Typical depictions of these transition models involve standard diagrams where state nodes are connected with transition arrows [VHVDWVW01]. Whilst this technique for visualising FST data is sufficient for small and simple models, larger and more complex models become more difficult to interpret.

Some state transition machines can have an unlimited number of states and therefore may need alternative methods to convey their data [VHVDWVW01]. Incremental improvements have been made to these methods, and new types of visual state transition designs have been created. Movements away from static visualisation towards interactive methods have improved the utility of the visual state transition designs [VHVDWVW01, VHvDWvW02]. These designs also push the diagrams from 2D into 3D, enabling a more in-depth exploration of the data. In these designs transition edge thickness is used to denote the volume of transitions – this distinction now accommodates high volume data, however, this does not accommodate the exploration of transitions with low volume in the same design. These methods also use very large state spaces by use of clustering states together [GVH06]. Blaas at al. [BBG*09] explored smooth transitions between states using animations and curved lines, however the visualisation does not differentiate between timesteps.

Varying types of transition graphs have been developed, often focusing on tree-like structures to visualise the data [PVW06], each level down the tree hierarchy depicts a new transition. Whilst this means you can show complete transition paths, this method does not scale as effectively. Arc diagrams are also used to enhance the level of analysis capable through the software.

State transition theory has been applied to a range of different data types, including fluid simulation data [JS10], software testing data [Cho78], and biological data [MOO*11] to name a few. Though not all use the standard node and edge diagram to depict the transitions. Other standard visualisation methods are sometimes used such as basic plots and heat maps. Whilst state transition data *can* be shown using these methods, they focus on exploring different aspects of the data.

Work processes are often visualised as flows which exist on a timeline [BMBW15, BKW13], here the data smoothly flows along an x-axis, but does not show the transitioning of states, but instead the existence of one state changing in value. Our method looks at the smooth motion between fixed states.

State transition visualisation is closely related to event sequence visualisation. The difference being where the focus is placed. The former looks at the way events transition into other event states and the latter places emphasis on the events themselves and the metadata associated with them. Many event sequence visualisations depict each event and sequence separately, avoiding frequency approaches [MLL*13, GS14, WGGP*11].

Our case study dataset of call centre interactions has previously been explored through combinations of frequency and non-frequency based approaches. A treemap has been used to produce an overview of the data, which can then be drilled down into, exploring individual events [RTL*16]. Parallel coordinates have also been used to identify trends in the overall call attributes [RLS*18],

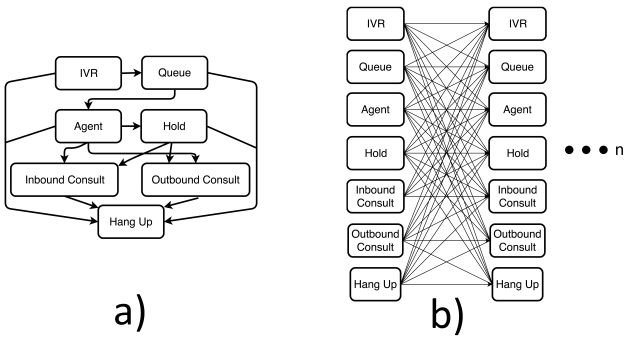

**Figure 2:** *This figure shows the differences between a) typical FST diagrams and our method of displaying the data b) whereby each state node is repeated for each time step.*

but not in the transitions. The work here is the first transition-centred design.

Other work with similar data rarely reaches the visualisation stage of analysis. Operational efficiency of call centres is heavily researched [BM02, AAM07, BGM*05], but mostly through non-visual reporting and would rarely explore visually beyond basic visual techniques.

In this research we apply finite state transition data to a modified Sankey diagram. This visualisation method typically depicts hierarchical data that contains elements of 'flow' to it [Sch08]. The popular javascript visualisation library D3.js can be used to create Sankey diagrams [DDD].

These diagrams are used to model data pathways. Energy usage is a common use case for this visualisation method [Sch08, SHS14, KKP09], however these are static images created to portray one data attribute. Explorative analysis is also possible using a Sankey diagram [RHF05], whereby interaction can be used to edit data within the window as well as mouseover edges for further information.

More complex Sankey diagram software overlay histogram bars at transition points to show an edge thickness comparison [WG11]. The software also enables users to interact with the diagram and mouseover to see a basic transition path rendered from the selected point. Event sequences have also been used in Sankey diagrams to show clickstream data [ZLD*15]. Our concept is similar to this, however their focus was to convert the diagram into a matrix view. Our goal is to further enhance the Sankey diagram by adding new, interactive features.

## 3. Application Background and Data Description

The goal of this software is to present the flow of state transitions throughout a journey. Our industry partners have provided us with a dataset from a customer facing call centre with the goal of exploring what the customer journeys look like through the call centre landscape. This data is broken down into seven different primary states.

1. IVR - Pre-recorded audio routing system
2. Queue - Waiting to speak to an agent

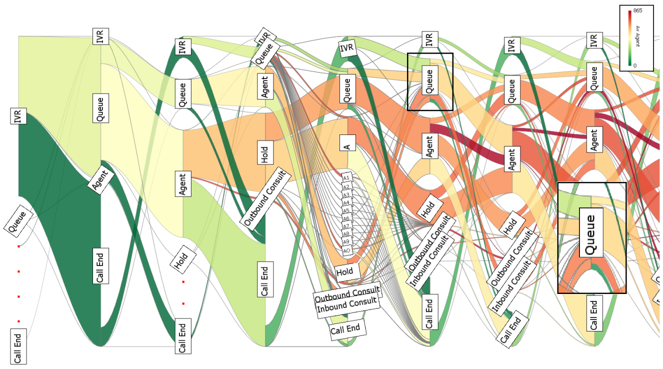

**Figure 3:** *This figure demonstrates a typical RiverState view as state transitions flow between time-oriented events (Q2). The agent state at timestep 5 has been broken down into 11 agent sub-states corresponding to different departments in the call centre (Q4). The bottom state at each axis represents a call end (Hang up state). Edges transitioning from these states represent repeat callers (Hang up –> IVR).*

3. Agent - A customer speaking to an agent
4. Hold - Agent pauses the call to handle customer concern
5. Inbound Consult - A second agent is brought into a call.
6. Outbound Consult - The current agent calls a second agent privately to consult.
7. Hang Up - The call is terminated

In addition to these states, the 'Agent' state data is broken down into ten different departments, which can reveal why the call was placed e.g. tech support, billing, sales etc.

The data includes a number of meta-data variables associated with the sequences of event transitions. These metrics are mostly duration and frequency values associated with the different states. Additionally, a call cost metric, a customer 'effort' metric (CES), and customer feedback score (NPS) are also included.

The dataset contains 24 hours of data, and almost 180,000 phone calls. Some of these are repeat calls from the same customers. To show this, the flow of our visualisation links the end of one call (Hang Up –> IVR) to the beginning of the next in order to fully represent the customer journey.

**Application Questions** Our exploration of the call centre dataset is tasked to answer a number of questions from the domain experts about the nature of the customer behaviour and the internal operations of the call centres.

- (Q1) Are there any patterns in the customer journeys?
- (Q2) What does the average customer journey look like?
- (Q3) Can we see a collective customer journey overview?
- (Q4) Can we break down the callers by agent department?
- (Q5) What factor affect the NPS (customer feedback) metric?
- (Q6) Can we identify groups of similar callers?

## 4. RiverState: Finite state transitions for millions of events.

We present the visual designs created to explore the state transition data. The main modification to the traditional FST machine design

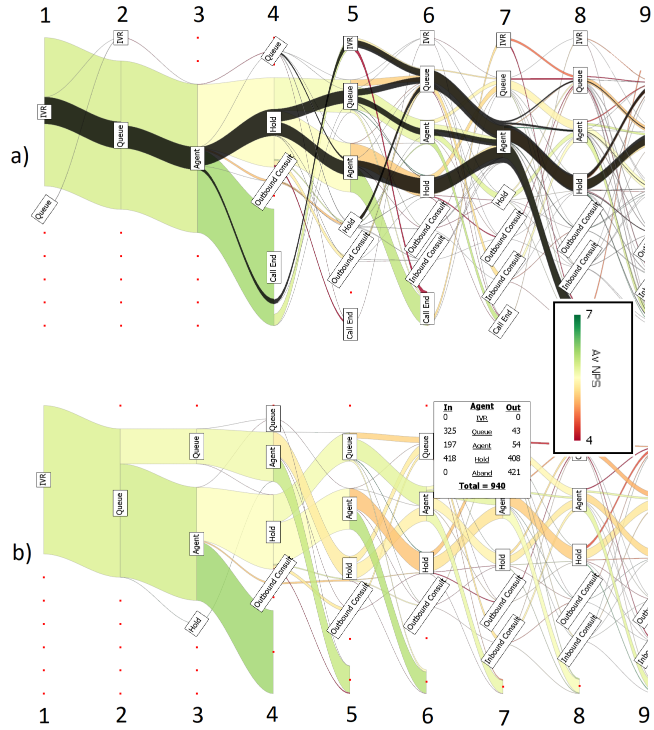

**Figure 4:** *This figure demonstrates the interactive features of the RiverState application. Image a) shows a filter applied to the agent state at axis 3. This also demonstrates the traceroute feature where the user has moved their mouse over the agent state at timestep 7. Image b) shows a meta-data filter applied so that only callers who responded to an NPS survey about the call are shown (Q1,Q2). Both images are coloured by this NPS score. Image b) also demonstrates the details on demand mouseover feature which displays the numerical data behind the transitions at the selected state.*

is that a state node exists for each state at every time step. Typically, an FST diagram features one node per state which limits the volume of data the design can present with the diagram (see Figure 2). In our design, each state node is repeated for each time step in 'axis'-like columns. Adding the time step dimension into the FST diagram enables users to find trends in repeat visits to states as well as reveal more information about complete journeys as opposed to singular transitions.

We recommend that the reader views the accompanying video in order to see the full feature set in use. The video can be viewed here: https://vimeo.com/273301145

### 4.1. RiverState - A Sankey Modification

In order to display frequency of each transition we use weighted edges and style the transitions like a Sankey diagram. Each edge that protrudes from a state node, lines up on their corresponding state node without overlap. Edges from a state node flow continuously into the next state node, maintaining the frequency of transitions originating from the first state node (see Figure 3). This reduces data occlusion and increases the aesthetic quality of the design. The rendering window can be scrolled sideways to move further along the state transition sequences. The design of this layout includes up to 17 state types with an unlimited number of time steps.

### 4.2. Interactive Brushing

We implement two complementary brushing systems in the application.

**Direct State Brushing** The first brushing option interactively filters calls based on the states they traverse. The user can click on any state (or sub-state) to apply a filter to it. Depending on the type of brush selected, the filter will either remove all calls that fall outside of that state, or filter all calls passing through that state. Being able to filter out calls at specific states enables the user to remove noise in a high volume dataset whilst simultaneously being able to focus on an area of interest. Enabling a filter whereby all calls outside of a brush are filtered enables the user to focus on specific patterns in the state transition journey and then infer future behaviour based on the data selection. If the user applies a brush to multiple states at the same timestep, an OR filter is applied, keeping calls that pass through either state. Examples of this filter type is shown in Figure 4 a) & Figure 5.

**Meta-data Brushing** The second brushing system utilises the meta-data associated with the state transitions (Q6). Stored in our call centre dataset are metrics such as call length, queue duration, customer feedback, event count etc. A full interface of sliders for each attribute enables the user to select a precise data range and update the RiverState graphics containing only values within the filter range. An example of this brushing can be seen in Figure 4 b). When the user adjusts the slider of the meta-data brush, the RiverState flow automatically updates to show just the selected data. The smooth transition of this can be seen in the accompanying video.

Incorporating both of these brushing systems into the application gives the user full control over the sub-selection of callers they would like to focus on. Using both simultaneously creates a powerful tool for pattern discovery and the analysis of long state transition sequences.

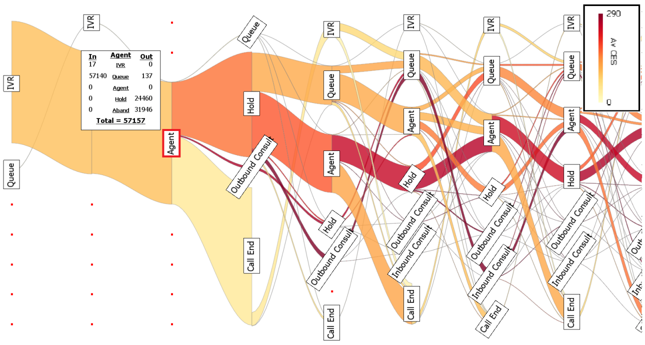

**Figure 5:** *This image shows the beginning of an average call (Q2). IVR –> Queue –> Agent. At that point slightly more than half of calls terminate there, and the other half continue to be placed on hold. The detail on demand tooltip is used to show the exact number of calls entering and exiting the states.*

### 4.3. Details on Demand

The RiverState application shows the flow of transitions over time through a visual depiction. A numerical representation of this data is available through a details-on-demand feature whereby the user hovers the mouse over a state and the in-flow and out-flow calls of the state is shown in a pop-up dialogue box. The size of the dialogue box can be changed through a slider user option.

This feature is useful when the scale of transitions is difficult to determine and the user is unable to see the volume throughput of each node. The details window is shown in Figure 4 b). In this window, the left column depicts the number of calls entering the currently selected state from each of the states labelled in the centre column. The right column of numbers count where these calls go after transition through the current state. i.e. Figure 4 b) shows 325 calls entering the agent state from the queue, 197 from another agent, and 418 from being kept on hold.

### 4.4. State Labelling

A user option enables states to be labelled with the state name. The size of these labels can be adjusted to ensure the user is able to remain informed of the state names without the labels occluding the graphics. We rotate larger boxes if the label size overlaps with other labels. This way more of the state connections can be seen whilst keeping a full view of the label.

### 4.5. Journey Tracing

A fundamental limitation of the standard Sankey diagram is that no complete state transition journey can be inferred. At each state, only one step in each direction can be viewed.

We have implemented a feature that enables the user to move their mouse over a state, and a new route is drawn, highlighting the complete path sequence of all journeys preceding and succeeding that state. An example of this feature can be seen in Figure 4 a). This feature shows the preceding journey that calls from the selected state have transitioned through but also their future journey. This helps identify new patterns in the call centre activities and customer behaviours.

As the states expand in size, they recalculate their position in the $y$ dimension, automatically adjusting to the increase in either their own size or the size of states along that timestep. This is the first step to prevent data overlap when growing the edges as it ensures the most efficient utilisation of space.

### 4.6. Transition Zooming - Exploring low-frequency transitions

In order to address the challenge of unevenly distributed data, we implement a novel feature that can explore patterns among low-frequency transitions. We implement a slider that increases or decreases the transition edge width so that more of the screen space can be utilised. However, due to our dataset being unevenly distributed towards early states, these state transition edges quickly expand to overlap whilst later transition edges do not grow enough to be clearly interpreted.

We address this by creating a fixed maximum size that each tran-

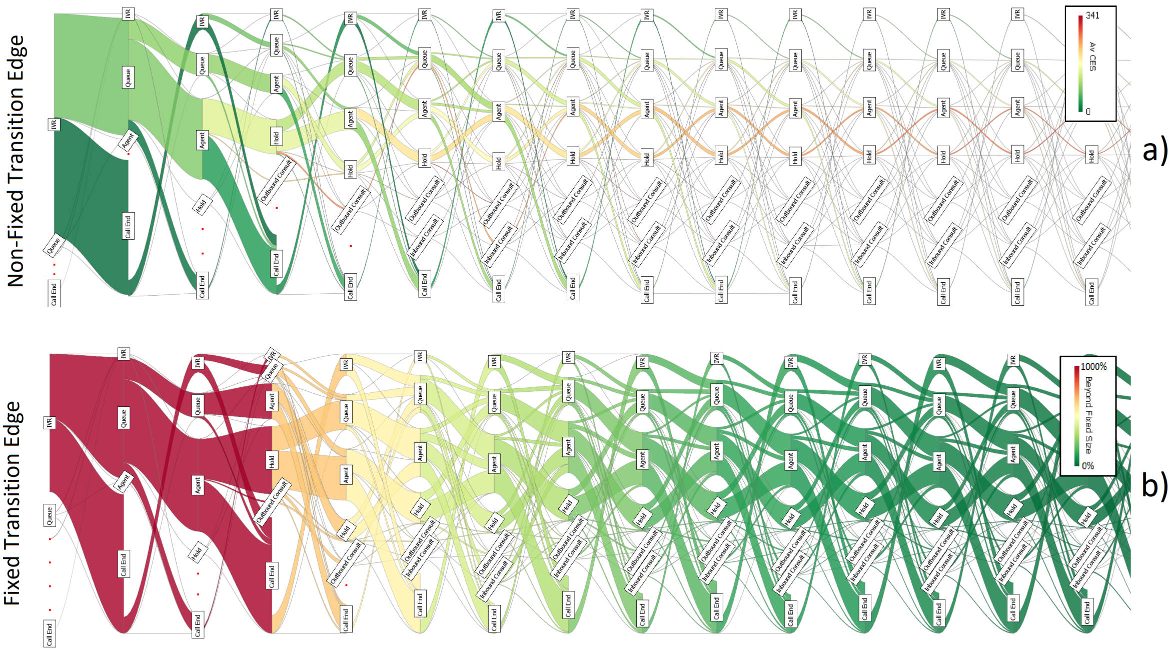

**Figure 6:** *a) shows the RiverState flow before the edge thickness is increased. b) shows the grown edges whereby a maximum edge width is imposed. We can now observe the customer journey trends later along the x-axis without obscuring the initial transitions (Q3).*

sition axis can expand to. Once the maximum size is reached, all states on that axis stop growing in size and instead begin to change in colour (see Figure 6). This way, the less frequent transitions can be seen and interpreted whilst maintaining the context of the earlier transitions.

### 4.7. Colour Mapping

A number of different colour metrics have also been implemented in the application. The user can select from the list of meta-data call attributes to colour the transitions by. Averages of each meta-data characteristic are calculated and used to provide the user further insight into the trends in the data.

Another user option enables the user to select the colour map that the application uses to depict the data. We have implemented a range of different colour maps, including colour blind friendly, diverging and sequential based on Harrower and Brewer [HB03].

An additional feature we implement enables the user to change the colour map's upper range. This is useful when brushes remove a lot of variance in the dataset and the old colour map range is too large to explore any differences between values.

## 5. Exploring Customer Journeys through the call Centre Landscape

The primary goal of the RiverState application is to explore the industry provided dataset of call centre interactions. We were tasked with identifying what customer journeys through their call centre experience looks like. The RiverState application has enabled us to make unique and interesting insights into the customer behaviour and identify previously unseen trends in the data.

### 5.1. Repeated pattern with later timestep calls

RiverState initially renders state transitions so that transition edge thickness is maintained along the graph. However, this results in later transitions to be too small for analysis, and increasing the overall width of the edges creates an obfuscated overlap of earlier nodes. Our transition zooming feature enables exploration of low volume transitions by enforcing the maximum size of a transition edge and instead increments the colour hue when the size value exceeds this.

Using this feature we observe that these lower frequency timesteps take on a repeated pattern which uniformly decreases in call volume over time due to calls ending. The initial five state-transitions follow a completely different pattern to state transitions after this point. See Figure 6.

### 5.2. Average Customer Journey

Using the RiverState application we can easily find the average journey a customer experiences using a combination of the state brushing and details on demand (Q1,Q2). See Figure 5.

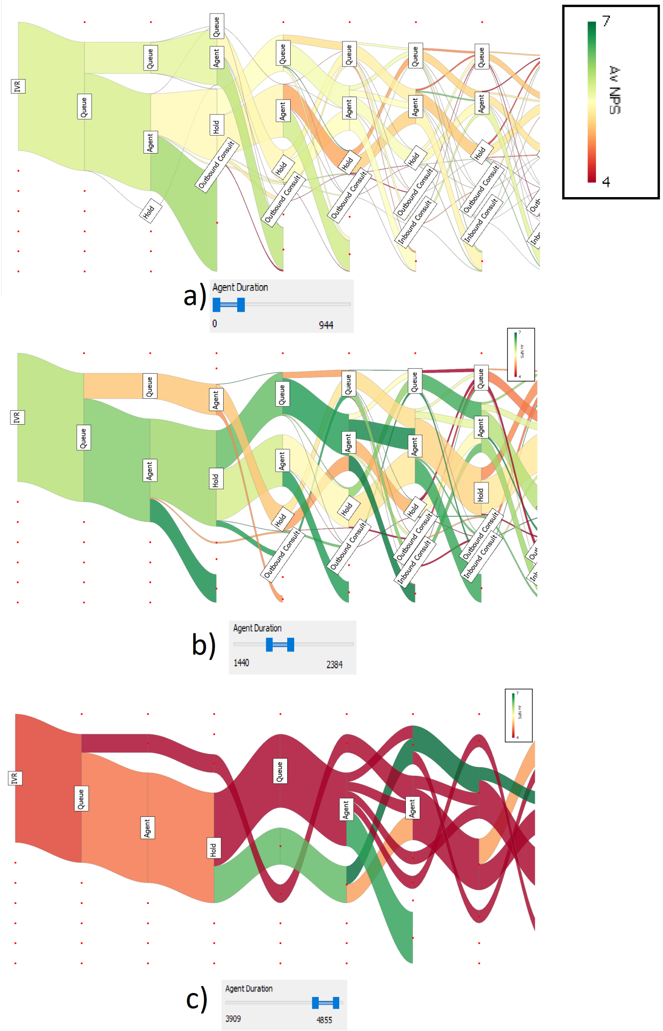

**Figure 7:** *This image shows the meta-data brush applied to the agent duration axis where colour is mapped to the NPS (Q6). You can see that image b) is generally more green than the other two, and image c) is more red. The journeys don't vary too much between a) and b), but c) looks different due to the longer nature of the calls, and less of them.*

Starting at the first axis timestep, the user can select the most significant state by call throughput, using the details on demand tooltip if the largest transition is not immediately apparent. It becomes clear that a typical call enters via the IVR, moves into the queue, and then transitions into speaking to an agent. Using the mouseover tooltip, we can see that out of 180,000 calls that enter the call centre, only 57,000 end up speaking to an agent, and 31,000 of those end the call immediately after the call ends. This suggests that either callers aren't making it to an agent due to long wait times or that their concern is resolved earlier (in the IVR information system). Of those who spoke to an agent, almost 25,000 (43% – or 13% of the total callers) are placed on hold – continuing their call.

### 5.3. The Net Promoter Score - A measure of satisfaction

The NPS is collected from a survey sent out to callers immediately after their call centre interaction. They are asked to, on a scale of one to ten, respond with how likely they are to recommend the com-

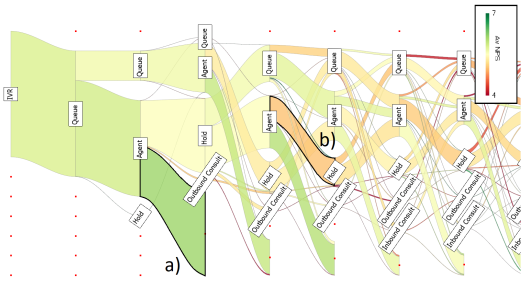

**Figure 8:** *This figure shows a meta-data filter including only NPS response callers where the colour is also associated with the NPS score (Q1,Q5,Q6). You can see that customers who ended the call in the shortest time frame (i.e. IVR –> Queue –> Agent –> End) are satisfied with the service (Transition a). Callers who were placed on hold for the second time are dissatisfied (Transition b).*

pany to family or friends. This feedback measure provides us a link between the journey the customer experiences and their own view of that experience.

**What influences the NPS?** We observe that a number of meta-data variables are strongly correlated with the NPS metric (Q5). Firstly, we look at the total time a caller spends with an agent. In Figure 7, we notice that as the meta-data slider moves along the brushing range, the colour of the RiverState graphic shifts slightly between image a) and image b) showing an increase in NPS, but after that point the scores decrease rapidly, shown in image c). This suggests that there is an optimum amount of time a caller should spend with an agent. Too long and it's a waste of time, too short and they don't feel satisfied or their problem isn't solved. This peak point lies between 20 and 40 minutes.

This phenomena is also observable for a number of other meta-data variables. The wait time has a strong correlation between duration and NPS, though the relationship is more linear than agent duration. A high number of agent events is also a good predictor of a low NPS. To see a demonstration of this, please see the accompanying video.

**Quick calls are better calls:** We also observe that the later timesteps are less likely to be favoured positively (Q1). Figure 8 shows the flow of calls entering from the IVR into the queue, and then most speak to an agent. The highest rated transition of callers end the call immediately after that. The first obviously unhappy transition is from the their second agent state about to be placed on hold. The RiverState application highlights scenarios that call centres should try to avoid – by warning of repeated transitions between agents.

## 6. Domain Expert Feedback

We have been collaborating with QPC Ltd., the domain experts in this case since 2014. All of the features we implement are inspired by our discussions with them.

To evaluate the RiverState application, we performed a multi-stage interview study with three domain experts who provided us with the industry data. Their knowledge of the dataset enables us to thoroughly test the capabilities of the software by searching for previously unseen insights. We perform an initial feedback session two months before the primary session to collect feedback and make modifications. During this first session, the main feature improvement was to broaden the filtering capabilities. From this, we implemented the meta-data filtering feature. Both sessions were video recorded and the audio was transcribed in order to analyse the feedback.

### 6.1. RiverState

We begin the primary feedback session by explaining the general concepts of the RiverState application. Expert 2 says *"For me the main part we've not focused on is the call end aspect"*, pointing at the flow of calls from the call end state to the IVR state, signifying customers calling back - *"I think there's definitely some stuff there that we've not looked at which could be interesting. Where calls on aggregate build to unhappiness."* Expert 3 says *"Having no call back is a good indication of a resolution to the call. If the call is resolved straight after talking to an agent, we know that's good. But what's happening to all of these calls over here,"* – he gestures to the rest of the plot where the resolution is not found. Expert 1 says *"The flow works. It helps us look at where this mass of calls are moving."*

We ask what features in the RiverState stand out to them *"I think what's interesting is how long it takes for some calls to make it to a conclusion. A larger proportion get bounced around than they really should as we know that's not good for their satisfaction level."* We ask what the software would be useful for – Expert 3 states *"It's a really good way of showing where journeys complete. To look at the difference between long and short calls."*

### 6.2. Edge Zoom Feedback

The Experts question the fact that we can't see later state transitions due to the low frequencies. We demonstrate the transition edge expansion feature without a maximum size imposed to show how quickly the plot gets obfuscated. *"Looks like a Jackson Pollock"* says Expert 1, noting how confusing the image has become. We drag the slider down, imposing a new maximum size for the transition axis which rescales the graphic. *"Ahh that's much better!"* Expert 2 claims. *"You can see the flow much better this way and those changing colours are a nice touch."*

We discuss the typical call pattern that appears when you expand the low-frequency calls. Expert 2 says *"So it's interesting, I guess this is driven by the nature of how you contact the call centre. You have to go in via the IVR, you need to be routed into a queue and the aim is to speak to an agent. I guess those are the desired states, so ultimately everything gravitates towards that – this is a good template of a typical journey."*

### 6.3. Brushing

We continue the demonstration, showing the basic brushing features, starting with the direct state brushing. *"What you've got here is good."* Expert 2 says. *"[Brushing the state directly] is a nice* *way of focusing on specific events and filtering out noise,"* he continues. We expand the agent state out, splitting the agent calls into the agent departments. *"Oh nice. The ability to interact and click into it is really nice,"* says Expert 3. We discuss for a time the value of knowing what department the caller is speaking to, and the effect that has on customer satisfaction. Expert 2 claims *"I think for me, the flexibility is the key part here. If you can scope down into these agent events and directly apply a filter to one of those, there's loads of potential with that."*

*"It certainly gives us a view that we haven't got at the moment."* Says Expert 1. *"Especially with the volume of data this shows. We only work on much smaller volume of calls."* Expert 3 says *"It does give you the power to work with more calls to begin with and then filter down. We don't currently have that."*

We show the meta-data brushing using the sliders by applying a filter to show only NPS response calls and then smoothly brushing the wait duration meta-data value. *"Nice, that's good,"* says Expert 1. We continue to apply different meta-data filters and Expert 2 says *"These meta-data filters are key to this software. Without them it's difficult to look closer at the data. It's good to be able to choose your own exact window of data and then see how that might vary from others."* He continues, *"it gives you a lot more options in terms of filtering from what we've ever done."* Expert 2 agrees – *"This feels like it's got a lot more flexibility."*

### 6.4. Details on Demand

As we finish demonstrating the brushing features, Expert 1 states *"It would be good if you could have a tooltip where you hover over the state to see what the values for it are."* We demonstrate our details on demand feature showing just that. *"That's useful. I like having the numerical values visible."*

### 6.5. Journey Tracing

They make a number of enquiries about being able to see where calls at a state have travelled from. Expert 3 states *"So we tried a basic Sankey diagram before, but we couldn't see very far in the data. like you said, it showed the next or previous steps but nothing more than that."* We show them the journey tracing feature that we developed to overcome this issue.

We show some more examples and he continues, *"I like how you can maintain the focus of the rest of the flow, and still see the journey traces. I suppose the same data is shown then you apply a brush to that point, but this way is more temporary and better to explore."*

### 6.6. Colour Mapping

We demonstrate our analysis of the NPS score using the colour mapping features, changing the range of the map to show the subtle differences in the different state transitions. We highlight a hold state that appears to be the first obviously unhappy transition. *"So this makes sense, this is the second time the caller is placed on hold and so it is going to be negative for them... I guess this visually reinforces what we knew about NPS scores reflecting how efficient the service is."* says Expert 2.

Expert 1 says, *"Colour is such an important component of these applications. Without it, we wouldn't see half the things we wanted to... Subtle differences in the data can be made clearer."* He continues to say *"NPS is the perfect example of this, looking at how these different transitions affect the NPS is a clear indicator to me that this works."*

### 6.7. Additional features

*"So this is very close to what we're trying to get to."* Expert 2 claims, *"We like the filtering options, and it would be good to see even more of those."* Expert 1 says *"We would also love to see these filters applied to a journey tracer."* The system would create more of a focus + context approach to the brushing.

As we conclude the session, Expert 1 summarises by saying *"That's the challenge of visualisation, there's so much data you're trying to pick out a point of interest, or key moment, or looking for somewhere to drill down into to explore. There are so many different permutations the visualisation can exist in, there's no real answer. We really just want to highlight or pinpoint where large amounts of effort is placed by the customer which might tip them into having a negative view. This software approaches this challenge in a really nice way."*

## 7. Conclusions and Future Work

In conclusion, we present novel application that presents finite state transition data as a flowing river metaphor. Our case study data contains call records from a large call centre and using the RiverState application we explore the operations and limitations of the call centre service whilst looking at the complete customer journey. Adapting the traditional Sankey diagram enables us to look at our industry dataset in a different way, identify customer behaviour and create an interface with the meta-data associated with the calls. The initial RiverState data overview can be filtered by direct state brushing or a comprehensive meta-data brush. A colour mapping feature enables the user to select a range of meta-data averages to colour the transitions by. A number of interactive mouse-over features create an environment whereby the user can explore the customer journey in detail – tracing the flow of the transition journey throughout the plot, and providing details on demand to the user.

We evaluate this software using domain expert feedback which demonstrates the real-world effectiveness of the visual design developed in this paper. The evaluation explores how the application can be used by industry and how each feature contributes to their end goal.

We plan to further develop this concept, apply more industry appropriate features conceived in the primary feedback session. We would like to implement a user option to build the journey tracing path feature based on meta-data associated with the call. We would also like to further incorporate the meta-data values into the plot through a glyph or additional mouseover options. The goal of this research is to create a useful application to present complex industry data in a unique way. We have produced a useful tool validated by our industry partners.

## 8. Acknowledgements

We would like to thank all who have contributed to this research. We thank QPC Ltd. for their financial support and their supply of the call centre data, and we thank KESSII who have been the primary funding body for this research. KESS is part funded by the Welsh Government's European Social Fund (ESF) convergence programme for West Wales and the Valleys. In addition to this, would like to thank Liam McNabb for providing valuable feedback on this work.

## References

[AAM07] Aksin Z., Armony M., Mehrotra V.: The modern call center: A multi-disciplinary perspective on operations management research. *Production and operations management 16*, 6 (2007), 665–688. 3

[BBG*09] Blaas J., Botha C., Grundy E., Jones M., Laramee R., Post F.: Smooth graphs for visual exploration of higher-order state transitions. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 969–976. 2

[BGM*05] Brown L., Gans N., Mandelbaum A., Sakov A., Shen H., Zeltyn S., Zhao L.: Statistical analysis of a telephone call center: A queueing-science perspective. *Journal of the American statistical association 100*, 469 (2005), 36–50. 3

[BKW13] Burch M., Kull A., Weiskopf D.: Aoi rivers for visualizing dynamic eye gaze frequencies. In *Computer Graphics Forum* (2013), vol. 32, Wiley Online Library, pp. 281–290. 2

[BM02] Batt R., Moynihan L.: The viability of alternative call centre production models. *Human Resource Management Journal 12*, 4 (2002), 14–34. 3

[BMBW15] Burch M., Munz T., Beck F., Weiskopf D.: Visualizing work processes in software engineering with developer rivers. In *Software Visualization (VISSOFT), 2015 IEEE 3rd Working Conference on* (2015), IEEE, pp. 116–124. 2

[Cho78] Chow T. S.: Testing software design modeled by finite-state machines. *IEEE Transactions on Software Engineering*, 3 (1978), 178–187. 2

[DDD] Data-Driven-Documents: D3-sankey. URL: https://github.com/d3/d3-sankey. 3

[FKH*00] Feinberg R. A., Kim I.-S., Hokama L., De Ruyter K., Keen C.: Operational determinants of caller satisfaction in the call center. *International Journal of Service Industry Management 11*, 2 (2000), 131–141. 1

[Gat99] Gates B.: Business at the speed of thought. *Business Strategy Review 10*, 2 (1999), 11–18. 1

[GS14] Gotz D., Stavropoulos H.: Decisionflow: Visual analytics for high-dimensional temporal event sequence data. *IEEE Transactions on Visualization and Computer Graphics 20*, 12 (2014), 1783–1792. 2

[GVH06] Groote J. F., Van Ham F.: Interactive visualization of large state spaces. *International Journal on Software Tools for Technology Transfer 8*, 1 (2006), 77–91. 2

[HB03] Harrower M., Brewer C. A.: Colorbrewer. org: an online tool for selecting colour schemes for maps. *The Cartographic Journal 40*, 1 (2003), 27–37. 6

[JS10] Jänicke H., Scheuermann G.: Visual analysis of flow features using information theory. *IEEE Computer Graphics and Applications 30*, 1 (2010), 40–49. 2

[KKP09] Kymakis E., Kalykakis S., Papazoglou T. M.: Performance analysis of a grid connected photovoltaic park on the island of crete. *Energy Conversion and Management 50*, 3 (2009), 433–438. 3

[KMHT17] Knapp A.-K., Marchand A., Hennig-Thurau T.: How to survive in a digital world? a comprehensive analysis of success factors for brick-and-mortar retail stores: An abstract. In *Creating Marketing Magic and Innovative Future Marketing Trends*. Springer, 2017, pp. 301–301. 1

[LYMB05] Laroche M., Yang Z., McDougall G. H., Bergeron J.: Internet versus bricks-and-mortar retailers: An investigation into intangibility and its consequences. *Journal of Retailing 81*, 4 (2005), 251–267. 1

[ML17] McNabb L., Laramee R. S.: Survey of surveys (sos)-mapping the landscape of survey papers in information visualization. In *Computer Graphics Forum* (2017), vol. 36, Wiley Online Library, pp. 589–617. 2

[MLL*13] Monroe M., Lan R., Lee H., Plaisant C., Shneiderman B.: Temporal event sequence simplification. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2227–2236. 2

[MOO*11] Miwakeichi F., Oku Y., Okada Y., Kawai S., Tamura Y., Ishiguro M.: Detection and visualization method of dynamic state transition for biological spatio-temporal imaging data. *IEEE Transactions on Medical Imaging 30*, 3 (2011), 859–866. 2

[PVW06] Pretorius A. J., Van Wijk J. J.: Visual analysis of multivariate state transition graphs. *IEEE Transactions on Visualization and Computer Graphics 12*, 5 (2006), 685–692. 2

[RHF05] Riehmann P., Hanfler M., Froehlich B.: Interactive sankey diagrams. In *Information Visualization, 2005. (INFOVIS) 2005. IEEE Symposium on* (2005), IEEE, pp. 233–240. 3

[RLS*18] Roberts R., Laramee R. S., Smith G. A., Brookes P., D'Cruze T.: Smart brushing for parallel coordinates. *IEEE Transactions on Visualization and Computer Graphics* (2018), 1–1. doi:10.1109/TVCG.2018.2808969. 2

[RTL*16] Roberts R., Tong C., Laramee R., Smith G. A., Brookes P., D'Cruze T.: Interactive analytical treemaps for visualisation of call centre data. In *Proceedings of the Conference on Smart Tools and Applications in Computer Graphics* (2016), Eurographics Association, pp. 109–117. 2

[Sch08] Schmidt M.: The sankey diagram in energy and material flow management. *Journal of Industrial Ecology 12*, 1 (2008), 82–94. 3

[SHS14] Soundararajan K., Ho H. K., Su B.: Sankey diagram framework for energy and exergy flows. *Applied energy 136* (2014), 1035–1042. 3

[VHVDWVW01] Van Ham F., Van De Wetering H., Van Wijk J. J.: Visualization of state transition graphs. In *Information Visualization, 2001. (INFOVIS)hum 2001. IEEE Symposium on* (2001), IEEE, pp. 59–66. 2

[VHvDWvW02] Van Ham F., van De Wetering H., van Wijk J. J.: Interactive visualization of state transition systems. *IEEE Transactions on Visualization and Computer Graphics 8*, 4 (2002), 319–329. 2

[WG11] Wongsuphasawat K., Gotz D.: Outflow: Visualizing patient flow by symptoms and outcome. In *IEEE VisWeek Workshop on Visual Analytics in Healthcare, Providence, Rhode Island, USA* (2011), American Medical Informatics Association, pp. 25–28. 3

[WGGP*11] Wongsuphasawat K., Guerra Gómez J. A., Plaisant C., Wang T. D., Taieb-Maimon M., Shneiderman B.: Lifeflow: visualizing an overview of event sequences. In *Proceedings of the SIGCHI conference on human factors in computing systems* (2011), ACM, pp. 1747–1756. 2

[ZLD*15] Zhao J., Liu Z., Dontcheva M., Hertzmann A., Wilson A.: Matrixwave: Visual comparison of event sequence data. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (2015), ACM, pp. 259–268. 3